I D E N T I F I C A T I O N

PRODUCT CODE:	MAINDEC-11-DQKUB-B-D VERSION /101/
PRODUCT NAME:	KD11-K Microdiagnostic
MAINTAINER:	Diagnostic Engineering
AUTHOR:	Don North
DATE CREATED:	18-Jan-1977
LAST REVISION:	22-Jun-1977

- TABLE OF CONTENTS -

1.0 ABSTRACT

The 'DCS' (Diagnostic Control Store) Module is a diagnostic tool specifically designed for the PDP-11/60 (KD11-K) Central Processor. Functioning as an alternate 2048 word control store, microcoded tests are executed to detect and isolate errors within the internals of the processor control and datapath hardware. Error indication information is provided by the DCS module; coupled with an indexed FAULT DIRECTORY, errors are resolved to the module level, when possible. Additional information is also provided resolving the error to a specific functional logic block. Significant benefits of this micro diagnostic approach are seen to be:

- Memory/I-O Device/UNIBUS independence
- Direct hardware microcontrol and visibility
- Extremely fast load and execution times
- Excellent coverage and resolution

2.0 REQUIREMENTS

A preliminary note:

Throughout this DCS User's Guide, the two terms "FAULT" and "ERROR" have been used interchangably. They both are to indicate "malfunctioning logic elements" (eg. busted IC's) and open/shorted ETCH runs, etc., in the unit under test. No distinction between the two terms is implied.

2.1 Hardware

To use the DCS, the following hardware is required:

1. DCS (Diagnostic Control Store) Module M7871 (KU116-BB)
2. PDP-11/60 (KD11-K) Central Processor
3. DL11-W Line Clock (required)
4. First 4K memory bank (required minimum)

2.2 Documentation / Listings

The available documentation for the DCS module user comprises the following items:

- DCS User's Guide (this document)
- FAULT DIRECTORY Listing; for module replacement information
- DCS Microcode Listing; when IC level debug is necessary
- DCS Maintenance Manual, Print Set; for detailed information on DCS hardware operation
- KD11-K Processor Maintenance Manual, Print Set; for IC level debug, base machine hardware specific information

Specific MAYNDEC component part numbers for the DCS documentation are as follows:

MD-11-DQKUB-*-D;
User's Guide, FAULT DIRECTORY (PAPER, 80 pages)

MD-11-DQKUB-*-LA;
DCS Microcode Listing (PAPER, 450 pages)

MD-11-DQKUB-*-FA;
User's Guide, FAULT DIRECTORY, DCS Microcode Listing (FICHE, 4 cards)

3.0 LOADING PROCEDURE

The DCS occupies slot 1 in the KD11-K processor backplane; thus if an ECS or UCS option is present, it must be removed. To load the DCS, the following sequence should be employed:

1. Power down the CPU
2. Remove the ECS/UCS option from processor slot #1, if present
3. Insert the DCS module into slot #1. Use caution while inserting the DCS module, as a slightly bowed board may require some gentle maneuvering to seat it in place.
4. Orient the DCS 'RUN/STOP' switch to 'STOP', and the 'NORMAL/VERIFY' switch to 'NORMAL'
5. Now power up the CPU

4.0 STARTING PROCEDURE

DCS execution can be initiated by two distinct methods:

4.1 Startup via the Operator's Console (KY11-P)

With the KD11-K CPU in "CONSOLE" ("HALTed") mode, simultaneously depress the KY11-P operator's console "CNTRL/DIAG" keys to start the DCS. (If no DCS/ECS/UCS happens to be present, there should be no effect). Section 5.1 (below) interprets the display on the operator's console while the DCS is executing, and after it stops. If this method fails to start the DCS, proceed to the next paragraph.

4.2 Startup via the DCS Switches

If it is desired for some reason to bypass use of the KY11-P operator's console to initiate the DCS, an alternative means is provided. This method would be used, for example, in the event that starting from the operator's console was not possible (using "CNTRL/DIAG") due to some hardware malfunction. The procedure is:

1. On the DCS module, set the 'NORMAL/VERIFY' switch to 'NORMAL'

2. Now set the 'RUN/STOP' switch (on DCS module also) to the 'RUN' position, or flip it from 'RUN'->'STOP'->'RUN'

The DCS should now assume control of the KD11-K CPU, irregardless of the previous state of the CPU ("CONSOLE", "RUN", or whatever).

NOTE

If the DCS 'RUN/STOP' switch was already in the 'RUN' position, then it is expected that the operator's console keys have no effect - the DCS is already enabled to execute, and is controlling the CPU. The DCS microcode does not monitor the console keypad for operator input.

If neither of the above methods produce a 'RUNNING' indication of the DCS (as per Section 5.1), then proceed to the next paragraph.

4.3 Using Console "INIT"

Set the DCS switches as detailed in Section 4.2 (ie, 'NORMAL' and 'RUN'). Now generate a processor "INIT" signal by simultaneously depressing the "HALT" and "START" keys on the operator's console. This should now initialize the CPU logic and restart the DCS, producing a 'RUNNING' indication (see Section 5.1).

If no 'RUNNING' indication is now present, then a problem exists in the power supply, system clocks, DCS module, or ???.

5.0 OPERATING PROCEDURE

5.1 DCS Indications While Executing

5.1.1 On the KD11-K Operator's Console

While the DCS is executing, the operator's console display should be approximately as follows:

"RUN" - ON Continuously
"PROC" - BLINKing
"USER" - BLINKing
"CONSOLE" - OFF Continuously
"BATTERY" - <indeterminate>

The 6 digit octal display should read:

(0 0 0 0 0 0) - In DCS pass 1;

(.0.0.0.0.0.0.) - In DCS passes 2-64.

If any of the above conditions are not met, then either:

1. The console hardware is inoperable, or

2. The DCS is hung in an 'error/scope loop'

(Not necessarily in that order)

5.1.2 On the DCS Module

On the DCS module, the two important indicators to watch at this time are the 'ERROR' and 'EOP' LEDs:

-----LEDS-----			
'ERROR'	'EOP'	State	Comment
OFF	OFF	Probable ERROR	DCS and/or processor HUNG
OFF	ON	* EOP *	Successful EOP
ON	OFF	* ERROR *	Genuine ERROR
ON	ON	Probable ERROR	DCS 'VERIFY MODE' indication

Note the 'Probable ---' notation. Either 'EOP' or 'ERROR' on the DCS should be lit; both OFF or both ON indicate non-standard conditions that require further investigation. Neither 'ERROR' nor 'EOP' indicates that the DCS and/or processor are in a HUNG condition (eg, clocks suppressed); see section 8 (Error Handling). Both 'ERROR' and 'EOP' is the standard DCS 'VERIFY MODE' indication. Make sure the DCS 'NORMAL/VERIFY' switch is set to 'NORMAL' (if this is intended), and then restart the DCS. See section 10 (Verify Mode) for further information.

5.2 DCS Execution Time

The execution time of the DCS will vary depending upon its mode of initiation.

5.2.1 "CNTRL/DIAG" Start From Operator's Console

When started via "CNTRL/DIAG" from the operator's console, and assuming no errors, DCS will execute with console display as detailed above for approximately 6 seconds (64. passes at 100 milliseconds/pass, about 350,000 microcycles/pass). After this time has elapsed, control should return to the KD11-K microcode, as if the CPU had just been powered up with the slide switch set to "HALT". Note that the machine will end up in console mode irregardless of the actual slide switch settings (RUN/BOOT/HALT). Section 5.3 interprets the console and DCS displays after this delay time, for successful 'END-OF-PASS' and 'ERROR' indications.

5.2.2 Start From DCS Switches

When started via setting 'RUN/STOP' to 'RUN' on the DCS module, the DCS will execute continuously, not returning to the KD11-K microcode until the 'RUN/STOP' switch is reset to the 'STOP' position. Throughout this time, the execution indications will be as previously detailed. When the switch is returned to 'STOP', the DCS may execute for a maximum of

approximately 6 more seconds; and then proceed as described directly above, entering "CONSOLE" mode.

5.3 DCS 'END-OF-PASS' and 'ERROR' Indications

5.3.1 On the DCS Module

Successful 'END-OF-PASS' and 'ERROR' conditions detected by the DCS are indicated most directly on the DCS module by the two LEDs labeled 'EOP' and 'ERROR'. 'EOP' is turned on at the end of each successful pass thru the DCS code - assuming 'ERROR' has not yet been turned on. In an error-free running situation, 'EOP' will be on; 'ERROR' off. Note that 'EOP' comes on at the end of the first pass (assuming no errors), and thus will appear to be on continuously.

In the event the DCS detects an error, the 'ERROR' LED will be turned on, and the 'EOP' LED turned off. The 'ERROR' LED will be latched in such a way that it cannot be turned off if the error disappears (i.e., a 'flaky' timing error), thus retaining the indication of an error. See section 8 (Error Handling) for a full treatment of the various error conditions: Detection, Indication, Scope-looping.

5.3.2 On the Operator's Console

Successful 'END-OF-PASS' and 'ERROR' conditions are indicated on the operator's console as follows:

1. A successful 64. passes and return to KD11-K processor microcode control, by a

   (.1,2,3,3,2,1.)

   in the console display, and the KD11-K processor "HALTed" in "CONSOLE" mode.

2. An error in pass 1 will usually (but not always) be indicated by a

   (0 0 0 0 0 0)

   in the console display.

3. An error in pass 2 thru -- will usually (but not always) be indicated by a

   (.0,0,0,0,0,0.)

   in the console display.

Note the qualifications in the previous statements:

For a successful 'END-OF-PASS' and exit back to KD11-K microcode control, only one display is valid:

(.1.2.3.3.2.1.)

For any other display except "(.1.2.3.3.2.1.)", check the DCS module 'EOP' and 'ERROR' LEDs for the most reliable indication of the result of execution.

See Section 8 for a full explanation of error processing.

6.0 RESTRICTIONS

6.1 Hardware

6.1.1 Cache and Memory Management (KT)

The DCS executes with both Cache and KT disabled:

1. Cache, by setting both the "Force Miss" bits
2. KT, by clearing the "Enable" bit

The DCS checks the most basic path from the UNIBUS to/from internal data paths. Further macro diagnostic programs are available for Cache (MD-11-DQKKA-*) and Memory Management (MD-11-DQKTA-*) fault diagnosis.

6.1.2 MOS Memory Battery Backup

The MOS memory battery backup (if present) must either be "Good", or else disabled. Otherwise, the micro diagnostic code WILL NOT execute without detecting an error.

Indications are: ERCD=5621/TNUA=7400, in TEST620C.

6.2 Software

6.2.1 Return To Console / DCS End-of-pass Processing

Console "HALT" mode is the ONLY exit provided from DCS back to base machine microcontrol. This action is due to:

1) The DCS, upon detecting an error, locks itself (and the processor) up in such a way that manual intervention by the operator is required for return to base machine microcontrol.

2) The DCS completely alters the internal microstate of the processor, destroying its previous contents, and leaving "garbage" in its place. Thus the full base machine power-up "INIT" sequence is generated by DCS to "clean-up" the processor prior to returning control.

7.0 TEST DESCRIPTION

7.1 Test Structure

7.1.1 Philosophy

The testing philosophy used in the design of the DCS microcode centers around two major points:

1. Start with as minimal a HARDCORE as possible.

2. Use only TESTED and VERIFIED logic elements to diagnose UNTESTED elements; add TESTED elements to the arsenal of logic available for further testing.

This method of test organization and construction presents the best approach for building a high resolution and high coverage diagnostic program. Section 7.4 (below) summarizes the actual testing order present in the DCS microcode, as designed using the above philosophy. Section 8.1 presents more specific details on the actual method of ERROP detection employed by the DCS microcode and hardware.

7.1.2 Mechanics

The manner in which DCS tests are setup is depicted in the following diagram:

```
          \ /
           .
  <setup ENUA/DCS-CNTR as test requires>
           .
           .
  <perform the actual test function>
      -may be inline microcode
      -may call subroutine
           .
           .
  <jump to BUT area, perform BUT(test) into TARGET>
           .
           .
  <enter TARGET area from BUT, ENUA::TNUA compare>
      -the DCS-CNTR was loaded above to enable compare here
      -the ENUA was setup at correct TARGET point
      -exit via BUTA(RETURN)
           .
          \ /
  <...next test now begins...>
```

7.2 DCS Microcode Conventions

7.2.1 Microdiagnostic Code Listing Organization

The format of the DCS microcode listing is presented in the following table, along with a brief description of each section (when necessary):

pages	contents
1-5	IDENTIFICATION, TABLE-OF-CONTENTS
6-9	DCS REGISTER LAYOUTS, MICROWORD BIT TABLE a number of graphic tables
10-24	MICROWORD FIELD DEFINITIONS, BASE MACHINE defines U<47:00> and their functions
25-26	DCS FIELD DEFINITIONS defines U<54:48> and their functions
27-37	"SIMPLE" MACROS these macros are combinations of FIELD definitions only
38-52	"ADVANCED" MACROS these macros are combinations of SIMPLE macros
53-380	---DCS-MICRODIAGNOSTIC-TESTS---
381-384	EOP/VERIFY-MODE MICROCODE microcode used in END-OF-PASS and VERIFY-MODE processing
385-393	GENERAL SUBROUTINES, COMMON CODE this section contains some VERY commonly used subroutines
394-395	JAMUPP MICROCODE all JAMUPP conditions enter here ...
396-402	BUT(---) TAKEOFF MICROWORDS most all DCS tests start their microbranch from this list of BUT's
403-417	BUT(---) TARGET MICROWORDS and most all DCS tests end up here, where they compare their ENUA:ITNUA in this 256. word table
418	BIT MAP OF DCS ADDRESS SPACE

a BIT MAP of the entire DCS address space, 1=USED, 0=FREE

X1-X10 SYMBOL/LINE-NUMBER/LOCATION CROSS REFERENCE
a very useful reference to find the location in the DCS listing of a particular symbolic label

X11-X18 LOCATION/SYMBOL CROSS REFERENCE
an expansion of the above BIT MAP, substituting the symbolic label for the 1/0's present in the above

X19 FREE/USED LOCATION SUMMARY
a quantitative summary of FREE/USED microlocations, by PAGE and TOTAL

7.3 What is/is-not tested

The DCS micro diagnostic code has been designed to detect and isolate errors within the "internals" of the KD11-K processor hardware. As such, it does not attempt (nor is it possible) to detect certain errors, which are processor related, but require devices external to the processor. Errors related to NPR/BR arbitration sequences, multiple BR priority level interrupt sequences, POWER FAIL / RESTART, etc., are a few of the elements of this general class. The following sections enumerate both classes of logic: TESTED by DCS, and UNTESTABLE by DCS.

7.3.1 TESTED by DCS

Essentially, the DCS is designed to test the "heart" of the KD11-K processor - those elements that must be functional to "bring up" the processor to a level whereupon further macro diagnostic programs could then be loaded and started (ie, M9301-YH boot/diagnostic, processor and peripheral diagnostics, etc.) to successfully isolate more complex processor and system related errors. Processor operation with either KT/KJ or CACHE enabled was not considered as part of this "heart" - as the processor will run perfectly fine without either of these facilities (albeit at a degraded performance level). Thus the DCS is able to concentrate on more thorough coverage and resolution in those portions of the hardware that are least "visible and testable" from a macro diagnostic, and very suitable to micro diagnosis.

Rather than present a full module by module list of which logic the DCS tests, refer to section 7.4 (below), which is an itemized "execution-time" summary of the DCS micro diagnostic tests. The following section lists those areas that the DCS cannot test (ie, uncontrollable or unobservable logic) or

would have required a prohibitive number of microwords to test effectively.

7.3.2 UNTESTABLE by DCS

From a functional point of view, the DCS operates on the KD11-K processor from the "inside out". Both the CACHE and KT/KJ are turned off (disabled); there is no FP11-E HFP unit assumed to be present; likewise no ECS/WCS options. No external I/O options are assumed on the UNIBUS, except the standard DL11-W console interface / line clock. The lowest 4K memory bank is assumed to be present. These restrictions impose constraints on the logic that can be exercised by DCS. The following list attempts to detail as specifically as possible, on a module by module basis, those functional areas of the processor that the DCS cannot fully diagnose:

-K2 UWORD-
- -full effects of processor "INIT"
- -NUA<11> (DCS can't "see" it)
- -CROM contents and address drivers
- -FP11-E related (FLPGO, HFP(CC), etc.)
- -UCON's HFP/KT/WCS
- -HFP FLAG-ROM contents

-K3 IR-DECODE-
- -CROM contents and address drivers
- -CROM extension roms
- -full effects of processor "INIT"
- -full rom contents (location by location) of:
  - -BYTE/CC, INSTR-5, FLTPT, CC(V/C), PS(CC)-BRANCH,
  - -BYTE-DFCODE, KT-DECODE roms
- -KT/KJ enable/select logic

-K4 DATAPATH-
- -full processor "INIT" effects
- -data i/O validity in some A/B/C-SPAD locations
- -SP/BY/KT selection
- -CACHE INVALIDATE logic

-K5 CACHE/KT-
- -processor UNIBUS operation with CACHE enabled
- -processor UNIBUS operation with KT/KJ enabled
- -status conditions of above, including
  - -RED/YELLOW ZONE, MM-ABORT, CACHE errors, etc.

-K6 TIMING-
- -ECL clock logic (must be "clocking" ...)
- -74S37 etc CLOCK DRIVERS - mostly HARDCORE
- -MAINTENANCE clock
- -some JAMUPP/PULSE-SUPPRESS conditions (see K7)
- -UNIBUS master arbitration, NPP/BA/PROCESSOR
- -CACHE control (NPR track, HIT, etc.)
  - -remember DCS turns CACHE O

-UNIBUS address drivers - full test of same
-full check of INTRNL-ADDR rom contents
-full processor "INIT" effects

-K7 STATUS-
-full processor "INIT" generation/effects
-POWER FAIL/RESTART logic
-BR-4/5/7 requests, BG arbitration
-SACK timeout (BG/NPR)
-NPR/NPG logic
-SERVICE conditions:
-FP11-E/YELLOW-ZONE/CONSOLE/PWRFAIL/CACHE
-JAMUPP conditions:
-KT-ABORT/RED-ZONE/CACHE+MEMORY+WCS-parity-error
-CONSOLE interface:
-KEYCODE input/DISPLAY output/LOCAL and REMOTE
-[see 11/60 microcode listing for "CONSOLE MICRO TEST", which tests these functions]
-STATUS mux bits:
-bits related to above conditions not asserted
-BATTERY BACKUP for MOS MEMORY OK

7.4 Global Test Order

The following list provides a summary of the major functional blocks of the DCS micro diagnostic code, presented in execution-time order. Note the progression from the innermost portions of the processor logic (microsequencing, IR decode); through the intermediate areas (ALU, SHIFT TREE, etc.); out to the external interface logic (UNIBUS cycles, INTR sequences).

NOTE

Most of the capitalized terms refer to specific hardware elements in the KD11-K processor. No attempt is made to explain their meaning - the unfamiliarized reader is referred to the processor logic block diagrams, print set, maintenance manual, and the micro diagnostic code listing for their definition.

The notation "[xxx/yyy]" signifies the micro tests in the range "TESTxxx" to "TESTyyy". All DCS tests are numbered octally, and are executed in ascending order. Some test numbers are further broken down into "TESTxxxA", "TESTxxxB", etc., when their functions are logically similar. Certain test numbers are non-existent (eg, there are no tests with numbers in the range 200-277).

[001/007] - NUA sequencing logic
-UPP sequencing, page changing

[010/011] - microsubroutine operation
  -RETURN register, BUTA(SUBR-B)/BUTA(RETURN) decode

[012A/050B] - IR decode logic and microbranch
  -INSTR =1, =5, -FLTPT decode
  -microbranch selection / execution
  -misc IR decode related microbranch logic
  -processor UCON IR load, EMIT constant generation

[101] - D-REGISTER/DBUF/BUSDIN/IR datapath
  -D-REGISTER load, ALU control (zeroes)
  -DBUF load, BUSDIN enable (UCON)

[102A/104B] - C-SCRATCHPAD
  -address modes (2-bit/4-bit), address lines
  -BUSDIN/CSP/ALU-B/D-REGISTER/DBUF/IR datapath, [1s/0s]

[105A/105E] - SR load/store
  -SR load/store/XMUX-enable
  -BUT(SR<3:0>) microbranch
  -SR/ALU-A/D-REGISTER/DBUF/IR datapath, [1s/0s]

[114A/122A4] - ALU logic functions, D[C] sources
  -ALU function/mode decode
  -ALU logic function execution
  -D[C] sources (ALUxx, CIN, save); D[C] microbranch

[130A1/136B2] - ALU arithmetic functions
  -ALU arithmetic function/mode decode
  -ALU arithmetic function execution
  -ALU carry logic (in, out, lookahead)
  -D[C] sources (COUTxx, CIN)

[320A/320F] - D[C] select logic, D=ZERO logic
  -D[C] 1/8 addressing
  -BUT(D=ZERO) decode logic

[350A/352D] - A/B-SCRATCHPADS
  -addressing, lines and mode (SF, DF, RIF)
  -data patterns [1s/0s]

[361A/371B] - SR/GUARD/XMUX, RES control
  -SR shift (left, right, hop)
  -GUARD register (shift, enable/disable, test)
  -RES/SR control

  -FLTPT assemble port

[372A/372B] - CUA(PROC mux) / BUTA(SUBR-A)
  -BUTA(SUBR-A) decode/execution, RETURN register

[373A/373B] - JAMUPP and BUTA(DIAGNOSE)
  -active BUTA(DIAGNOSE) decode
  -JAMUPP clock suppress logic via external JAM

[374A1/376A] - A/B-SCRATCHPADS
  -rewrite modes test (A/B, HI/LO, etc.)
  -BYTE WRITE, DAD control
  -R-IOR-1 / FLTPT-INHIBIT addressing

[410A/410E] - BYTE / BYTE-CONSTANT / D=ZERO (loop)
  -BYTE/WORD rom decode, microbranch
  -BYTE-CONSTANT CSP addressing, DAD control
  -BUTR(D=ZERO) decode (full test)

[500C/500F] - PREFETCH / OVERLAP / SP-DEFEAT (loop)
  -PREFETCH rom decode, microbranch
  -OVERLAP rom decode
  -SP-WRITE-DEFEAT decode/control

[503A/510F] - processor UCON registers / control
  -FLAGS/EXFLAGS - read/write/microbranch
  -FPS - read/write/microbranch
  -PS - read/write/microbranch
  -MULTIPLE BUT - input/select/output
  -SERVICE/INTR decode logic, microbranch

[511A1/511B4] - MOVE FROM SAME STACK (MFSS) logic
  -decode / microbranch

[512A1/512E2] - KT SRC/DST addressing logic for ASP/BSP
  -rom decode / control

[520A/520E] - INSTR BRANCH rom
  -IR/PS inputs, microbranch output

[533A/537] - SHIFT TREE (AMUX/BMUX/CMUX/SENDMUX)
  -data path (1s/0s)
  -function decode / mux select
  -RES control / select
  -COUNTER load / read

[551A/551C] - base machine COUNTER
  -active BUTA(COUNT) decode/microbranch
  -COUNTER count execution

[610A1/610D2] - PS condition code NZVC generation
  -INSTR CLASS decode
  -BYTE/WORD CC mux select
  -CC rom addressing/data

[620A/624D] - microbreak and JAMUPP
  -MICROBREAK REGISTER load/enable/compare
  -JAMUPP via microbreak, JAM register & STATUS
  -CUA TRACKING, lock/unlock
  -JAMUPP inhibit micro-operation logic
  -JAMUPP CLEAR

[701A/701D] - BA register

-19, BIT load/read via STATUS mux
-microbranch conditions

[710A/722C] - UNIBUS function decode, error conditions
-ODD ADDR/INTERNAL ADDR/SSYN TIMEOUT errors
-18,/16, bit BA modes, I/O PAGE decode, CONSOLE mode
-SERVICE/JAM register inputs (STATUS mux)
-bus function decode (DATI, DATO, etc.)

[730A/731E] - UNIBUS cycles to/from memory
-DATI(B)(P)/DATO(B) execution, side effects
-UNIBUS data lines, control lines C0, C1
-ALLOW ODD ADDR, BYTE/WORD operations
-DBUF, UNIBUS data latches load/enable
-DATI-CLKIR decode/execution
-DATIP/PROC-BBSY/BUTA(LAST) logic
-clock suppress / restart logic

[740A/740D] - UNIBUS cycle function modification
-BUTA(INSTR-1)/PREFETCH alteration of BUS CODE
-bus cycle YANK (SP DEFEAT) decode

[761A/763D] - UNIBUS interrupt (BP INTR) logic
-bus reset, microbranch on status
-line clock INTR enable, at BR6
-PS PRIORITY level/INTR PRIORITY level interaction
-SERVICE port conditions
-ALLOW BUS GRANT / VECTOR LOAD logic

8.0 ERROR HANDLING

8.1 What is an 'ERROR' ?

The concept of an 'ERROR' in DCS terms is very simple. It involves the use of the ENUA (Expected NUA), TNUA (Tracking NUA), and DCS COUNTER registers; all of which are local to the DCS module. The 11/60 processor itself has no control over the setting/clearing of 'ERROR'; in fact, it cannot directly determine whether 'ERROR' is set or clear.

The ENUA register (12 bits) is loaded from the EMIT field of the microword, under control of a DCS rom extension bit. It is setup at the beginning of a test to reflect the "EXPECTED" micro address after the test microbranch ("BUT") is executed.

The TNUA register (12 bits) is loaded continuously as the DCS microcode executes, TRACKING the progress of the microaddress field. This register contains the value of the "RECEIVED" micro address after the test microbranch is executed.

The DCS-CNTR is loaded with a value from (00)-(17) (octal), from the EMIT field of the microword, under DCS rom extension control. This register continuously counts up every microcycle. When the contents of this register is (17), the DCS hardware compares ENUA and TNUA, and does the following:

Set 'ERROR'="1" if DCS-CNTR=(17) and ENUA<>TNUA
else leave 'ERROR' unchanged from its previous value.

This is the manner by which DCS is able to set 'ERROR'. All DCS tests use this method.

Note also that the DCS hardware "locks up" the loading of the ENUA and TNUA registers after 'ERROR' is set, preserving their contents. Thus only the FIRST 'ERROR' will be recorded. There is no provision to detect subsequent errors until the previous ones are eliminated. See the DCS Maintenance Manual and Print Set for more detailed information.

8.2 FAULT DIRECTORY Format and Use

8.2.1 Basic Structure

The FAULT DIRECTORY is essentially a tabular summary of all ERROR codes the DCS is able to generate - a total of 432 entries occupying 52 pages. Each individual ERROR code entry in the FAULT DIRECTORY contains a short description of the test, and the module replacement information pertaining to that test. For ease of reference, the ERROR codes have been organized into ascending numerical order, in the range 4000(8)-6777(8).

8.2.2 Basic Use - with an example

This section describes how to use the FAULT DIRECTORY after the DCS has been run, and has indicated an error is present in the KD11-K processor.

Assume for the purpose of explanation that the DCS was started, and has returned the following values:

ERCD = 4616 (ERror CoDe)
TNUA = 7405 (Tracking NUA)

with EOP=<OFF>, and ERROR=<ON>

1. Going to the FAULT DIRECTORY, we find the entry for ERROR code 4616 to be on page 9, entry number 73.

2. Some general information about the failing test is first obtained:

   a) 'Symbolic label' - A reference to the DCS microtest which failed, in this instance TEST-115-A2.

   b) 'Line number' - A reference to the line number in the DCS microcode listing where the failing test is located (here, line number 5983).

   c) 'ENUA' - The Expected NUA of this test, in this case 7412. Note that the obtained TNUA (7405) is not the same as the test's ENUA (7412); thus the ERROR.

   d) The remainder of the line contains a short description of the function performed by this test; in this instance we note the test was diagnosing the ALU portion of the DATA-PATH module.

3. We now note that the TNUA we obtained was 7405. Scanning downward in the column of TNUA entries for this test, we find it listed as the fourth entry. More information, specific to this particular error, can now be obtained:

   a) 'Module sequence' - These 3+ columns contain (scanning left to right) the top 3+ choices of processor modules to inspect/replace, in order to locate and correct the fault(s). The module choices are listed using "slot"

      notation (ie, #s, where s=the slot), and a "confidence factor" to indicate the percent confidence that replacing this particular module will eliminate the fault(s). The best choice is the module called out in the first column ("#1"); then "#2" etc. Note that the percentages are rounded to the nearest 5%, and may therefore not always add up to exactly 100%.

   b) To the right of the module choices is summarized the

IC information obtained from the FAULT INSERTION effort of the DCS/KD11-K (signified by "FI"). IC information is referenced to a particular module by the notation:

K4=E23,E33-36,E89-E90,E101: K2=E12,E15,E69;

- -
- N O T E -
CALLOUT OF SPECIFIC IC'S ON A MODULE IS --NOT-- INTENDED INTENDED TO BE AN "EXHAUSTIVE-ONLY-THESE-ARE-THE-ONES" LIST. IT IS INTENDED TO PROVIDE REFERENCE TO A SPECIFIC FUNCTIONAL AREA OF A MODULE, AND GIVE REFERENCE TO THOSE IC'S WHICH CAUSED THE FAILURE DURING THE FAULT INSERTION EFFORT OF HARD STUCK-HIGH/-LOW AND ADJACENT-PIN-SHORT TYPE FAULTS. AGAIN, DO NOT ASSUME THIS LIST TO BE ALL INCLUSIVE OF THE POSSIBLE CHOICES FOR FAULTY IC'S.
- -

Another type of entry is of the format:

K404=ALU/CARRY-LOOKAHEAD;
or Kmppm=functional-description-of-logic-block

which references a particular module (#m) and page (#pp) in the KD11-K Processor Print set. This notation is used when specific fault insertion data is not available for a test.

4. In the instance when the TNUA obtained does not match any of those provided under a given test/ERROR code entry, a wild-card character ("?") has been used to allow a match with any octal digit. Thus 740? matches 7400, 7401, ..., 7407. These entries should be used for further information or when a specific TNUA is not present.

5. If there is no TNUA listed which matches the obtained TNUA, and also no wildcard entry is present; then the information about the functional nature of the test (from above), along with an intelligent interpretation of the obtained TNUA, will be required. The following table lists some TNUA's that might be obtained in such a case:

(see table on following page)

TNUA	Cause
4000	DCS forced to its starting address
4777\ 4756 > 4747/	an unexpected JAMUPP condition occurred
7361	in UNIBUS function tests, a JAMUPP did not occur when expected
7400- 7777	the "standard" BUT() target area for DCS micro-tests

8.3 'SCOPE Loop' Facility

8.3.1 General Information

The 'SCOPE loop' implementation provided by DCS is almost identical to that provided in the standard MAINDEC macro diagnostic program. What the 'SCOPE loop' does is to repeatedly execute the same sequence of diagnostic test code; this allows the technician to 'scope' appropriate logic signals in an effort to zero-in on the fault.

The DCS 'SCOPE loop' occurs ONLY and ALWAYS when 'ERROR' is set. There are no user options to change the size or range of the loop - all these parameters have been fixed in microcode and hardware. The loops have been setup to be as tight, and as useful, as was possible. Most are in the range of 10-30 microwords, although some (three, in particular) are larger.

8.3.2 Implementation and Use

A DCS extension rom control bit is used to enable the 'SCOPE loop' check at selected points in DCS code execution. These points are recognized by the following:

```
SCOPE123:
     <possible some other functions>
     NEXT,   BUTD[SCOPE],   !NO ERROR:  "TEST124" [+1, WORDS]
             J/TEST124      !   ERROR:  "LOOP123" [-5, WORDS]
```

The two comments "ERROR/NO-ERROR" tell the user where the DCS code will branch, depending upon the current state of 'ERROR'. Usually, the 'NO-ERROR' condition falls thru to the next word (eg, +1, words). For the 'ERROR' case, the loop is ALWAYS backwards (ie, up the page, toward the point where the error was detected). The "-number" notation gives a relative count of the number (approximately) of micro words backwards in the dump.

This facility can be used in two modes - dynamic and static. Either mode must be entered via the use of the DCS 'RUN/STOP' switch set to the 'RUN' position, as this then enables the DCS code to execute continuously. The results are generally undefined if the switch is set to 'STOP', and the "DIAG" button was used to enter the loop.

Dynamic mode requires the use of an oscilloscope, logic analyzer, etc., and the determination of an appropriate logic signal on which to sync. The DCS microcode then automatically remains in this tight loop to allow observation of the suspected faulty signals, at processor cycle speed.

Static mode is entered in the same manner as dynamic; but afterwards the "SINGLE-MICROSTEP/MAINTENANCE-CLOCK" feature of the 11/60 processor (on K6 TIMING module, the two switches - see prints) is enabled. This allows the processor to be single micro - stepped, under user control. The additional debug features of the DCS can now be employed: the BUSDIN/DOUT display LEDs (16), and the (2) "free" LEDs. See the DCS Maintenance Manual for further details. Also available are the NUA (Next-U-Address) LEDs on the 11/60 processor "UWORD" module (K2). Note that these 'point' at the NEXT microword to be executed, not the current.

9.0 REVISION HISTORY

9.1 Revision Number

After a successful 'END-OF-PASS' indication, console internal exam functions can be used to obtain the 'DCS Microcode Revision Number', stored in the macro machine general register R5. BIT<15> of this number will also be set, indicating successful 'END-OF-PASS' was reached. The initial version of the microcode will display:

(100101)

with subsequent versions to be:

(100102)
(100103) etc

If, however, an 'ERROR' is obtained, one might still be able to obtain the DCS microcode revision number. In this instance BIT<15> will/should be clear, and the lower bits the revision code:

(000101)
(000102) etc

However, this number must be taken with caution, as the error may or may not have influenced the storing of the revision number in the register.

9.2 Revisions To DCS Code

Note that ALL revisions to the DCS microcode / FAULT DIRECTORY are to be documented in this section, with the following information supplied:

a) A short description of the problem(s) found, and how they were corrected.

b) Updated MAINDEC (MD) and DCS (uCODE) revision information.

c) Date of fix, and person responsible for fix.

d) The test/ERROR codes affected by the changes.

Note that the changes MUST also be incorporated into the DCS microcode listing, and/or the FAULT DIRECTORY listing at the appropriate points. Actual microcode changes will be entered as ECO's to individual ROM patterns on the DCS module (M7871). Contact PDP-11/60 Support Engineering for the procedure.

REVISION MO/uCODE	Date	Who	Explanation
A/101	18-Jan-77	DNN	Initial Release
B/101	22-Jun-77	DNN	No microcode changes; documentation added/updated.

10.0 DCS VERIFICATION / SELF-TEST

'VERIFY MODE' is a self-check mode designed to verify the operation of the DCS module and its associated error detection/indication support logic.

10.1 Requirements

This mode of operation requires that a known good PDP-11/60 system (as described in section 2) be used to test/verify a DCS module, so that errors detected by the DCS are due to the DCS module under test, and not due to the other system components. The set of PDP-11/60 processor macro diagnostics, or a known good DCS module, can be used to perform such a verification of the host system.

10.2 Verification Method

The method (or algorithm) used to perform the DCS self-verification is as follows:

Hardware on the DCS module is conditioned to execute a single pass thru the DCS microcode, via setting the DCS 'NORMAL/VERIFY' switch to the 'VERIFY' position. This also alters the 12 bit hardware counter on the DCS module from a 'Pass Counter' to the 'Verify Counter'.

At the start of a 'Verify Pass', this counter is preset to a specific value; predetermined so that when 'END-OF-PASS' is signaled by the DCS microcode, this counter will have a value of octal (7777), or be at the point of overflow (carry out) enabled.

As the DCS executes in 'VERIFY MODE', this counter is incremented whenever:

1. A microword is executed from page 7, or
2. A microword is executed with the 'VERIFY' bit (a page 4-6 only DCS ROM extension bit) asserted. These 'VERIFY' bits have been scattered, more or less at random, throughout the DCS microcode. Thus this counter will be incremented at random intervals during a 'Verify Pass'.

The DCS code executes approximately 350,000 microwords per pass; thus the counter will overflow between 2-85 times (depending upon the number of 'VERIFY' bits and page 7 references encountered) before the 'END-OF-PASS' / 'Verify Counter' overflow match. Physically, the verify count is retained modulo 4096 (12 bits), with only the low order bits of the count used in the comparison.

A verification will be considered successful only if a verify counter overflow point exactly matches the microword which signals 'END-OF-PASS' (done only once) in the DCS microcode.

10.3 Procedure And Indications

10.3.1 Procedure

To run the DCS in verify mode, the following procedure is followed:

1. Install the DCS in the PDP-11/60 as detailed in Section 3

2. Set the DCS switches:

   'RUN/STOP' = 'STOP' and
   'NORMAL/VERIFY' = 'VERIFY'

3. Now set:
   'RUN/STOP' = 'RUN'

4. The DCS now executes a single 'Verify Pass'

5. At the end of the 'Verify Pass' the DCS enters a microcode loop, in which:

   - An error is forced with specific
     'ENUA', 'TNUA', and 'ERROR code' values
   - 'END-OF-PASS' is repeatedly signaled
   - A 'Scope Loop' branch is executed

   See the DCS microcode listing, under 'Verify Mode Code' for the exact sequence of operations.

6. At this time examine the DCS module LEDs for comparison with their expected contents, as noted below.

7. To return control to the PDP-11/60 after a 'Verify Pass', position:

   'PUN/STOP' = 'STOP' and
   'NORMAL/VERIFY' = 'NORMAL'

   And then generate a "CONSOLE INIT" ("START/HALT") on the operator's console

10.3.2 Indicators

Only the status described below is acceptable to signal a successful DCS verification. Assuming a known good PDP-11/60 system, any deviation from the description (below) should be considered an indication of a fault in the DCS module under

test.

After a 'Verify Pass', indications on the DCS module will be:

'TNUA' = (7522)
'ERROR' = (4255)
(Note the alternating ON/OFF pattern)

'ENUA' = (7523) was loaded to force
an 'ERROR' indication

'EOP' LED = ON, Approx. 1/2 brilliance
'ERROR' LED = ON, continuously

This will be the only instance when both
the 'EOP' and 'ERROR' LEDs should be on
simutaneously.

Indications on the PDP-11/60 console should be:

"RUN" LED = OFF continously
"PROC" LED = OFF continously
"USER" LED = OFF continously
"CONSOLE" LED = OFF/ dimly lit
"BATTERY" LED = <indeterminate>

Octal display = (212121), with the
decimal points either on or off.

11.0 MISCELLANEOUS

11.1 ACT/APT/XXDP

The DCS module is not directly supported by ACT/APT/XXDP software at this time.

11.2 Macro Instruction Interface

11.2.1 DCS Presence

Presence of a DCS module in slot 1 is indicated by a bit in the "WHAMI" register:

NOTE

See PDP-11/60 documentation for a full description of the "MED" instruction.

```
MED     ,022            ;READ WHAMI => R0
BIT     #BIT08,R0       ;BIT<08>=1, DCS PRESENT
                        ;BIT<08>=0, NO DCS IN SLOT-1
```

11.2.2 DCS Register Access

Access to several of the internal registers and status bits is also possible via the "MED" instruction:

```
MED     ,152            ;READ DCS 'TNUA'
```

After execution, R0's contents is as follows:

```
0000_$TNUA<11:00>
```

Similarly:

```
MED     ,153            ;READ DCS 'EOP/ERROR'
```

and R0's contents:

```
ERROR#01#EOP#ERRCOD<11:00>
```

BIT<15> = ERROR(1)H
BIT<14:13> = "01", code for DCS module
BIT<12> = EOP(1)H

11.2.3 Macro Instruction Startup Of DCS

The DCS may also be started via a "MED" instruction. Note, however, that this is a one-way tranfer of control; there is

no means to re-enter an executing macroinstruction program without operator intervention at the operator's console.

This method simulates the operator depression of the "CNTRL/DIAG" keys via loading the KD11-K microaddress pointer (MUA) with the starting address of the "DIAGNOSE" key service routine.

```
        MOV     #011410,R0      ;(1141) = "DIAGNOSE"
        MED     ,347            ;WRITE MUA
        ;NEVER COME BACK TO HERE
        ;DCS ALWAYS EXITS TO "CONSOLE" MODE
```

###)	ERROR code	Symbolic label	Line number	ENUA	TNUA	→Module sequence→ #1/K6	#2/K6	#3/K6	Test summary + Print reference + Chip information
1)	4000	TEST001	2633	6252	NUA SEQUENCING, PAGE (4) => (2) (2==6); UBFs(36)				
					4000	K3/99	../..	../..	FI; K3=E6,E15
					4203	K2/99	../..	../..	FI; K2=E41,E47
					4282	K2/90	K3/5	K4/5	FI; K2=E13-E14,E34-E36,E40-E41,E46-E47,E58-E59,E62, E64-E65,E67-E68,E72-E74,E78-E80,E91; K3=E34, E111; K4=E3
					4283	K2/99	../..	../..	FI; K2=E14
					4286	K2/99	../..	../..	FI; K2=E34,E46,E58,E68,E74,E80
					4287	K2/99	../..	../..	FI; K2=E40,E65
					4747	K2/75	K3/25	../..	FI; K2=E2,E29,E52-E53,E79; K3=E6,E26
					4786	K2/99	../..	../..	FI; K2=E70
					4757	K3/99	../..	../..	FI; K3=E6,E7
					4776	K2/99	../..	../..	FI; K2=E60
					4777	K2/75	K3/10	K6/5	K4/5 K7/5 FI; K2=E1-E7,E10-E18,E21-E25,E27-E39, E41-E56,E57-E59,E61-E81,E84,E90,E93-E96,E100, E104; K3=E1-E20,E22,E24-E26,E32,E39,E41,E48, E101; K6=E40,E43,E60-E61,E66,E79-E81,E83,E92; K4=E18,E26,E29-E30,E38,E71,E87; K7=E36,E52,E74
					5130	K2/99	../..	../..	FI; K2=E94
					5247	K2/99	../..	../..	FI; K2=E47
					5276	K2/99	../..	../..	FI; K2=E6,E33,E60
					6000	K2/99	../..	../..	FI; K2=E7-E8,E19,E40
					6062	K2/55	K3/45	../..	FI; K2=E4,E7,E117; K3=E89,E91
					6201	K2/99	../..	../..	FI; K2=E70-E71
					6212	K3/55	K2/45	../..	FI; K3=E89,E91; K2=E2,E7
					6231	K2/99	../..	../..	FI; K2=E62
					6235	K2/99	../..	../..	FI; K2=E62
					6242	K3/55	K2/45	../..	FI; K3=E21,E81; K2=E2,E8
					6250	K3/55	K2/45	../..	FI; K3=E21,E81; K2=E2,E8
					6253	K2/55	K3/45	../..	FI; K2=E2,E60,E66,E69,E75,E81; K3=E21,E23,E44,E68
					6256	K2/55	K3/45	../..	FI; K2=E2,E8-E9,E21,E27,E60,E66,E69,E75,E81; K3=E21-E22,E37-E39,E54,E56,E74-E75
					6257	K3/90	K2/20	../..	FI; K3=E37-E38,E74; K2=E8
					6272	K2/95	K3/5	../..	FI; K2=E2-E3,E6-E7,E12,E15,E18,E24,E30,E36; K3=E52
					6277	K3/99	../..	../..	FI; K2=E2,E13
					6282	K2/80	K3/20	../..	FI; K2=E1,E3,E6-E7,E12,E15,E18,E24,E30,E36; K3=E51, E58,E78
					6372	K2/99	../..	../..	FI; K2=E7
					6682	K2/80	K3/20	../..	FI; K2=E1,E19,E42,E59,E65,E68,E74,E80; K3=E23,E63
					7097	K2/99	../..	../..	FI; K2=E117
					7231	K2/99	../..	../..	FI; K2=E14
					7262	K2/99	../..	../..	FI; K2=E35,E47,E58
					7782	K2/99	../..	../..	FI; K2=E68
					7777	K2/99	../..	../..	FI; K2=E68

Module codes: K1/DC8 K2/UNIBUS K3/INSTRUCTION DECODE K4/DATAPATH K5/KT CACHE K6/TIMING K7/STATUS

###)	ERROR code	Symbolic label	Line number	ENUA	TNUA	→Module sequence→ #1/K6	#2/K6	#3/K6	Test summary + Print reference + Chip information
2)	4755	VFY003	18313	7523	THIS ERROR-CODE EXPECTED IN DC8 "VERIFY MODE"				
					7522	../..	../..	../..	THIS TNUA EXPECTED IN DC8 "VERIFY MODE"
					7522	K6/80	K7/40	K3/5	K2/5 K4/5 FI; K6=E3-E8,E12,E14,E22,E26,E31-E32, E34,E37-E40,E42-E43,E46,E54,E56,E61-E62,E64,E69, E71-E72,E76,E78-E81,E86-E88,E90,E94-E96,E98-E102, E104-E105,E108; K7=E2,E4,E8,E12,E14-E15,E18-E23, E28-E37,E34-E35,E37-E40,E43,E45-E47,E49,E52,E54, E59,E62-E63,E66,E68,E70-E71,E97,E98,E97-E98; K3=E2,E6-E8,E10,E12,E18,E21-E26,E34,E52,E54,E64; K2=E2-E3,E6-E9,E21,E36,E50,E60,E66,E75,E81; K4=E6,E16
					????	K1/99	../..	../..	DC8 VERIFY MODE, BAD TNUA LATCHED
3)	4377	TEST007	2749	7303	NUA SEQUENCING, PAGE (4) => (3) (3==7); UBF=(35)				
					7301	K2/99	../..	../..	FI; K2=E61
					7303	K2/80	K4/35	K3/5	K6/5 FI; K2=E4,E10,E16,E22,E25-E26,E28,E31,E37, E43,E49,E52,E70-E71,E76-E77,E97; K4=E1,E12-E15, E29,E38,E70; K3=E13; K6=E34
					7307	K2/60	K3/40	../..	FI; K2=E9,E21,E27; K3=E21-E22
					7343	K2/95	../..	../..	FI; K2=E7
					7376	K2/99	../..	../..	FI; K2=E34
					7777	K2/99	../..	../..	FI; K2=E23,E29
4)	4450	TEST730C1	17711	7402	BUSA(LAST) CLEARS PROC BUSY (DATIP); UNIT=(0020529) ON BUSDIN				
					7400	K5/99	../..	../..	FI; K5=E94
					7401	K6/50	K7/35	K5/15	FI; K6=E4,E81,E70,E79,E103; K7=E11,E39,E46,E49; K5=E95,E97,E108
5)	4451	TEST740D	17653	7402	INSTR1=-OVERLAP, BUS CYCLE YANKED				
					4747	K5/65	K6/35	../..	FI; K5=E95; K6=E40
					7401	K6/99	../..	../..	FI; K6=E43
6)	4452	TEST740C	17619	7402	-INSTR1=PREFETCH, BC<0>=BC<0>				
					4747	K2/99	../..	../..	FI; K2=E41
					7401	K2/90	../..	../..	FI; K2=E105
7)	4453	TEST740B	17591	7402	INSTR1=PREFETCH, BC<0>="0"				
					7401	K2/65	K3/35	../..	FI; K2=E82,E105; K3=E66
8)	4454	TEST731F	17467	7432	BUS DATI=CLKIR, BA=(000000); PR=(000125) E98/(432) LOADED				
					4747	K1/99	../..	../..	FI; K1=E1
					7402	K6/99	../..	../..	FI; K6=E5
					7407	K6/99	../..	../..	FI; K6=E4,E106
					7412	K6/85	K3/15	../..	FI; K6=E3,E5; K3=P46

Module codes: K1/DC8 K2/UNIBUS K3/INSTRUCTION DECODE K4/DATAPATH K5/KT CACHE K6/TIMING K7/STATUS

###)	FRROR code	Symbolic label	Line number	ENUA	TNUA	->Module sequence-> #1/K6	#2/K6	#3/K6	Test summary - Print reference - Chip information
9)	4455	TEST731P	17428	7402	BUS DATI-CLKIR, BA=(000000); SERVICE=(000340)				
					4747	K5/99	../..	../..	FI; K5=E19
					7400	K5/99	../..	../..	FI; K6=E108
10)	4456	TEST731C	17481	7432	BUS DATI, BA=(000000); MEM(000000)=(000125) VIA DATO/DATOB/DATI				
					7400	K5/99	../..	../..	FI; K6=E93
					7402	K5/99	../..	../..	FI; K6=E36,E43,E85,F90,E93
					7405	K4/99	../..	../..	FI; K4=E28
					7420	K5/99	../..	../..	FI; K5=E94,E105
					7434	K6/88	K7/19	../..	FI; K6=E11,E55,E71; K7=E49
11)	4457	TEST731R	17363	7412	IR=(125200) E78/(412); NO BUS FUNCTIONS CLOCK-IR EXCEPT DATI-CLKIR				
					7400	K2/65	K3/35	../..	FI; K2=E34,E86; K3=E48
12)	4460	TEST730C	17188	7402	DATIP HOLDING BUS (PROC BBSY); DU=(125252) ON BUSDIN				
					7401	K6/88	K7/35	../..	FI; K6=E70,E79,E103; K7=E17,E62
13)	4461	TEST730D	17233	7402	BUS DATO, BA=(000000), DATIB=BYTE=ODD, BA=(000001); SERVICE=(000340)				
					4747	K5/50	K2/50	../..	FI; K6=E47; K2=E105
					7400	K5/90	K7/15	../..	FI; K6=E38-E39,E54; K7=E49
					7401	K5/99	../..	../..	FI; K6=E38
14)	4462	TEST730F	17286	7402	BUS DATO, BA=(000000), DATIB=BYTE=ODD, BA=(000001); DU=BUSDIN=(052525)				
					7400	K6/90	K3/10	K4/5	FI; K4=E8,E46,E66-E67,E72,E74-E75,E80,E83-E84,E86, E88,E90,E92-E93,E108; K5=E53,E63,E99-E101,E109, E113-E113,E118,E122; K4=E28
					7403	K6/99	../..	../..	FI; K6=E92-E93
15)	4464	TEST722B	17004	7402	INVALIDATE, ODD ADDR JAM; JAM=(101004)				
					????	K6/99	../..	../..	K605=UNIBUS-FUNCTION-DECODE
16)	4465	TEST722C	17029	7402	INVALIDATE, 16, BIT PBA, =I/O PAGE(6); SERVICE=(002340)				
					7400	K6/99	../..	../..	FI; K6=E89
					????	K6/99	../..	../..	K605=UNIBUS-FUNCTION-DECODE
17)	4466	TEST730R	17149	7402	BUS DATIP, BA=(000001); DU=(125252) AFTER ON BUSDIN				
					4747	K7/65	K6/35	../..	FI; K7=E16,E25,E32; K6=E85,E76,E68
					7400	K6/85	K3/5	../..	FI; K6=E66-E67,E74-E75,E83-E84,E92-E93; K5=E53,E63, E68
					7401	K6/65	K7/15	../..	FI; K6=E11,E19,E47,E55,E69,E66-E67,E69,E71,E74-E78, E79,E83-E84,E92-E93,E98-E99,E101,E103,E105; K7=E10,E17,F21,E32-E33,E49,E101
					7403	K6/99	../..	../..	FI; K6=E92-E93

Module codes: K1/DCS K2/UNORD K3/IRDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

###)	FRROR code	Symbolic label	Line number	ENUA	TNUA	->Module sequence-> #1/K6	#2/K6	#3/K6	Test summary - Print reference - Chip information
18)	4471	TEST730A	17086	7402	BUS DATO, BA=(000000), D=(125252); CHECK DBUF=D AFTER				
					4747	K6/65	K7/25	K5/10	FI; K6=E2-E3,E19-E20,E26,E40,E76,E85,E87-E88, E96-E98; K7=E2,E10,E32-E33,E36,E57-E58,F60; K5=E35,E42-E43,E62
					7400	K5/99	../..	../..	FI; K5=E90,E107-E108,E114,E120-E121
					7401	K5/65	K7/20	K4/10	K3/5 FI; K6=E11,E56,E64,E69,E72,E85,E101,E103; K7=E10,E33; K4=E28,E55; K3=E42
					7403	K6/50	K5/25	K2/25	FI; K6=E101; K5=E91; K2=E100
19)	4472	TEST132R2	7572	7417	BUTR(X=D[C]=XX), D[C]=COUT15=1?; A(0)+B(1)+CI(1)=D(0)+CO(1)				
					7413	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE
20)	4473	TEST136R2	8010	7402	(122645)+(132264)+(0)=(170360); CHECK COUT15=(0)				
					7403	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE
21)	4474	TEST00A	2730	4377	NUA SEQUENCING, NO 'BUT'				
					4777	K2/99	../..	../..	FI; K2=E17,E61
22)	4475	TEST136B1	7983	7434	(122645)+(132264)+(0)=(170360); CARRY LOOKAHEAD LOGIC				
					7400	K4/99	../..	../..	FI; K4=E47
					7417	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE
23)	4476	TEST136A2	7961	7403	(058132)+(132264)+(1)=(007417); CHECK COUT15=(1)				
					7266	K2/80	K3/20	../..	FI; K2=E8-E9,E21,F27; K3=E21
					7402	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE
24)	4477	TEST135B2	7904	7413	(058132)+(113226)+(0)=(170360); CHECK COUT15=(0)				
					7417	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE
25)	4500	TEST133A2	7830	7403	(103607)+(103607)+(1)=(007417); CHECK COUT15=(1)				
					7402	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE
26)	4501	TEST133B1	7852	7434	(074170)+(074170)+(0)=(170360); CARRY LOOKAHEAD LOGIC				
					7402	K4/50	K2/50	../..	FI; K4=E69; K2=E102
					7420	K4/99	../..	../..	FI; K4=E49
					7421	K3/99	../..	../..	FI; K3=E71
					7417	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE
27)	4502	TEST133R2	7879	7402	(074170)+(074170)+(0)=(170360); CHECK COUT15=(0)				
					7403	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE
28)	4503	TEST134A2	7742	7403	(045513)+(141701)+(1)=(007417); CHECK COUT15=(1)				
					7402	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE

Module codes: K1/DCS K2/UNORD K3/IRDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

###	ERROR code	Symbolic label	Line number	ENUA	TNUA	->Module sequence-> #1/##	#2/##	#3/##	Test summary = Print reference = Chip information
29)	4504	TEST134B1	7764	7434					(132264)+(036074)+(0)=(170360); CARRY LOOKAHEAD LOGIC
					7477	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE
30)	4505	TEST134B2	7791	7401					(132264)+(036074)+(0)=(170360); CHECK COUT15=(0)
					7403	K3/99	../..	../..	FI; K5=E95
					7407	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE
31)	4506	TEST135A2	7855	7417					(122645)+(064551)+(1)=(007417); CHECK COUT15=(1)
					7413	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE
32)	4507	TEST135B1	7877	7434					(055132)+(113226)+(0)=(170360); CARRY LOOKAHEAD LOGIC
					7477	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE
33)	4511	TEST134A1	7705	7434					(045513)+(141703)+(1)=(007417); CARRY LOOKAHEAD LOGIC
					7477	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE
34)	4513	TEST135A1	7818	7434					(122645)+(064551)+(1)=(007417); CARRY LOOKAHEAD LOGIC
					7477	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE
35)	4515	TEST136A1	7931	7434					(055132)+(132264)+(1)=(007417); CARRY LOOKAHEAD LOGIC
					7421	K4/75	K3/25	../..	FI; K4=E64; K3=E74
					7477	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE
36)	4516	TEST132A2	7527	7403					BUTR(D[C]=BA00), D[C]=COUT15=*1*; A(1)+B(0)+CI(1)=D(0)+CO(1)
					7402	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE
37)	4517	TEST132B1	7547	7434					ALU=DIVIDE=A-MINUS-B; A(052525), B(052525), D(000000), D[C]=*1*
					7400	K3/80	K4/20	../..	FI; K3=E61,E63,E61; K4=E16
					7402	K3/99	../..	../..	FI; K3=E51
					7477	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE
38)	4521	TEST132A1	7502	7434					ALU=A-MINUS=B; A(125252), B(125252), D(000000)
					7400	K4/70	K3/30	../..	FI; K4=E49; K3=E47,E92
					7402	K4/55	K3/45	../..	FI; K4=E6,E11,E17,E26,E96; K3=E41,E82
					7477	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE
39)	4523	TEST133A1	7600	7434					(103607)+(103607)+(1)=(007417); CARRY LOOKAHEAD LOGIC
					7400	K4/75	K3/25	../..	FI; K4=E8,E49,E71; K3=E53
					7402	K3/99	../..	../..	FI; K3=E41,E84
					7420	K4/99	../..	../..	FI; K4=E49,E88
					7477	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE
40)	4530	TEST130B1	7334	7434					ALU=A-PLUS-B-PLUS-1; A(125252), B(125252), D(052525)
					7400	K4/70	K3/15	K2/15	FI; K4=E49,E101-E102,E110; K3=E53; K2=E50
					7402	K4/80	K3/20	../..	FI; K4=E49,E98,E101; K3=E41

Module codes: K1/DCS K2/UWORD K3/IRDECODE K4/DATAPATH K5/RTCACHE K6/TIMING K7/STATUS

###	ERROR code	Symbolic label	Line number	ENUA	TNUA	->Module sequence-> #1/##	#2/##	#3/##	Test summary = Print reference = Chip information
	4530	TEST130B1	(Continued)						ALU=A-PLUS-B-PLUS-1; A(125252), B(125252), D(052525)
					7406	K4/99	../..	../..	FI; K4=E110
					7420	K4/99	../..	../..	FI; K4=E49,E102,E112
					7421	K4/99	../..	../..	FI; K4=E71
					7436	K4/99	../..	../..	FI; K4=E112
					7477	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE
41)	4532	TEST130B2	7365	7403					BUTR(COUNT=D(C)), D[C]=COUT15=*1*; A(1)+B(1)+CI(0)=D(0)+CO(1)
					7402	K2/80	K3/30	K4/25	FI; K2=E35,E41,E61,E77,E85; K3=E4,E6-E7,E11,E15, E43; K4=E61-E84
42)	4537	TEST320D	8197	7403					BUTR(COUT07=DOUT07=XX), TARGET=*01*
					7401	K3/88	K4/15	../..	FI; K3=E51,E55-E58,E95; K4=E10
					7407	K3/99	../..	../..	FI; K3=E74
43)	4541	TEST131A1	7392	7434					ALU=A-PLUS-B-PLUS-PS[C]; A(125252), B(052525), D(177777), PS[C]=(0)
					7400	K3/75	K4/25	../..	FI; K3=E35,E41,E66,E71,E84; K4=E49
					7402	K3/99	../..	../..	FI; K3=E82
					7477	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE
44)	4543	TEST320A	8098	7403					D[C] ADDR SELECT, CODE=*010*=ALU00=*1* ONLY ONE SET
					7402	K4/99	../..	../..	K404=D[C]=SELECT-LOGIC
45)	4544	TEST131A2	7423	7401					BUTR(D[C]=BA00), D[C]=COUT15=*0*; A(1)+B(0)+CI(0)=D(1)+CO(0)
					7403	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE
46)	4546	TEST131B1	7443	7434					ALU=DIVIDE=A-PLUS-B; A(052525), B(125252), D(177777), D[C]=*0*
					7400	K3/99	../..	../..	FI; K3=E82
					7402	K3/99	../..	../..	FI; K3=E61,E84
					7477	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE
47)	4550	TEST130A2	7314	7402					BUTR(COUNT=D(C)), D[C]=COUT15=*0*; A(0)+B(0)+CI(1)=D(1)+CO(0)
					7403	K4/99	../..	../..	FI; K4=E63-E64
48)	4552	TEST320B	8138	7400					BUT(D<14:00>=00D15), D=(000001), TARGET=*00*
					7402	K2/99	../..	../..	FI; K2=E40,E88,E94,E99,E112
49)	4553	TEST762A1	17899	NONE					DL11-W ENABLED FOR BR6 INTR; DID NOT RESPOND W/ RG-SERVICE-L WITHIN 22. MS.
					4553	K7/85	K6/10	K5/5	K3/9 FI; K7=E3,E5-E6,E11,E13,E16-E17,E19-E20, E25-E26,E28,E35-E36,E39,E76,E80; K6=E25,E34,E52; K5=E1; K3=E74
50)	4554	TEST320E	8217	7403					D[C] ADDR SELECT, CODE=*101*=COUT07=*1* ONLY ONE SET
					7401	K4/99	../..	../..	FI; K4=E63
51)	4556	TEST320F	8256	7405					BUTR(COUT07=DOUT07=XX), TARGET=*10=*
					7407	K3/99	../..	../..	FI; K3=F56,E95

Module codes: K1/DCS K2/UWORD K3/IRDECODE K4/DATAPATH K5/RTCACHE K6/TIMING K7/STATUS

###	ERROR code	Symbolic label	Line number	ENUA	TNUA	#1/##	#2/##	#3/##	Test summary + Print reference + Chip information
						=>Module sequence=>			
52)	4562	TEST320C	8157	7413					BUTR(X+D[C]+XX), D[C]=COUT07="0", COUT15="1"
					7417	K4/99	../..	../..	FI; K4=E63-E64
53)	4566	TEST551C	14539	NONE					B.M. DATAPATH COUNTER, BUTA(COUNT-18-377+D[C]), COUNT FROM (000)-(377)
					4531	K3/99	../..	../..	FI; K3=E56
					????	K2/80	K4/20	../..	K207=ACTIVE-BUT; K409=CNTR
54)	4570	TEST122A2	7172	7403					BUTR(COUNT+D[C]), D[C]=CINMUX="1", ALU=A-IOR-B
					7402	K3/99	../..	../..	FI; K3=E43,E61,E71
55)	4571	TEST721A	16840	4777					DATOB="BYTE, ODD ADDR JAM, BAU(120001)
					7341	K6/99	../..	../..	K408=UNIBUS-FUNCTION-DECODE
56)	4572	TEST537A	14413	7402					SHIFTER, 'SENDMUX' CMUX<00> OUTPUT ON CMUX/LEFT-1
					7400	K4/90	K5/5	K3/5	FI; K4=E77-E78,E80; K5=E97; K3=E61
57)	4573	TEST721B	16894	7402					DATOB="BYTE, ODD ADDR JAM; JAM=(101004)
					7407	K7/99	../..	../..	K705=JAM-FLAGS,K708=STATUS-MUX
58)	4575	TEST451A	14469	NONE					B.M. DATAPATH COUNTER, BUTA(SR1+0+COUNT-18-377), COUNT FROM (000)-(377)
					4531	K3/75	K4/25	../..	FI; K3=E36,E54; K4=E91
					4547	K3/90	K4/10	../..	FI; K3=E36,E54,E56,E64; K4=E91
					4555	K4/90	K2/10	../..	FI; K4=E20,E54,E87,E90-E91; K2=E55
					????	K2/80	K4/20	../..	K207=ACTIVE-BUT; K409=CNTR
59)	4576	TEST551P	14498	7400					B.M. DATAPATH COUNTER, BUTA(COUNT-18-377), COUNT FROM (000)-(377)
					7401	K3/99	../..	../..	FI; K3=E64
					7402	K3/99	../..	../..	FI; K3=E44
					????	K2/80	K4/20	../..	K207=ACTIVE-BUT; K409=CNTR
60)	4577	TEST721C	16918	7402					DATOB="BYTE, 16. BIT PBA, =I/O PAGE(5); SERVICE=(002340)
					7400	K6/99	../..	../..	FI; K6=E59
61)	4601	TEST115A1	5939	7412					ALU=NOT-A; A(052525), B(177777), D(125252), BITS<15:12>=(12)
					7400	K4/88	K7/10	K2/5	FI; K40E1,E4-E6,E7,E15-E16,E22,E24,E34,E40-E42, E98; K7=E41,E70; K2=E70
					7402	K4/99	../..	../..	FI; K40E33-E34
					7403	K4/75	K3/25	../..	FI; K4=E26; K3=E38
					7405	K4/55	K3/45	../..	FI; K4=E14,E21,E23; K3=E43,E52-E53,E84
					7407	K4/99	../..	../..	FI; K4=E4
					7410	K4/99	../..	../..	FI; K4=E34
					7413	K4/99	../..	../..	FI; K4=E26,E34,E62
					7416	K4/99	../..	../..	FI; K4=E17,E34,E54
					7417	K4/99	../..	../..	FI; K40E1,E6,E9,E10,F14-E18,E20-E21,E23-E24, E26-E27,E29,E37,E44,E46

Module codes: K1/DCS K2/UWORD K3/IRDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

###	ERROR code	Symbolic label	Line number	ENUA	TNUA	#1/##	#2/##	#3/##	Test summary + Print reference + Chip information
						=>Module sequence=>			
62)	4602	TEST115B4	6044	7402					BUTR(COUNT+D[C]), D[C]=PS[C]="0"
					7403	K4/50	K3/45	K2/10	FI; K4=E2-E3,E56,E63; K3=E36,E57,E63; K2=E70
63)	4603	TEST114A2	5906	7402					BUTR(COUNT+D[C]), D[C]=CINMUX=PS[C]="0", ALU=ZERO
					7403	K4/50	K3/50	../..	FI; K4=E56,E63,E72,E75; K3=E22,E41,E54,E71-E73
					7406	K3/99	../..	../..	FI; K3=E74
64)	4604	TEST115A4	5967	7403					BUTR(COUNT+D[C]), D[C]=CINMUX="1", ALU=NOT-A
					7402	K4/55	K3/45	../..	FI; K4=E2-E3,E58,E63,E71-E72,E78,E85; K3=E41,E43, E46,E54,E61-E62,E71,E76
65)	4605	TEST121C2	7009	7412					ALU=A-XOR-B; A(000000), B(125252), D(125252), BITS<11:06>=(52)
					7414	K7/99	../..	../..	FI; K7=E30
					7417	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN=DECODE
66)	4606	TEST116B	6326	7434					ALU=ZERO; A(125252), B(052525), D(000000)
					7402	K3/99	../..	../..	FI; K3=E102
					7417	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN=DECODE
67)	4607	TEST114A	5874	7434					ALU=ZERO; A(177777), B(177777), D(000000)
					7400	K3/99	../..	../..	FI; K3=E43,E62
					7417	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN=DECODE
68)	4611	TEST115C1	6104	7412					ALU=NOT-A; A(052525), B(000000), D(125252), BITS<15:12>=(12)
					7400	K3/99	../..	../..	FI; K3=E51,E82
					7405	K4/99	../..	../..	FI; K4=E6,E17,E23,E31-E34,E40-E43
					7417	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN=DECODE
69)	4612	TEST115B3	6077	7432					ALU=NOT-A; A(125252), B(177777), D(052525), BITS<05:00>=(25)
					7420	K4/90	K5/10	../..	FI; K4=E28,E37,E40-E41; K5=E68
					7427	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN=DECODE
70)	4613	TEST115B2	6060	7405					ALU=NOT-A; A(052525), B(177777), D(052525), BITS<11:06>=(25)
					7400	K4/90	K2/10	../..	FI; K4=E37,E41-E42,E89,E108; K2=E26
					7402	K4/99	../..	../..	FI; K4=E98
					7417	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN=DECODE
71)	4614	TEST115B1	6021	7405					ALU=NOT-A; A(125252), B(177777), D(052525), BITS<15:12>=(08)
					7400	K4/99	../..	../..	FI; K4=E16
					7401	K4/99	../..	../..	FI; K4=E34
					7404	K4/99	../..	../..	FI; K4=E14
					7407	K4/99	../..	../..	FI; K4=E14
					7412	K4/99	../..	../..	FI; K4=E14,E23
					7415	K4/99	../..	../..	FI; K4=E34
					7417	K4/90	K6/10	../..	FI; K4=E4,E12-E13,E21-E24; K6=E34

Module codes: K1/DCS K2/UWORD K3/IRDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

#(8)	ERROR code	Symbolic label	Line number	ENUA	TNUA	->Module sequence-> #1/##	#2/##	#3/##	Test summary - Print reference - Chip information
72)	4615	TEST115A3	6000	7425					ALU=NOT-A; A(052525), B(177777), D(125252), BITS<05:00>=(52)
					7410	K4/90	K5/10	../..	FI; K4=E31,E37,E40-E41,E44,E61,E75-E76; K5=E63,E66
					7477	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE
73)	4616	TEST115A2	5983	7412					ALU=NOT-A; A(052525), B(177777), D(125252), BITS<11:06>=(52)
					7400	K4/80	K2/15	K3/5	FI; K4=E5-E7,E11,E17-E18,E20,E24,E26,E28,E32, E41-E43,E60,E66-E68,E88-E89,E100; K2=E32-E33, E38-E39,E44-E45,E50-E51,E53-E54; K3=E27,E41,E62
					7402	K3/99	../..	../..	FI; K3=E53,E61
					7403	K5/99	../..	../..	FI; K5=E53
					7405	K4/90	K3/15	../..	FI; K4=E10,E22,E24,E98; K3=E84
					7434	K4/50	K2/50	../..	FI; K4=E71; K2=E117
					7477	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE
74)	4617	TEST115C2	6126	7412					ALU=NOT-A; A(052525), B(000000), D(125252), BITS<11:06>=(52)
					7400	K4/99	../..	../..	FI; K4=E41-E42
					7477	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE
75)	4620	TEST116A3	6271	7412					ALU=NOT-A-AND-B; A(000000), B(125252), D(125252), BITS<11:06>=(52)
					7400	K4/99	../..	../..	FI; K4=E41-E42,E69
					7477	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE
76)	4621	TEST116A2	6257	7403					BUTR(COUNT=D[C]), D[C]=ALU15="1", D=(125252)
					7402	K4/99	../..	../..	FI; K4=E2-E3,E56,E63-E64
77)	4623	TEST116A1	6235	7412					ALU=NOT-A-AND-B; A(000000), B(125252), D(125252), BITS<15:12>=(12)
					7400	K4/99	../..	../..	FI; K4=E15,E24,E31-E34,E40-E43
					7477	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE
78)	4624	TEST115D3	6203	7432					ALU=NOT-A; A(125252), B(000000), D(052525), BITS<05:00>=(25)
					7477	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE
79)	4625	TEST115D2	6186	7405					ALU=NOT-A; A(125252), B(000000), D(052525), BITS<11:06>=(25)
					7477	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE
80)	4626	TEST115D1	6164	7405					ALU=NOT-A; A(125252), B(000000), D(052525), BITS<15:12>=(05)
					7477	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE
81)	4627	TEST115C3	6143	7425					ALU=NOT-A; A(052525), B(000000), D(125252), BITS<05:00>=(52)
					7420	K4/50	K3/50	../..	FI; K4=E40; K3=E11
					7477	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE
82)	4630	TEST116C5	6429	7401					BUTR(D[C]=BA00), D[C]=D[C]="0"
					7401	K4/99	../..	../..	FI; K4=E63-E64,E78

Module codes: K1/DCS K2/UWORD K3/IRDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

#(8)	ERROR code	Symbolic label	Line number	ENUA	TNUA	->Module sequence-> #1/##	#2/##	#3/##	Test summary - Print reference - Chip information
83)	4631	TEST116C4	6413	7432					ALU=NOT-A-AND-B; A(000000), B(052525), D(052525), BITS<05:00>=(25)
					4631	K7/99	../..	../..	FI; K7=E43
					7477	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE
84)	4632	TEST116C3	6396	7405					ALU=NOT-A-AND-B; A(000000), B(052525), D(052525), BITS<11:06>=(25)
					7477	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE
85)	4633	TEST116C2	6381	7402					BUTR(COUNT=D[C]), D[C]=ALU15="0", D=(052525)
					7403	K4/99	../..	../..	FI; K4=E63-E64
86)	4635	TEST116C1	6359	7405					ALU=NOT-A-AND-B; A(000000), B(052525), D(052525), BITS<15:12>=(05)
					7477	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE
87)	4636	TEST116A5	6305	7403					BUTR(D[C]=BA00), D[C]=D[C]="1"
					4747	K2/99	../..	../..	FI; K2=E44,E85
					7401	K4/90	K3/10	../..	FI; K4=E2-E3,E56,E63-E64,E93,E104,E113-E114; K3=E44,E62
88)	4637	TEST116A4	6288	7425					ALU=NOT-A-AND-B; A(000000), B(125252), D(125252), BITS<05:00>=(52)
					7420	K4/99	../..	../..	FI; K4=E40,E68
					7477	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE
89)	4640	TEST117B2	6592	7412					ALU=A-AND-NOT-B; A(177777), B(052525), D(125252), BITS<11:06>=(52)
					7400	K4/99	../..	../..	FI; K4=E109
					7477	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE
90)	4641	TEST117B1	6570	7412					ALU=A-AND-NOT-B; A(177777), B(052525), D(125252), BITS<15:12>=(12)
					7400	K4/99	../..	../..	FI; K4=E4,E10,E23-E24
					7410	K4/99	../..	../..	FI; K4=E90
					7477	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE
91)	4642	TEST117A4	6549	7413					BUTR(X=D[C]=XX), D[C]=CINNUX=D[C]="0"; ALU=A-AND-NOT-B
					7417	K3/65	K4/35	../..	FI; K3=E71,E75; K4=E63
					7633	K3/99	../..	../..	FI; K3=E68
					7453	K3/99	../..	../..	FI; K3=E70
					7513	K3/99	../..	../..	FI; K3=E70
					7513	K3/99	../..	../..	FI; K3=E70
					7663	K3/99	../..	../..	FI; K3=E70
92)	4643	TEST117A3	6534	7432					ALU=A-AND-NOT-B; A(177777), B(125252), D(052525), BITS<05:00>=(25)
					7420	K4/65	K3/35	../..	FI; K4=E88-E99; K3=E61
					7477	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE
93)	4644	TEST117A2	6517	7405					ALU=A-AND-NOT-B; A(177777), B(125252), D(052525), BITS<11:06>=(25)
					7402	K7/99	../..	../..	FI; K7=E37

[Continued]

Module codes: K1/DCS K2/UWORD K3/IRDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

###	ERROR code	Symbolic label	Line number	ENDA	TNUA	#1/%%	#2/%%	#3/%%	Test summary - Print reference - Chip information
	1644	TEST117A2	(Continued)						ALU=A-AND-NOT-B; A(177777), B(125252), D(052525), BITS<11:06>=(28)
					7434	K5/99	../..	../..	FI; K5=E95
					7477	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE
94)	4645	TEST116D		6482	7434	ALU=ZERO; A(052525), B(125252), D(000000)			
					7477	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE
95)	4647	TEST117A1		6495	7405	ALU=A-AND-NOT-B; A(177777), B(125252), D(052525), BITS<15:12>=(05)			
					7400	K4/85	K3/15	../..	FI; K4=E5,E13,E21,E23; K3=E71
					7417	K3/99	../..	../..	FI; K3=E84
					7477	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE
96)	4650	TEST120R4		6799	7402	BUTR(COUNT=DIC1), D(C)=CINMUX=0, ALU=A-AND-B			
					7403	K3/99	../..	../..	FI; K3=E71
97)	4651	TEST120R3		6784	7432	ALU=A-AND-B; A(052525), B(177777), D(052525), BITS<05:00>=(25)			
					7420	K4/99	../..	../..	FI; K4=E40,E88
					7477	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE
98)	4652	TEST120R2		6767	7405	ALU=A-AND-B; A(052525), B(177777), D(052525), BITS<11:06>=(25)			
					7400	K4/99	../..	../..	FI; K4=E41-E42
					7477	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE
99)	4653	TEST120R1		6745	7405	ALU=A-AND-B; A(052525), B(177777), D(052525), BITS<15:12>=(05)			
					7412	K4/99	../..	../..	FI; K4=E13,E15,E20,E22,E24,E31-E34,E40-E43
					7477	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE
100)	4654	TEST120A3		6724	7425	ALU=A-AN -B; A(125252), B(177777), D(125252), BITS<05:00>=(52)			
					7477	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE
101)	4655	TEST120A2		6707	7412	ALU=A-AND-B; A(125252), B(177777), D(125252), BITS<11:06>=(52)			
					7477	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE
102)	4656	TEST117C1		6633	7434	ALU=A-AND-NOT-B; A(000000), B(000000), D(000000)			
					7477	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE
103)	4657	TEST117B3		6609	7425	ALU=A-AND-NOT-B; A(177777), B(052525), D(125252), BITS<05:00>=(52)			
					7420	K4/99	../..	../..	FI; K4=E88-E89
					7477	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE
104)	4660	TEST121B4		6960	7401	BUTR(DIC)=BA00), D(C)=ALU00=0, D=(125252)			
					7403	K4/99	../..	../..	FI; K4=E86,E63
105)	4661	TEST121B3		6945	7425	ALU=A-XOR-B; A(177777), B(052525), D(125252), BITS<05:00>=(52)			
					7477	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE
106)	4662	TEST121B2		6928	7412	ALU=A-XOR-B; A(177777), B(052525), D(125252), BITS<11:06>=(52)			
					7477	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE

Module codes: K1/DCS K2/UWORD K3/IRDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

###	ERROR code	Symbolic label	Line number	ENDA	TNUA	#1/%%	#2/%%	#3/%%	Test summary - Print reference - Chip information
107)	4663	TEST121B1		6906	7412	ALU=A-XOR-B; A(177777), B(052525), D(125252), BITS<15:12>=(12)			
					7477	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE
108)	4665	TEST121A1		6832	7405	ALU=A-XOR-B; A(000000), B(052525), D(052525), BITS<15:12>=(05)			
					7477	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE
109)	4667	TEST120A1		6685	7412	ALU=A-AND-B; A(125252), B(177777), D(125252), BITS<15:12>=(12)			
					7477	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE
110)	4671	TEST122A1		7148	7434	ALU=A-IOR-B; A(000000), B(000000), D(000000)			
					7400	K4/99	../..	../..	FI; K4=E1,E7-E8,E11,E13,E22,E92,E98,E100-E102, E109-E112
					7402	K4/99	../..	../..	FI; K4=E7,E10,E21-E22,E101
					7417	K4/99	../..	../..	FI; K4=E110
					7420	K4/99	../..	../..	FI; K4=E22,E102,E112
					7421	K4/99	../..	../..	FI; K4=E22,E112
					7422	K4/99	../..	../..	FI; K4=E112
					7424	K4/99	../..	../..	FI; K4=E21,E112
					7426	K4/99	../..	../..	FI; K4=E102
					7430	K4/99	../..	../..	FI; K4=E102
					7433	K4/99	../..	../..	FI; K4=E102
					7477	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE
111)	4673	TEST130A1		7283	7434	ALU=A-PLUS-B-PLUS-0; A(052525), B(052525), D(125252)			
					4777	K2/99	../..	../..	FI; K2=E79
					7400	K4/60	K3/40	../..	FI; K4=E1,E6,E49,F64,E88-E89,E98,E101,E108,E110; K3=E38,E42,E82,E92
					7402	K4/70	K3/30	../..	FI; K4=E21,E49,E98,E100-E102,E109-E112; K3=E41,E43, E82-E83,E84
					7420	K4/99	../..	../..	FI; K4=E49,E102,E112
					7421	K4/85	K3/15	../..	FI; K4=E49,E71; K3=E71
					7423	K4/99	../..	../..	FI; K4=E112
					7427	K4/99	../..	../..	FI; K4=E112
					7431	K4/99	../..	../..	FI; K4=E102
					7477	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE
112)	4674	TEST121A4		6885	7403	BUTR(DIC)=BA00), D(C)=ALU00=1, D=(052525)			
					7401	K4/99	../..	../..	FI; K4=E56,E61,K114
113)	4675	TEST121A3		6871	7432	ALU=A-XOR-B; A(000000), B(052525), D(052525), BITS<05:00>=(25)			
					7420	K4/99	../..	../..	FI; K4=E10
					7477	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE
114)	4676	TEST121A2		6854	7405	ALU=A-XOR-B; A(000000), B(052525), D(052525), BITS<11:06>=(25)			
					7477	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE

Module codes: K1/DCS K2/UWORD K3/IRDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

###	ERROR code	Symbolic label	Line number	ENUA	TNUA	->Module sequence-> #1/##	#2/##	#3/##	Test summary - Print reference - Chip information
115)	4677	TEST117C2	6655	7417					BUTR(X=D(C)=XX), D(C)=CINHUX=D(C)="1", ALU=A-AND-NOT-B
					4756	K3/50	K2/50	../..	FI; K3=E12,E18; K2=E80-E81
					7413	K4/90	K2/10	../..	FI; K4=E2-E3,E56,E63,E104; K2=E1
116)	4701	TEST350	8337	NONE					(NUA SEQUENCING LOGIC ERROR)
					????	K1/99	../..	../..	INTERNAL DCB ERROR
117)	4705	TEST352A	8670	7434					BSPLO ADDRESSING, USING BSP/RIF ADDRESS=(02)
					7403	K4/99	../..	../..	FI; K4=E13
					7417	K4/99	../..	../..	FI; K4=E13
					7477	K4/99	../..	../..	K407=BSP-ADDRS-MUX/REG(RIF)
118)	4707	TEST121C1	6987	7412					ALU=A-XOR-B; A(000000), B(125252), D(125252), BITS<15:12>=(12)
					7477	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE
119)	4710	TEST763D	18149	7407					BUTR(BG-SERVICE) NEGATED; AFTER INTR DETECTED/SERVICED
					4747	K5/99	../..	../..	FI; K5=E81-E82
					7403	K7/99	../..	../..	K703=BR-REQUEST-INTR,K704=BR-GRANT-INTR
120)	4711	TEST762F	18031	7403					BUTR(SERVICE) ASSERTED; DL11-W BR6 INTR PRESENT
					7402	K2/99	../..	../..	FI; K2=E100
121)	4712	TEST351C	8614	7434					ASPHI ADDRESSING, USING ASP/RIF ADDRESS=(05)
					4747	K7/99	../..	../..	FI; K2=E1
					7477	K4/99	../..	../..	K406=ASP-ADDRS-MUX/REG(RIF)
122)	4713	TEST352B	8692	7434					BSPLO ADDRESSING, USING BSP/RIF ADDRESS=(03)
					7477	K4/99	../..	../..	K407=BSP-ADDRS-MUX/REG(RIF)
123)	4716	TEST121C4	7040	7417					BUTR(X=D(C)=XX), D(C)=ALU07="1", D=(125252)
					7413	K4/99	../..	../..	FI; K4=E56,E63
124)	4717	TEST121C3	7025	7425					ALU=A-XOR-B; A(000000), B(125252), D(125252), BITS<05:00>=(52)
					7420	K4/99	../..	../..	FI; K4=E88
					7477	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE
125)	4720	TEST763B	18108	7402					VECTOR LOADED FROM DL11-W LINE CLOCK; CHECK=(000100)
					7400	K7/85	K6/15	../..	FI; K7=E13,E17,E20-E21,E28; K6=E34
126)	4721	TEST763C	18128	7401					BUTR(VECTOR-LOAD) NEGATED; AFTER INTR DETECTED/SERVICED
					7403	K7/90	K6/20	../..	FI; K7=E10,E21,E94; K6=E26
127)	4722	TEST351B	8591	7434					ASPLO ADDRESSING, USING ASP/RIF ADDRESS=(03)
					7477	K4/99	../..	../..	K406=ASP-ADDRS-MUX/REG(RIF)

Module codes: K1/DCS K2/UWORD K3/IRDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

###	ERROR code	Symbolic label	Line number	ENUA	TNUA	->Module sequence-> #1/##	#2/##	#3/##	Test summary - Print reference - Chip information
128)	4723	TEST351C	8612	7434					ASPHI ADDRESSING, USING ASP/RIF ADDRESS=(04)
					4777	K2/99	../..	../..	FI; K2=E1
					7271	K2/99	../..	../..	FI; K2=E28
					7400	K4/99	../..	../..	FI; K4=E14
					7477	K4/99	../..	../..	K406=ASP-ADDRS-MUX/REG(RIF)
129)	4724	TEST352C	8713	7434					BSPHI ADDRESSING, USING BSP/RIF ADDRESS=(04)
					7400	K4/99	../..	../..	FI; K4=E12
					7477	K4/99	../..	../..	K407=BSP-ADDRS-MUX/REG(RIF)
130)	4725	TEST352D	8735	7434					BSPHI ADDRESSING, USING BSP/RIF ADDRESS=(05)
					7477	K4/99	../..	../..	K407=BSP-ADDRS-MUX/REG(RIF)
131)	4726	TEST121D2	7083	7405					ALU=A-XOR-B; A(177777), B(125252), D(052525), BITS<11:06>=(25)
					7477	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE
132)	4727	TEST121D1	7061	7405					ALU=A-XOR-B; A(177777), B(125252), D(052525), BITS<15:12>=(05)
					7477	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE
133)	4730	TEST762E	18006	7402					DL11-W BR6 INTR PENDING, LAST BUS DATO; SERVICE=(043740)
					7400	K3/99	../..	../..	FI; K3=E50
					7401	K2/99	../..	../..	FI; K2=E100
					7407	K7/99	../..	../..	K708=STATUS-MUX
134)	4732	TEST351A	8569	7434					ASPLO ADDRESSING, USING ASP/RIF ADDRESS=(02)
					7400	K4/99	../..	../..	FI; K4=E9,E14
					7402	K7/99	../..	../..	FI; K7=E52
					7403	K4/68	K2/35	../..	FI; K4=E15,E86; K2=E52
					7417	K4/99	../..	../..	FI; K4=E4,E15
					7477	K4/99	../..	../..	K406=ASP-ADDRS-MUX/REG(RIF)
135)	4734	TEST122A3	7188	7402					BUT(D<14:00>=D0D15), D=(000000), TARGET="10"
					7400	K2/95	K3/5	../..	FI; K2=E40,E86,E88-E89,E94-E95,E104-E105; K3=E56, E66
					7403	K3/90	../..	../..	FI; K3=E54,E95
					7406	K3/99	../..	../..	FI; K3=E74
136)	4735	TEST122A4	7202	7401					BUT(D<14:00>=D0D15), D=(125252), TARGET="01"
					7400	K3/99	../..	../..	FI; K3=E84-E85,E63,E95
					7403	K2/85	K3/15	../..	FI; K2=E40,E86,E90; K3=E56
137)	4736	TEST121D4	7115	7413					BUTR(X=D(C)=XX), D(C)=ALU07="0", D=(052525)
					7417	K4/99	../..	../..	FI; K4=E56,E63

Module codes: K1/DCS K2/UWORD K3/IRDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

###	ERROR code	Symbolic label	Line number	ENUA	TNUA	->Module sequence-> #1/%%	#2/%%	#3/%%	Test summary - Print reference - Chip information
138)	4737	TEST121D9	7100	7432	ALU=A+XOR=B; A(177777), B(125252), D(052525), BITS<05:00>=(25)				
					7434	K2/99	../..	../..	FI; K2=E2
					7477	K6/90	K3/10	../..	K406=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE
139)	4740	TEST762C	17946	7403	BUTR(BG=SERVICE) ASSERTED: BR6 INTR PENDING, PROC PRIO=(5)				
					7407	K7/99	../..	../..	FI; K7=E3,E5,E11,E20,E28
140)	4741	TEST762D	17980	7417	BUTR(BG=SERVICE+FP=SERVICE) ASSERTED; BR6 INTR PRESENT				
					7407	K3/80	K7/20	../..	FI; K3=E80,E76; K7=E80
141)	4742	TEST350C	8471	7434	ASPLO ADDRESSING, USING ASP/DF ADDRESS FROM (7):(0)				
					7400	K4/80	K3/20	K5/5	K3/5 FI; K4=E1,E5,E7,E10,E18-E16,E23,E25,E31-E34, E40-E43,E99,E110; K3=E48-E47,E89; K5=E6); K2=E55
					7402	K4/65	K3/35	K6/9	FI; K4=E1,E6,E15,E17,E22,E33,E06,E92,E99-E102, E109-E112; K3=E19,E24-E26,E45,E47; K6=E40
					7403	K3/99	../..	../..	FI; K3=E59
					7406	K4/60	K3/40	../..	FI; K4=E6; K3=E46
					7417	K4/99	../..	../..	FI; K4=E43
					7420	K4/99	../..	../..	FI; K4=E25,E31-E33,E102,E112
					7421	K4/99	../..	../..	FI; K4=E31
					7422	K4/99	../..	../..	FI; K4=E31
					7424	K4/99	../..	../..	FI; K4=E31
					7426	K4/99	../..	../..	FI; K4=E32
					7430	K4/99	../..	../..	FI; K4=E32
142)	4743	TEST350D	8489	7434	ASPHI ADDRESSING, USING ASP/DF ADDRESS FROM (0):(7)				
					7400	K4/99	../..	../..	FI; K4=E6-E7,E16,F17,E42,E99-E100,E109
					7402	K4/75	K3/25	../..	FI; K4=E17,E34,E99; K3=E45
					7417	K4/99	../..	../..	FI; K4=E15,E109
					7420	K4/99	../..	../..	FI; K4=E40-E41,E74,E91,E99,E111
					7421	K4/99	../..	../..	FI; K4=E111
					7422	K4/99	../..	../..	FI; K4=E111
					7424	K4/99	../..	../..	FI; K4=E111
					7426	K4/99	../..	../..	FI; K4=E92
					7433	K4/99	../..	../..	FI; K4=E99
143)	4744	TEST740A	17556	7402	INSTR1=PREFETCH, BC<0>=BC40				
					4747	K2/99	../..	../..	FI; K2=E24
					7401	K2/99	../..	../..	FI; K2=E105
					7403	K5/99	../..	../..	FI; K5=E97
144)	4745	TEST761A	17726	7402	BUTR(SERVICE) NEGATED: AFTER CLEAR ALL SERVICE CONDITIONS, UNIBUS INIT				
					4747	K7/99	../..	../..	FI; K7=E57
					7403	K7/99	../..	../..	K707/9=UNIBUS-UCON=INIT,K704=BG-GRANT-INTR

Module codes: K1/DC8 K2/UNORD K3/IRDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

###	ERROR code	Symbolic label	Line number	ENUA	TNUA	->Module sequence-> #1/%%	#2/%%	#3/%%	Test summary - Print reference - Chip information
145)	4750	TEST131B2	7474	7413	BUTR(X=D(C)=XX), D(C)=COUT15=="0"; A(0)+B(1)+CI(0)=D(1)+CO(0)				
					7403	K4/90	K3/10	../..	K406=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE
146)	4751	TEST762B	17924	7407	BUTR(BG=SERVICE) NEGATED; BR6 INTR PENDING, PROC PRIO=(6)				
					7403	K7/99	../..	../..	FI; K7=E11,E76
					7407	K7/99	../..	../..	K703=BR-REQUEST-INTR
147)	4752	TEST350A	8417	7434	BSPLO ADDRESSING, USING BSP/BF ADDRESS FROM (0):(7)				
					7400	K4/90	K2/10	../..	FI; K4=E1,E5,E7-E9,E12,E14,E17,E21,E26,E35,E92, E99-E102,E109-E112; K2=E30,E55
					7402	K4/99	../..	../..	FI; K4=E22,E100
					7403	K4/99	../..	../..	FI; K4=E47
					7407	K4/99	../..	../..	FI; K4=E20
					7410	K4/99	../..	../..	FI; K4=E6,E13
					7413	K7/99	../..	../..	FI; K7=E25
					7417	K4/99	../..	../..	FI; K4=E4,E13,E15,E17,E22,E26,E109
					7420	K4/99	../..	../..	FI; K4=E92,E111
					7421	K4/99	../..	../..	FI; K4=E111
					7422	K4/99	../..	../..	FI; K4=E111
					7424	K4/99	../..	../..	FI; K4=E111
					7426	K4/99	../..	../..	FI; K4=E92
					7430	K4/99	../..	../..	FI; K4=E92
					7431	K4/99	../..	../..	FI; K4=E92,E99,E111
148)	4753	TEST350B	8444	7434	BSPHI ADDRESSING, USING BSP/BF ADDRESS FROM (7):(0)				
					7400	K4/80	K2/15	K7/5	K5/5 FI; K4=E6-E7,E11-E15,E17,E21-E22,E26,E59, E87,E96,E101,E110; K2=E32,E44,E80,E53; K7=E66; K5=E66
					7417	K4/85	K5/15	../..	FI; K4=E13,E15,E22; K5=E66
					7420	K4/99	../..	../..	FI; K4=E102,E112
149)	4755	TEST731A	17321	7402	BUS DATOB=BYTE=ODD, BA=(000001); DBUF=D=(000000)				
					4415	K2/99	../..	../..	FI; K2=E30
					4747	K6/65	K2/35	../..	FI; K6=E5; K2=E16
					????	K6/99	../..	../..	K605=UNIBUS-FUNCTION-DECODE
150)	4760	TEST720C	16813	7402	DATIB=-BYTE, 16. BIT PBA, =I/D PAGE(3); SERVICE=(100340)				
					7400	K5/99	../..	../..	FI; K6=E89
151)	4761	TEST763A	18062	4733	ALLOW-BG(1)H GIVEN TO BR6 INTR; BUTR (VECTOR-LOAD) ASSERTED				
					4747	../..	../..	../..	!! ALSO CHECK GRANT CONTINUITY CARD !!
					4731	K7/95	K2/5	K6/5	K5/5 FI; K7=E4-E5,E8,E10,E12-E13,E17-E16,E20-E22, E25,E29,E33,E35,E37,E41,E51,E59,E61,E76-E77,E79, E94,E101; K2=E10; K6=E56; K5=E53
					4756	K7/99	../..	../..	FI; K7=E4-E7

Module codes: K1/DC8 K2/UNORD K3/IRDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

###	ERROR code	Symbolic label	Line number	ENUA	TNUA	#1/##	#2/##	#3/##	Test summary + Print reference + Chip information
						=>Module sequence=>			
152)	4771	TEST761B	17765	7401	BUTR(VECTOR-LOAD) NEGATED; AFTER UNIBUS INIT				
					7403	K7/99	../..	../..	K707/9=UNIBUS-UCON-INIT,K704=BG-GRANT-INTR
					7403	K3/99	../..	../..	FI; K3=E44
153)	4773	TEST762A	17829	4343	(NUA SEQUENCING LOGIC ERROR)				
					4747	K6/75	K5/20	K7/5	K4/9 FI; K6=E11,E19,E25,E29,E33,E49=E50; K5=E2, E10,E33,E80,E82,E97; K7=E66; K4=E48
154)	4774	TEST761C	17786	7407	BUTR(BG-SERVICE) NEGATED; AFTER UNIBUS INIT				
					7403	K7/99	../..	../..	K707/9=UNIBUS-UCON-INIT,K704=BG-GRANT-INTR
155)	4775	TEST010	2802	5245	BUTA(SUBR-B) => BUTA(RETURN) SEQUENCE, RETURN="1010 1010 0101"=(5245)				
					4245	K2/65	K3/35	../..	FI; K2=E35,E41,E61,E77,E88; K3=E4,E7,E15,E43
					4376	K2/90	K3/15	../..	FI; K2=E33,E35,E39=E40,E45=E48,E81,E84; K3=E34
					4777	K2/99	../..	../..	FI; K2=E9,E13=E14,E41,E57,E63,E82,E85=E86,E91=E92, E97,E106,E110,E116,E118
					5005	K2/99	../..	../..	FI; K2=E25
					5045	K2/85	K4/15	../..	FI; K2=E7,E13,E25,E31,E37,E43,E49,E52,E70,E97; K4=E12-E13
					5105	K2/99	../..	../..	FI; K2=E11
					5125	K2/99	../..	../..	FI; K2=E13
					5205	K2/80	K4/15	../..	FI; K2=E4,E7,E10,E13,E16,E22,E25,E28,E71,E97; K4=E1
					5240	K2/99	../..	../..	FI; K2=E26
					5241	K2/85	K4/15	../..	FI; K2=E4,E7,E10,E14,E16,E22,E26,E20,E62,E91; K4=E20,E86
					5242	K2/99	../..	../..	FI; K2=E14
					5244	K2/70	K4/30	../..	FI; K2=E8,E14,E26,E58,E64,E67,E71,E73,E79,E91; K4=E12,E14
					5245	K2/65	K4/15	../..	FI; K2=E25,E31,E35,E37,E43,E49,E52,E58,E62,E64,E67, E73,E77,E79,E91,E97) K4=E3,E70
					5247	K2/99	../..	../..	FI; K2=E8,E14,E26
					5252	K2/99	../..	../..	FI; K2=E14
					5258	K2/99	../..	../..	FI; K2=E8,E14,E26
					5256	K2/99	../..	../..	FI; K2=E8
					5267	K2/99	../..	../..	FI; K2=E14
					5265	K2/69	../..	../..	FI; K2=E7,E13,E25
					5345	K2/99	../..	../..	FI; K2=E7,E13,E25
					5365	K2/99	../..	../..	FI; K2=E7,E13
					5376	K2/99	../..	../..	FI; K2=E40
					7245	K2/99	../..	../..	FI; K2=E35,E41,E77
					7247	K2/99	../..	../..	FI; K2=E40
					7273	K3/85	K2/45	../..	FI; K3=E4,E6=E7,E11,E15,E43; K2=E61,E77,E85
					7401	K2/99	../..	../..	FI; K2=E41
					7777	K2/99	../..	../..	FI; K2=E11

Module codes: K1/DCS K2/UWORD K3/IFDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

###	ERROR code	Symbolic label	Line number	ENUA	TNUA	#1/##	#2/##	#3/##	Test summary + Print reference + Chip information
						=>Module sequence=>			
156)	5146	TEST005	2711	4474	NUA SEQUENCING, PAGE (5) => (4), UBF=(34)				
					4574	K3/80	K2/80	../..	FI; K3=E58; K2=E7
					5441	K3/60	K2/40	../..	FI; K3=E10,E18,E20; K2=E30,E54
157)	5443	TEST722A	16951	4777	INVALIDATE, ODD ADDR JAM; BAa(140001)				
					7341	K6/75	K4/15	K3/15	FI; K6=E71,E76,E81,E88,E96; K4=E51; K3=E11
					????	K6/99	../..	../..	K605=UNIBUS-FUNCTION=DECODE
158)	5465	TEST720A	16730	4777	DATIB=BYTE, ODD ADDR JAM, BAa(000001)				
					7341	K5/60	K3/50	../..	FI; K5=E67,E94; K3=E34
					????	K6/99	../..	../..	K605=UNIBUS-FUNCTION=DECODE
159)	5467	TEST713A	16594	4777	DATI=NOINT, ILLEGAL INTERNAL ADDR JAM, BAa(177776)				
					????	K6/99	../..	../..	K605=UNIBUS-FUNCTION=DECODE,INTERNAL=ADDR=DETECT
160)	5471	TEST712A	16460	4777	DATI, INTERNAL ADDRESS JAM, BAa(177776)				
					????	K6/99	../..	../..	K605=UNIBUS-FUNCTION=DECODE,INTERNAL=ADDR=DETECT
161)	5473	TEST711A	16328	4777	DATOB=BYTE, SSYN TIMEOUT JAM, BAa(160001)				
					7341	K5/70	K4/20	K7/5	K5/5 FI; K6=E4,E11,E28,E62,E64,E78=E79,E98=E99, E101=E102; K4=E18,E28; K7=E60; K5=E55
					????	K6/99	../..	../..	K605=UNIBUS-FUNCTION=DECODE
162)	5475	TEST710A	16157	4777	DATO, ODD ADDRESS ERROR JAM, BAa(160001)				
					4747	K6/99	../..	../..	FI; K6=E60
					7341	K6/75	K7/15	K4/10	K5/5 K3/5 FI; K6=E2,E4,E10,E38,E40,E42=E43, E46=E47,E54=E56,E60,E64,E69,E71=E72,E76,E79,E81, E89,E95,E98=E99,E103,E108; K7=E22,E25,E39,E63, E48=E46,E65; K4=E9,E18,E26,E30,E71; K5=E44,E64; K3=E82
					7401	K5/99	../..	../..	FI; K5=E97
					7445	K6/59	../..	../..	FI; K6=E43
					????	K6/89	../..	../..	K605=UNIBUS-FUNCTION=DECODE
163)	5477	TEST702A	16040	7402	LOAD BA<17:16>="01", 18, BIT MODE, READ THRU STATUS=MUX(SERVICE)<9:8>				
					4747	K2/99	../..	../..	FI; K2=E34
					7400	K6/40	K5/25	K4/20	K7/10 K2/5 FI; K6=E11,E51=E54,E59; K5=E4,E13, E15,E26,E54,E84,E97; K4=E20,E65; K7=E48,E56; K2=E61
164)	5500	TEST374D2	10178	7434	A/B SP WRITE FCN WR(B,HI,A=ADDR) AND WR(B,HI,B=ADDR)				
					7477	K4/99	../..	../..	K406=SP-REWRITE-CNTL,K406/7=A/B=SPADS
165)	5502	TEST374F2	10251	7414	A/B SP WRITE FCN WR(AB,LO,A=ADDR) AND WR(AB,LO,B=ADDR)				
					7400	K2/99	../..	../..	FI; K2=E117

(Continued)

Module codes: K1/DCS K2/UWORD K3/IFDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

###	ERROR code	Symbolic label	Line number	ENUA	TNUA	#1/%% ->Module sequence->	#2/%%	#3/%%	Test summary + Print reference + Chip information
	5402	TEST374E2	(Continued)						A/B SP WRITE FCN WR(AB,LO,A=ADDR) AND WR(AB,LO,B=ADDR)
					7420	K4/99	../..	../..	FI; K4=E10
					7477	K4/99	../..	../..	K405=SP=REWRITE-CNTL,K406/7=A/B=SPADS
166)	5504	TEST374F2	10324	7434					A/B SP WRITE FCN WR(AB,HI,A=ADDR) AND WR(AB,HI,B=ADDR)
					7477	K4/99	../..	../..	K405=SP=REWRITE-CNTL,K406/7=A/B=SPADS
167)	5505	TEST712R	16512	7402					DATI, INTERNAL ADDR; JAM=(001000)
					7400	K6/70	K7/15	K3/19	K5/5 FI; K6=E46,E49-E51,E91,E99; K7=E19,E59,E61, E81; K3=E46,E69; K5=E13,E72
					7401	K6/99	../..	../..	FI; K6=E76
					7777	K6/99	../..	../..	K405=INTERNAL-ADDR-DETECT
168)	5507	TEST410	10666	NONE					(NUA SEQUENCING LOGIC/DCS ERROR)
					7777	K1/99	../..	../..	INTERNAL DCS ERROR
169)	5511	TEST620R	15262	7406					BUTM(INIT=JAM) NEGATED=0 AFTER CLR=JAM=ERRDR& UCDN
					7407	K7/90	K2/10	../..	FI; K7=E18,E61,E51; K2=E112
170)	5512	TEST701D	16011	7402					BUTR(D[C]=BA00), BA<00>=0
					7403	K3/99	../..	../..	FI; K3=E64
171)	5513	TEST716A	10595	5440					BUTA(R=IOR=1) DOES NOT CAUSE A BRANCH
					5441	K3/99	../..	../..	FI; K3=E64
172)	5517	TEST379A	10512	7432					DAD/3 CAUSES BYTE-WRITE(LO) TO ASPLO; DAD/1 OR DAD/2 NOT
					7400	K3/80	K4/20	../..	FI; K3=E17,E20,E36,E38,E46; K4=E5,E99
					7402	K3/75	K4/15	../..	FI; K3=E27,E36,E38,E46,E49,E69-E80; K4=E5,E33
					7433	K4/99	../..	../..	FI; K4=E26
					7434	K3/99	../..	../..	FI; K3=E59
173)	5521	TEST102A	5225	7432					CSP ADDR&&&, LOC(02)); EMIT/CSP/ALU=B/D/DBUF/IR WITH DATA=(000138)
					7400	K4/40	K3/40	../..	FI; K4=E47,E55,E86-E89,E95-E96,E98,E105,E108; K3=E8,E8-E10,E14,E32,E35,E42,E92
					7402	K4/70	K3/30	../..	FI; K4=E85,E98,E98; K3=E12-E13,E32,E35,E42
					7403	K4/99	../..	../..	FI; K4=E96
					7410	K3/75	K4/25	../..	FI; K3=E1-E3,E8,E13-E14,E42; K4=E1,E30
					7417	K4/99	../..	../..	FI; K4=E105,E109
					7420	K4/70	K6/10	K2/10	K3/9 K7/5 K5/5 FI; K4=E1,E4,E7-E8,E11-E12, E31-E32,E35,E40-E41,E52,E56,E65-E66,E72,E74,E79, E80-E92,E96,E100-E102,E105-E106,E109-E112; K6=E65,E67,E73,E75,E108; K2=E83,E94,E98,E100, E102-E104,E107-E108,E111,E113-E114; K3=E72; K7=E05-E06; K5=E63,E66
					7421	K4/99	../..	../..	FI; K4=E87
					7427	K6/55	K4/30	K7/15	FI; K6=E73; K4=E89; K7=E17
					7433	K4/90	K3/15	../..	FI; K4=E89,E105; K3=E8
					7434	K4/75	K3/10	K6/10	K7/5 K2/5 FI; K4=E1-E5,F7,E9-E11,E16,E20-E26,

Module codes: K1/DCS K2/UWORD K3/IRDECODE K4/DATAPATH K5/XTCACHE K6/TIMING K7/STATUS

###	ERROR code	Symbolic label	Line number	ENUA	TNUA	#1/%% ->Module sequence->	#2/%%	#3/%%	Test summary + Print reference + Chip information
	5421	TEST102A	(Continued)						CSP ADDRESS, LOC(02); EMIT/CSP/ALU=B/D/DBUF/IR WITH DATA=(000125)
					7434	K4/75	K3/10	K6/10	E28-E30,E38,E47,E51,E55-E56,E59,E64,E66,E70-E71, E74,E87-E89,E96,E98-E99,E108; K3=E2-E3,E12,E35, E42-E43,E53,E59,E82,E84; K6=E31,E34,E42,E82, E103; K7=E76; K2=P102
174)	5423	TEST701B	15962	7403					BUTR(D[C]=BA00), BA<00>=1
					7402	K3/99	../..	../..	K108=MICROBRANCH
175)	5424	TEST374C2	10105	7434					A/B SP WRITE FCN WR(B,LO,A=ADDR) AND WR(B,LO,B=ADDR)
					7400	K3/99	../..	../..	FI; K3=E45
					7402	K4/99	../..	../..	FI; K4=E10,E17,E21-E22
					7410	K3/99	../..	../..	FI; K3=E91
					7421	K4/99	../..	../..	FI; K4=E21-E22
					7477	K4/99	../..	../..	K405=SP=REWRITE-CNTL,K406/7=A/B=SPADS
176)	5429	TEST004	2692	5146					NUA SEQUENCING, NO "BUT"
					5147	K3/99	../..	../..	FI; K3=E38
					5156	K3/90	../..	../..	FI; K3=E37
					5346	K3/99	../..	../..	FI; K3=E23
177)	5531	TEST103A	5370	7412					CSP ADDRESS, LOC(13)) EMIT/CSP/ALU=B/D/DBUF/IR WITH DATA=(125200)
					7400	K6/99	../..	../..	FI; K4=E59,E66,E74,E92,E102,E109-E112
178)	5533	TEST610B1	14969	7413					PB[CC]=(13); IR=(105300), CC=NOM=ADDR(132)
					7401	K3/99	../..	../..	FI; K3=E55
					7403	K3/99	../..	../..	FI; K3=E63,E72
					7411	K3/90	K6/10	../..	FI; K3=E55,E62,E65,E85; K4=E72
					7417	K3/60	K2/40	../..	FI; K3=E85,E72; K2=E105
179)	5535	TEST610C1	14987	7407					PB[CC]=(07); IR=(072000), CC=NOM=ADDR(437)
					7405	K3/99	../..	../..	FI; K3=E33
					7406	K3/99	../..	../..	FI; K3=E62
					7415	K2/99	../..	../..	FI; K2=E6
180)	5541	TEST105A	5592	7432					SR LOAD/READ, SR=(000128), REG=SR/LOAD; ALU=A/D/DBUF/IR PATH
					7400	K4/85	K3/10	K2/5	FI; K4=E1,E3,E11,E16,E18,E20,E27,E30,E35-E36, E48-E47,E52,E80; K3=E5,E10,E14,E51,E66; K2=E99
					7402	K4/85	K3/10	K2/5	FI; K4=E30,E35,E48-E47; K3=E8,E14; K2=E101
					7403	K4/50	K3/50	../..	FI; K4=E80; K3=E9
					7406	K3/99	../..	../..	FI; K3=E9
					7410	K4/99	../..	../..	FI; K4=E16,E52
					7412	K3/99	../..	../..	FI; K3=E8
					7417	K4/99	../..	../..	FI; K4=E27,E36,E47,E52
					7420	K4/90	K3/10	../..	FI; K4=E1,E3,E7,E11,E16,E20,E26,E31-E32,E34-E35. E37,E40-E42,E44,E47-E48,E52-E54,E57,E60-E62, E67-E68,E75-E76,E80; K3=E41,E43,E51,E67,E71,E75, E82 (Continued)

Module codes: K1/DCS K2/UWORD K3/IRDECODE K4/DATAPATH K5/XTCACHE K6/TIMING K7/STATUS

###	ERROR code	Symbolic label	Line number	ENUA	TNUA	->Module sequence-> #1/%%	#2/%%	#3/%%	Test summary - Print reference + Chip information
	5541	TEST105A	[Continued]						SR LOAD/READ, SR=(000125), RE@=SR/LOAD; ALU=A/D/DBUF/IR PATH
					7427	K4/99	../..	../..	FI; K4=E1,E7,E16,E33,E35
					7430	K4/99	../..	../..	FI; K4=E1
					7431	K4/99	../..	../..	FI; K4=E36,E45
					7433	K4/99	../..	../..	FI; K4=E35,E37,E47,E86
					7434	K4/99	K3/5	../..	FI; K4=E1,E4,E7,E11,E16,E30-E32,E38-E36,E39,E43, E46,E47-E49,E82-E53,E55,E57,E65,E80; K3=E102
181)	5545	TEST610D1	15101	7410					P@(CC)=(10); IR=(072000), CC=ROM=ADDR(216)
					7412	K3/99	../..	../..	FI; K3=E33,E62
182)	5547	TEST104A	5516	7425					CSP BA@CON ADDR, LOC(16); EMIT/CSP/ALU=B/D/DBUF/IR WITH DATA=(000152)
					7400	K4/99	../..	../..	FI; K4=E38,E96
					7417	K4/99	../..	../..	FI; K4=E38
					7420	K4/99	../..	../..	FI; K4=E74,E111-E112
					7434	K3/60	K4/35	K2/5	FI; K3=E1-E3,E12-E13; K4=E1,E30,E38; K2=E70
183)	5551	TEST050A	5077	7405					BUT(IN@TR5), IR=(172500); E@@(725)=(0@)
					7477	K3/99	../..	../..	K305=IN@TR5=DECODE
184)	5553	TEST101A	5161	7434					ALU=ZERO, D/DBUF/IR PATH; BUT(IN@TR5) FOR IR=ZERO
					7400	K4/65	K6/20	K3/10	K7/5 K2/5 FI; K4=E33-E34,E42-E43,E81,E59,E82, E95,E100-E101,E109-E110; K6=E73,E82,E91,E96) K3=E1,E43,E81,E71,E83; K7=E52,E54; K2=E70
					7402	K4/80	K6/15	K3/5	FI; K4=E33-E34,E51,E82,E95,E100-E101; K6=E91; K3=E13
					7410	K3/99	../..	../..	FI; K3=E43,E82,E102
					7411	K6/99	../..	../..	FI; K6=E85,E82
					7412	K7/65	K2/30	K3/5	FI; K7=E28,E41,E52,E54,E70; K2=E51,E83; K3=E43
					7417	K6/99	../..	../..	FI; K6=E82
					7420	K4/75	K6/15	K7/5	K5/5 K3/5 K2/5 FI; K4=E31-E32,E30,E40-E41, E47,E55,E66,E74,E79,E92,E102,E111=E112; K6=E65, E73; K7=E41,E48,E65,E70; K5=E14,E54; K3=E82,E84; K2=E34,E05
					7421	K4/80	K6/15	../..	FI; K4=E31,E40,E47,E72,E74,E111-E112; K6=E65
					7422	K7/78	K4/20	K6/5	FI; K7=E6-E7,E18,E23,E31,E45,E48,E65-E67,E69,E74, E78,E91-E92; K4=E31,E40,E72,E74,E111-E112; K6=E43,E66
					7424	K4/80	K6/20	../..	FI; K4=E31,E40,E74,E79,E111-E112; K6=E65
					7426	K6/80	K3/50	../..	FI; K6=E73; K3=E51,E82
					7427	K4/99	../..	../..	FI; K4=E74
					7431	K3/99	../..	../..	FI; K3=E43
					7477	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU=FCN=DECODE
185)	5554	TEST102D	5330	7425					CSP ADDRESS, LOC(01); EMIT/CSP/ALU=B/D/DBUF/IR WITH DATA=(000152)
					7400	K6/99	../..	../..	FI; K4@=4
					7402	K5/99	../..	../..	FI; K6=E44
					7420	K4/60	K6/10	K3/5	K2/5 K5/5 K7/5 FI; K4=E35,E52,E65-E66,E72,

Module codes:	K1/DCS	K2/UWORD	K3/IRDECODE	K4/DATAPATH	K5/KTCACHE	K6/TIMING	K7/STATUS

###	ERROR code	Symbolic label	Line number	ENUA	TNUA	->Module sequence-> #1/%%	#2/%%	#3/%%	Test summary - Print reference + Chip information
	5554	TEST102D	[Continued]						CSP ADDRESS, LOC(01); EMIT/CSP/ALU=B/D/DBUF/IR WITH DATA=(000152)
					7420	K4/60	K6/10	K3/5	E74,E79,E@8-E92,E102,E106,E111-E112; K6=E65,E67, E73,E75; K3=E72; K2=E94,E104; K5=E66,E106; K7=E55-E86
					7434	K4/99	../..	../..	FI; K4=E38,E70,E@8-E89,E98,E108
186)	5561	TEST047A	4986	7412					BUT(IN@TR5), IR=(122566); E@@(325)=(12)
					7400	K3/99	../..	../..	FI; K3=E@@
					7405	K3/99	../..	../..	FI; K3=E@@
					7413	K3/99	../..	../..	FI; K3=E@@
					7416	K3/99	../..	../..	FI; K3=E@@
187)	5563	TEST701A	15925	7402					LOAD BA<15:00>=(052925), READ THRU BA/XT-ALU/PBA/STATUS-MUX(PBA)
					7400	K5/55	K6/15	K7/15	K4/15 K3/5 FI; K5=E3,E7,E9-E11,E13,E25,E27,E30, E32,E37-E39,E43,E45-E47,E50,E58-E60,E65-E67, E69-E70,E72,E79-E82,E93-E94; K6=E19,E25,E29,E33, E36,E38,E49,E54,E59,E70,E76,E90,E96,E104,E108; K7=E48,E58,E64-E65,E73,E80-E82,E88; K4=E29,E39, E48,E57,E65,E87; K3=E9,E14,E33,E64
					7401	K4/45	K3/35	K6/30	FI; K4=E67,E65,E87; K3=E5,E8,E10,E33; K6=E31
					7403	K7/50	K4/50	../..	FI; K7=E80; K4=E39
188)	5564	TEST102C	5303	7412					CSP ADDRESS, LOC(04); EMIT/CSP/ALU=B/D/DBUF/IR WITH DATA=(125200)
					7400	K4/65	K6/15	K2/10	K5/5 K3/5 FI; K4=E35-E36,E45,E51-E56,E59,E66, E72,E79-E80,E82,E89,E91-E92,E95,E98,E100-E102, E105,E108-E110; K6=E73,E75,E82,E84,E91,E93; K2=E05-E96,E104; K5=E53,E63; K3=E56,E95
					7403	K4/75	K6/25	../..	FI; K4=E45,E51,E64; K6=E91
					7405	K4/99	../..	../..	FI; K4=E98
					7420	K3/99	../..	../..	FI; K3=E17,E20,F27,E36
					7432	K3/99	../..	../..	FI; K3=E32
					7434	K4/75	K2/25	../..	FI; K4=E29,E38,E70,E88-F89,E98,E108; K2=E5,E11,E17, E33,E29,E84
189)	5565	TEST624A	15817	7434					CLOCKING D REGISTER PROPOGATED THRU MICROBREAK JAM (P2=T)
					7477	K4/99	../..	../..	K403/4=U=CLKD-CNTRL
190)	5571	TEST624A	15724	4777					MICROBREAK JAM AT MICROADDRESS=(62561); FLAG<0> CLEARED IN JAM WORD
					7777	K3/99	../..	../..	K311=MICROBREAK-LOGIC
191)	5577	TEST103A	5423	7432					CSP ADDRESS, LOC(15); EMIT/CSP/ALU=B/D/DBUF/IR WITH DATA=(000125)
					7411	K2/80	K3/20	../..	FI; K2=E2,E8-E9,E21,E27,E60,E66,E69,E75,E81; K3=E21-E22,E74
					7415	K3/99	../..	../..	FI; K3=E21
					7420	K4/99	../..	../..	FI; K4=E74,E111-E112
192)	5573	TEST623	15680	7402					BUTA(CUA=TRACK) RESETS CUA TO TRACKING MUA VALUE
					4747	K3/99	../..	../..	FI; K3=E91 [Continued]

Module codes:	K1/DCS	K2/UWORD	K3/IRDECODE	K4/DATAPATH	K5/KTCACHE	K6/TIMING	K7/STATUS

###	ERROR code	Symbolic label	Line number	ENUA	TNUA	->Module sequence-> #1/##	#2/##	#3/##	Test summary - Print reference - Chip information
	5673	TEST623	(continued)		BUTA(CUA-TRACK) RESETS CUA IO TRACKING NUA VALUE				
					7400	K2/99	../..	../..	FI; K2=E3,E6,E9,E10,E24-E25,E30,E33,E39,E48,E51, E54,E60,E66,E75,E81,E86-E87,E113
193)	5674	TEST102A		5277	7405				CSP ADDRESS, LOC(10); EXIT/CSP/ALU=B/D/DBUF/IR WITH DATA=(152500)
					7400	K4/65	K6/30	K2/5	K5/5 FI; K4=E36,E38,E42-E43,E51,E89,E92,E99,E98, E105,E108-E110; K6=E81,E84,E91; K2=E96; K5=E63
					7402	K4/65	K6/15	K5/10	K2/10 K3/5 FI; K4=E48,E51,E64,E78,E95,E98, E100-E101; K6=E91,E93; K5=E53; K2=E95; K3=E85
					7403	K2/99	../..	../..	FI; K2=E34
					7412	K4/99	../..	../..	FI; K4=E89,E92,E102,E108
					7434	K4/80	K7/5	K3/5	K2/5 FI; K4=E4,E38,E70,E88-E89,E98-E99,E108; K7=E21; K3=E102; K2=E84
194)	5675	TEST822C		15457	7402				CLR-JAM-ERRORS RESETS STATUS-MUX(JAM)=(001000) AFTER UBREAK JAMUPP
					7407	K7/99	../..	../..	K705=UBRK-JAM-FLAGS,K707=JAM-CLEAR,K708=STATUS-MUX
195)	5676	TEST103C		5449	7425				CSP ADDRESS, LOC(16); EMIT/CSP/ALU=B/D/DBUF/IR WITH DATA=(000152)
					7402	K4/99	../..	../..	FI; K4=E100-E101
					7420	K3/99	../..	../..	FI; K3=E93
196)	5677	TEST822B		15434	7401				BUTR(OTHER-JAM) NEGATED="0" AFTER CLR-JAM-ERRORS
					7403	K7/99	../..	../..	FI; K7=E51-E52,E57
197)	5600	TEST10581		5666	7412				BUT(SR3=0), SR=(000152); SR<3:0>="1010"
					7410	K4/99	../..	../..	FI; K4=E44
					7413	K3/99	../..	../..	FI; K3=E67
					7416	K3/91	../..	../..	FI; K3=E75
198)	5601	TEST822A		15605	7401				UCONS "START-DELAY" & "CLR-NPR-TIMEOUT" DONT EFFECT "CLR-JAM-ERRORS"
					7401	K7/99	../..	../..	FI; K7=E36
199)	5602	TEST610C2		15038	7401				PS(CC)=(01); IR=(072000), CC=ROM=ADDR(037)
					7403	K3/99	../..	../..	FI; K3=E62
					7405	K3/99	../..	../..	FI; K3=E55,E66
200)	5603	TEST621F		15521	7402				PRDC-MUX(CUA=PORT)=(055226); CUA-LOCKED, EXFLAGS SET, PREFETCH=JAM CLEAR
					7400	K2/99	../..	../..	FI; K2=E46,E56,E90,E100,E108
201)	5604	TEST374A2		9959	7434				A/B SP WRITE FCN WR(A,LO,A=ADDR) AND WR(A,LO,B=ADDR)
					7400	K4/80	K3/20	../..	FI; K4=E11,E17,E23,E47; K3=E34
					7402	K4/75	K3/25	../..	FI; K4=E17,E26; K3=E91
					7420	K4/70	K3/30	../..	FI; K4=E10,E17,E21-E24,E26; K3=E32,E38,E44,E50,E53
					7417	K4/99	../..	../..	K405=SP-REWRITE-CNTL,K406/7=A/B-SPADS
202)	5605	TEST621E		15495	7402				STATUS-MUX(JAM-PORT)=(001001) AFTER MICROBREAK JAM
					7400	K2/75	K7/25	../..	FI; K2=E56,E76,E83,E100; K7=E56,E73

Module codes: K1/DCS K2/UWORD K3/IRDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

###	ERROR code	Symbolic label	Line number	ENUA	TNUA	->Module sequence-> #1/##	#2/##	#3/##	Test summary - Print reference - Chip information
203)	5606	TEST374B2		10032	7434				A/B SP WRITE FCN WR(A,HI,A=ADDR) AND WR(A,HI,B=ADDR)
					7417	K4/99	../..	../..	K405=SP-REWRITE-CNTL,K406/7=A/B-SPADS
204)	5607	TEST621D		15476	7403				BUTR(OTHER-JAM) ASSERTED="1" AFTER MICROBREAK JAM
					7401	K7/70	K3/30	../..	FI; K7=E32,E38,E57,E73; K3=E44
205)	5610	TEST105A1		5619	7405				BUT(SR3=0), SR=(000125); SR<3:0>="0101"
					7407	K3/99	../..	../..	FI; K3=E77
					7416	K3/99	../..	../..	FI; K3=E76
					7425	K3/99	../..	../..	FI; K3=E48
206)	5611	TEST621C		15458	7406				BUTM(INIT-JAM) STAYS NEGATED AFTER MICROBREAK JAM
					7407	K2/99	../..	../..	FI; K2=E46
207)	5612	TEST610B2		14918	7406				PS(CC)=(06); IR=(005300), CC=ROM=ADDR(253)
					7402	K3/99	../..	../..	FI; K3=E85,E89,E86
					7404	K3/85	K4/45	../..	FI; K3=E85,E65; K4=E78
					7414	K3/99	../..	../..	FI; K3=E55
208)	5613	TEST621A		15440	7402				CLOCKING D-REGISTER PROPAGATED THRU MICROBREAK JAM (P3-7)
					7427	K4/99	../..	../..	K403/4=U-CLKD-CNTRL
209)	5614	TEST376A1		10619	7434				SP/SP-ADDRESS: FLTPT/BIT02="0"; R-IDR=1/BIT00="1" FORCED
					7400	K3/40	K2/40	K4/25	FI; K3=E50,E57,E94,E97; K2=E55,E108,E112; K4=E4,E9, E34,E101
					7410	K2/65	K3/35	../..	FI; K2=E95,E110; K3=E47,E96
210)	5615	ERROR621A		15426	0000				MICROBREAK JAMUPP AT (5522) ATTEMPTED; DID NOT OCCUR
					5615	K3/75	K2/10	K5/10	K6/5 K7/5 FI; K3=E32,E41,E52,E61,E81,E91,E101, E111-E112; K2=E35,E41,E70,E77; K5=E60-E61,E88, E77; K6=E39-E40; K7=E48,E57
211)	5616	TEST113B		16640	7402				DATI-NOINT, ILLEGAL INTERNAL ADDR; JAM=(001040)
					7400	K6/70	K7/15	K5/15	FI; K6=E46,E51; K7=E74,E88; K5=E38,E72
212)	5617	TEST621A		15344	4777				MICROBREAK JAM AT MICROADDRESS=(5522); ACTIVE-BUT, WR-CSP LATCHES CLEARED
					4747	K3/99	../..	../..	FI; K3=E91,E111
213)	5620	TEST105E		5743	7415				SR LOAD/READ, SR=(152500); SR<15:12>="1101"
					7405	K5/99	../..	../..	FI; K5=E80
					7411	K4/75	K6/10	K5/5	K2/5 FI; K4=E33-E34,E39,E45-E46,E51,E54,E78,E95, E98,E100-E101; K6=E91,E93; K5=E53; K2=E95
					7414	K4/85	K6/10	K5/5	K2/5 FI; K4=E33-E34,E36,E39,E45-E46,E51,E62,E80, E82,E95,E98,E100-E101; K6=E9),E93; K5=E53; K2=E95

Module codes: K1/DCS K2/UWORD K3/IRDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

###	ERROR code	Symbolic label	Line number	ENUA	TNUA	->Module sequence-> #1/##	#2/##	#3/##	Test summary - Print reference - Chip information
214)	5621	TFST620C	15309	7402					STATUS=NUX(JAM-PORT)=(001000) AFTER CLR-JAM-ERRORS UCON
					7400	../..	../..	../..	!! MOS BATTERY BACKUP MUST BE GOOD !!
					7400	K7/95	K2/5	K6/3	FI; K7=E48-E49,E53,E56-E58,E64-E66,E73,E80-E82, E88; K2=E37,E44,E66,E70; K6=E51
					7401	K7/85	K2/15	../..	FI; K7=E28,E37; K2=E100
					7403	K7/99	../..	../..	FI; K7=E80
215)	5622	TFST610A2	14806	7406					PS[CC]=(06); IR=(105200), CC=ROM-ADDR(145)
					7404	K3/90	K4/10	../..	FI; K3=E58,E62,E64,E72,E85; K4=E72
					7407	K3/99	../..	../..	FI; K3=E62,E72
216)	5623	TFST620B	15286	7401					BUTR(OTHER-JAM) NEGATED=0 AFTER CLR-JAM-ERRORS UCON
					7403	K7/80	K6/20	K3/5	FI; K7=E18,E26,E32,E36,E49,E57-E58,E74; K6=E2-E3, E62,E64,E105; K3=E44
217)	5624	TFST610D2	15145	7410					PS[CC]=(10); IR=(072000), CC=ROM-ADDR(116)
					7414	K3/99	../..	../..	FI; K3=E63,E66
218)	5625	TFST3758	10563	7432					DAD/3 CAUSES BYTE-WRITE(LO) TO BSPLO; DAD/1 OR DAD/2 NOT
					7400	K3/99	../..	../..	FI; K3=E46
					7402	K4/99	../..	../..	FI; K4=E99-E100
219)	5627	TEST610A1	14744	7405					PS[CC]=(05); IR=(105200), CC=ROM-ADDR(685)
					7400	K3/99	../..	../..	FI; K3=E57
					7401	K3/70	K2/30	../..	FI; K3=E55,E72; K2=E108
					7404	K3/98	../..	../..	FI; K3=E62,E72
					7406	K3/99	../..	../..	FI; K3=E62,E66,E85
					7407	K3/60	K4/20	../..	FI; K3=E34,E55,E62,E66,E72; K4=E72
					7411	K3/99	../..	../..	FI; K3=E34,E55,E66
					7412	K3/98	../..	../..	FI; K3=E8,E8-E10,E14,E36,E72
					7415	K3/99	../..	../..	FI; K3=E55,E62-E63,E72
220)	5631	TEST046A	4920	7426					BUT(INSTR5), IR=(000200); E78(E00)=(26)
					6000	K2/99	../..	../..	FI; K2=E47
					7426	K2/99	../..	../..	FI; K2=E25
					7427	K3/99	../..	../..	K305=INSTR5-DECODE
221)	5633	TFST045A	4870	7405					BUT(DMO&SMO&BYTE), IR=(004300); DMO, -SMO, BYTE (SWAB)
					7404	K3/99	../..	../..	FI; K3=E95
222)	5635	TEST044A	4833	7447					BUT(INSTR1), IR=(010250), CLASS-A(IFC)
					7417	K3/99	../..	../..	FI; K3=E90,E99
					7464	K3/99	../..	../..	FI; K3=E90,E110

Module codes: K1/DCS K2/UWORD K3/IRDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

###	ERROR code	Symbolic label	Line number	ENUA	TNUA	->Module sequence-> #1/##	#2/##	#3/##	Test summary - Print reference - Chip information
223)	5636	TEST105D	5716	7405					SR LOAD/READ, SR=(152500); ALU-A/D/DBUF/IR PATH
					7400	K4/99	../..	../..	FI; K4=E27,E38-E36,E39,E42-E43,E48,E67
					7402	K2/95	K4/5	../..	FI; K2=E6,E9,E12,E18,E26,E30,E33,E39,E45,E51,E54, E60,E66,E89,E75,E86-E88,E94,E99,E103; K4=E77
224)	5637	TEST043B	4796	7711					BUT(INSTR1), IR=(161707); CLASS-G(DOP*-SMO)
					5126	K6/99	../..	../..	FI; K6=E42
					7710	K3/99	../..	../..	FI; K3=E105
					7717	K3/99	../..	../..	FI; K3=E107
					7750	K3/99	../..	../..	FI; K3=E90
225)	5641	TEST043A	4767	7404					BUT(DMO&SMO&BYTE), IR=(161707); DMO, -SMO, -BYTE (SUB)
					7400	K3/99	../..	../..	FI; K3=E74
					7405	K3/99	../..	../..	FI; K3=E109
226)	5642	TEST105C	5690	7412					SR LOAD/READ, SR=(125200); ALU-A/D/DBUF/IR PATH
					7400	K4/99	../..	../..	FI; K4=E27,E32-E37,E39,E41-E43,E45-E46,E48,E60,E62, E67-E68,E72
					7403	K4/99	../..	../..	FI; K4=E13-E34,E39,E45-E46,E54,E64,E77-E78
					7405	K4/99	../..	../..	FI; K4=E45-E46
					7420	K4/99	../..	../..	FI; K4=E20
					7434	K4/99	../..	../..	FI; K4=E29
227)	5643	TEST042B	4731	7714					BUT(INSTR1), IR=(144020); CLASS-G(DOP*-SMO)
					7417	K3/99	../..	../..	FI; K3=E90,E110
					7514	K3/99	../..	../..	FI; K3=E99
					7614	K3/99	../..	../..	FI; K3=E100
					7710	K3/99	../..	../..	FI; K3=E106,E120
					7715	K2/99	../..	../..	FI; K2=E77
					7717	K3/99	../..	../..	FI; K3=E97,E105-E107
228)	5645	TEST042A	4703	7401					BUT(DMO&SMO&BYTE), IR=(144020); -DMO, -SMO, BYTE
					7400	K3/70	K2/30	../..	FI; K3=E84,E60,E65,E85; K2=E19,E41,E77-E78
					7403	K5/50	K3/50	../..	FI; K5=E98; K3=E109
					7408	K2/99	../..	../..	FI; K2=E78
229)	5646	TEST104B	8543	7432					CSP BASCON ADDR, LOC(16); EMIT/CSP/ALU=B/D/DBUF/IR WITH DATA=(000125)
					7400	K4/99	../..	../..	FI; K4=E30
					7402	K4/99	../..	../..	FI; K4=E31,E100-E101
2301	5647	TFST041B	4667	7517					BUT(INSTR1), IR=(120777); CLASS-B(DOP*-MOV*SMO*-DMO)
					7617	K3/99	../..	../..	FI; K3=E118
2311	5651	TEST041A	4638	7412					BUT(IR15-12), IR=(120777); IR<15:12>="1010"
					7042	K2/99	../..	../..	FI; K2=E1

Module codes: K1/DCS K2/UWORD K3/IRDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

###	EPROM code	Symbolic label	Line number	ENDA	TNUA	->Module sequence-> #1/##	#2/##	#3/##	Test summary - Print reference - Chip information
232)	5652	TEST105B	5642	7425					SR LOAD/READ, SR=(000152); ALU=A/D/DBUF/IR PATH
					7420	K4/70	K3/30	../..	FI; K4=E31-E32,E35,E37,E40-E41,E44,E52-E53,E57,E61, E75-E76; K3=E43,E51,E74,E76-E77,E82,E102
					7432	K4/99	../..	../..	FI; K4=E55
233)	5653	TEST040B	4802	7517					BUT(INSTR1), IR=(050777), CLASS=B(DOP=-MOV=SNO=-DNO)
					7417	K3/99	../..	../..	FI; K3=E80
234)	5655	TEST040A	4574	7405					BUT(IR15-12), IR=(050777); IR<15:12>="0101"
					7401	K3/99	../..	../..	FI; K3=E121
235)	5657	TEST037A	4531	7446					BUT(INSTR1), IR=(060205), CLASS=A(DOP=SNO=DNO)
					7417	K3/35	K3/35	K2/35	FI; K7=E17; K3=E115; K2=E73
					7447	K3/99	../..	../..	FI; K3=E94
236)	5661	TEST036A	4486	7455					BUT(INSTR1), IR=(150506), CLASS=A(DOP=SNO=DNO)
					7457	K3/99	../..	../..	FI; K3=E94
237)	5663	TEST035A	4442	7443					BUT(INSTR1), IR=(030701), CLASS=A(DOP=SNO=DNO)
					7417	K3/99	../..	../..	FI; K3=E60,E80,E88,E90,E95,E97,E105,E107,E110, E115-E117,E120
					7441	K3/45	K6/30	K5/30	FI; K3=E43,E94,E107; K6=E91-E92; K5=E87,E98
					7442	K3/99	../..	../..	FI; K3=E94,E105
					7460	K3/99	../..	../..	FI; K3=E90
					7540	K3/99	../..	../..	FI; K3=E120
238)	5665	TEST034A	4398	7511					BUT(INSTP1), IR=(005112), CLASS=B(SOP=-DNO)
					7517	K3/99	../..	../..	FI; K3=E88
					7751	K3/99	../..	../..	FI; K3=E98
239)	5667	TEST033A	4354	7517					BUT(INSTR1), IR=(106274), CLASS=B(SOP=-DNO)
					7417	K3/99	../..	../..	FI; K3=E80,E100,E108,E120
					7515	K2/99	../..	../..	FI; K2=E114
					7562	K3/99	../..	../..	FI; K3=E108
					7716	K3/99	../..	../..	FI; K3=E100
240)	5670	TEST047B	5019	7413					BUT(IP11=FLOAT), IR=(122566); FP=ROM(534)=(13)
					7477	K3/99	../..	../..	K306=FP=INSTR=DECODE
241)	5671	TEST032B	4317	7612					BUT(INSTR1), IR=(110126), CLASS=C(MOV=SNO=-DNO)
					7451	K3/99	../..	../..	FI; K3=E108
					7602	K3/99	../..	../..	FI; K3=E4,E6-E7,E11,E15,E43,E118
					7752	K3/99	../..	../..	FI; K3=E85,E94,E104

Module codes: K1/DCS K2/UNIBD K3/IRDECODE K4/DATAPATH K5/MTCACHE K6/TIMING K7/STATUS

###	EPROM code	Symbolic label	Line number	ENDA	TNUA	->Module sequence-> #1/##	#2/##	#3/##	Test summary - Print reference - Chip information
242)	5673	TEST032A	4288	7412					BUT(MOV/DR?=IR5=3), IR=(110126); IR<5:3>="010", MOV
					7402	K3/80	K5/20	../..	FI; K3=E57,E97; K5=E69
243)	5674	TEST103D	5476	7405					CSP ADDRESS, LOC(07); EMIT/CSP/ALU=B/D/DBUF/IR WITH DATA=(152500)
					7322	K3/99	../..	../..	FI; K3=E22
					7400	K4/99	../..	../..	FI; K4=E59,E109-E110
244)	5675	TEST031B	4253	7604					BUT(INSTR1), IR=(010242), CLASS=C(MOV=SNO=-DNO)
					7404	K3/99	../..	../..	FI; K3=E98
					7417	K3/80	K5/15	../..	FI; K3=E87,E90,E113,E119; K5=E98
					7514	K3/55	K6/30	K5/15	FI; K3=E115-E117; K6=E91-E92; K5=E87
					7607	K3/99	../..	../..	FI; K3=E98
					7614	K3/99	../..	../..	FI; K3=E97,E118
					7640	K3/99	../..	../..	FI; K3=E90
					7700	K3/99	../..	../..	FI; K3=E99,E110
					7704	K3/99	../..	../..	FI; K3=E98,E120
245)	5677	TEST031A	4224	7402					BUT(DNO&SNO=BYTE), IR=(010242); -DNO, SNO, -BYTE
					7008	K3/99	../..	../..	FI; K3=E56
					7400	K3/99	../..	../..	FI; K3=E47,E56,E60,E80,E90,E109,E119-E120
					7406	K3/99	../..	../..	FI; K3=E109
246)	5701	TEST030A	4117	7552					BUT(INSTR1), IR=(005204), CLASS=D(SOP=DNO)
					7313	K3/99	../..	../..	FI; K3=E2
					7553	K3/99	../..	../..	FI; K3=E87
247)	5703	TEST027A	4074	7561					BUT(INSTR1), IR=(106102), CLASS=D(SOP=DNO)
					????	K3/99	../..	../..	K306/8=INSTR1=DECODE
248)	5705	TEST026A	4029	7574					BUT(INSTR1), IR=(105403), CLASS=D(SOP=DNO)
					7570	K3/99	../..	../..	FI; K3=E100
					7574	K2/99	../..	../..	FI; K2=E67
					7757	K6/99	../..	../..	FI; K6=E92
249)	5707	TEST025A	3986	7543					BUT(INSTR1), IR=(006303), CLASS=D(SOP=DNO)
					7540	K3/99	../..	../..	FI; K3=E99
					7557	K3/99	../..	../..	FI; K3=E105
250)	5710	TEST014D	3436	7401					BUT(IR11=FLOAT), IR=(000125); FP=ROM(020)=(01)
					7400	K3/99	../..	../..	FI; K3=E86
251)	5711	TEST024B	3949	7557					BUT(INSTR1), IR=(005706), CLASS=D(SOP=DNO)
					5125	K3/99	../..	../..	FI; K3=E12
					7306	K3/99	../..	../..	FI; K3=E5
					7407	K3/99	../..	../..	FI; K3=E44

(Continued)

```
     ERROR                          Line             ->Module sequence->
###) code  Symbolic label         number  ENUA  TNUA  #1/%%  #2/%%  #3/%%  Test summary - Print reference - Chip information
---  ----- ---------------        ------  ----  ----  -----  -----  -----  -----------------------------------------------------

     5711  TEST024B               [Continued]  BUT(INSTR1), IR=(005706), CLASS=D(SOP*DMO)
                                          7417  K3/99  ../..  ../..  FI; K3=E80,E88,E108,E117
                                          7547  K3/99  ../..  ../..  FI; K3=E118
                                          7580  K3/99  ../..  ../..  FI; K3=E120
                                          7583  K3/99  ../..  ../..  FI; K3=E106
                                          7585  K3/99  ../..  ../..  FI; K3=E103,E107
                                          7556  K3/99  ../..  ../..  FI; K3=E103,E105
                                          7577  K3/99  ../..  ../..  FI; K3=E108

252) 5712  TEST030B               4147  7424  BUT(IR11!FLOAT), IR=(005204); FP-ROM(240)=(04)
                                          7420  K3/99  ../..  ../..  FI; K3=E86
                                          7425  K4/60  K2/20  K3/20  FI; K4=E32,E41-E43,E59,E66,E71,E79,E82,E92,E102,
                                                                         E109-E110; K2=E96,E101,E104,E110,E119; K3=E12,
                                                                         E43,E52,E71,E82,E96,E102
                                          7426  K3/99  ../..  ../..  FI; K3=E86

253) 5713  TEST030C               4166  7403  BUTR(IR11), IR=(005204); IR11="1"
                                          7401  K3/99  ../..  ../..  K307=MICROBRANCH

254) 5715  TEST024A               3920  7402  BUTR(DR6/TL), IR=(005706); DR6
                                          7403  K3/99  ../..  ../..  FI; K3=E93

255) 5717  TEST023A               3877  7560  BUT(INSTR1), IR=(106004), CLASS=D(SOP*DMO)
                                          8122  K3/99  ../..  ../..  FI; K3=E14,E23
                                          5401  K3/99  ../..  ../..  FI; K3=E1
                                          7401  K3/99  ../..  ../..  FI; K3=E19
                                          7417  K3/99  ../..  ../..  FI; K3=E47,E60,E74,E80,E88,E90,E99,E104,E108-E110,
                                                                         E116,E119-E120
                                          7460  K3/99  ../..  ../..  FI; K3=E100
                                          7510  K3/99  ../..  ../..  FI; K3=E109
                                          7520  K3/99  ../..  ../..  FI; K3=E100
                                          7540  K3/99  ../..  ../..  FI; K3=E85,E108,E110
                                          7557  K3/99  ../..  ../..  FI; K3=E97,E105-E108,E118
                                          7561  K3/85  K4/15  ../..  FI; K3=E79,E87,E103,E105; K4=E9
                                          7570  K3/99  ../..  ../..  FI; K3=E76,E118
                                          7760  K3/99  ../..  ../..  FI; K3=E96,E104,E114,E119

256) 5721  TEST022A               3833  7757  BUT(INSTR1), IR=(003063), CLASS=F(BRANCH)
                                          0000  K3/99  ../..  ../..  FI; K3=E6
                                          5800  K3/99  ../..  ../..  FI; K3=E11
                                          7417  K3/60  K7/40  ../..  FI; K3=E95,E104; K7=E13,E101
                                          7517  K3/99  ../..  ../..  FI; K3=E28

257) 5722  TEST015C               3519  7655  BUT(INSTR1), IR=(000162), CLASS=E(JMP)
                                          7651  K3/99  ../..  ../..  FI; K3=E103,E106
                                          7654  K3/99  ../..  ../..  FI; K3=E93,E106
                                          7657  K3/99  ../..  ../..  FI; K3=E69,E89,E103
```



```
     ERROR                          Line             ->Module sequence->
###) code  Symbolic label         number  ENUA  TNUA  #1/%%  #2/%%  #3/%%  Test summary - Print reference - Chip information
---  ----- ---------------        ------  ----  ----  -----  -----  -----  -----------------------------------------------------

258) 5723  TEST021B               3775  7757  BUT(INSTR1), IR=(002315), CLASS=F(BRANCH)
                                          7417  K3/99  ../..  ../..  FI; K3=E89,E96,E104,E109
                                          7751  K3/99  ../..  ../..  FI; K3=E116

259) 5725  TEST021A               3746  7400  BUT(DMO!SMO!BYTE), IR=(002315); -DMO, -SMO, -BYTE
                                          7401  K3/99  ../..  ../..  FI; K3=E54,E85,E87-E88,E116
                                          7402  K3/99  ../..  ../..  FI; K3=E56,E109-E110,E114,E119
                                          7403  K3/50  K6/25  K5/25  FI; K3=E112,E114; K6=E83; K5=E59
                                          7404  K3/99  ../..  ../..  FI; K3=E74,E90,E109,E119-E120
                                          7406  K7/99  ../..  ../..  FI; K7=E20
                                          7600  K2/99  ../..  ../..  FI; K2=E36
                                          7606  K2/99  ../..  ../..  FI; K2=E15

260) 5726  TEST050B               5108  7412  BUT(INSTR5), IR=(175200); E88(752)=(12)
                                          7434  K7/99  ../..  ../..  FI; K7=E13,E28,E94,E101
                                          7477  K3/99  ../..  ../..  K305=INSTR5-DECODE

261) 5727  TEST020C               3711  7757  BUT(INSTR1), IR=(001257), CLASS=F(BRANCH)
                                          7417  K3/99  ../..  ../..  FI; K3=E95-E96,E104,E114
                                          7740  K3/99  ../..  ../..  FI; K3=E99
                                          7755  K3/99  ../..  ../..  FI; K3=E117

262) 5730  TEST047C               5039  7406  BUT(MOV/DR7!IR5-3), IR=(122566); IR<5:3>="110", -MOV
                                          7407  K3/99  ../..  ../..  K308=MICROBRANCH

263) 5731  TEST020B               3689  7405  BUT(MOV/DR7!IR6-3), IR=(001257); IR<5:3>="101", -MOV
                                          7407  K3/99  ../..  ../..  FI; K3=E84
                                          7415  K3/99  ../..  ../..  FI; K3=E94,E120

264) 5733  TEST020A               3662  7412  BUT(IR9-6), IR=(001257); IR<9:6>="1010"
                                          7232  K3/99  ../..  ../..  FI; K3=E88
                                          7402  K3/70  K6/30  ../..  FI; K3=E76,E83,E85-E86,E89; K6=E82-E83
                                          7413  K6/99  ../..  ../..  FI; K6=E73

265) 5734  TEST030D               4189  7405  BUT(INSTR5), IR=(005204); E88(453)=(05)
                                          7400  K3/99  ../..  ../..  FI; K3=E88
                                          7403  K3/99  ../..  ../..  FI; K3=E88
                                          7407  K3/99  ../..  ../..  FI; K3=E88
                                          7415  K3/99  ../..  ../..  FI; K3=E88
                                          7417  K3/99  ../..  ../..  FI; K3=E88
                                          7425  K3/99  ../..  ../..  FI; K3=E60
                                          7426  K3/99  ../..  ../..  FI; K3=E38,E78,E109

                                          7437  K3/99  ../..  ../..  FI; K3=E38
                                          7477  K3/99  ../..  ../..  K305=INSTR5-DECODE
```

###	FPDR code	Symbolic label	Line number	ENUA	TNUA	->Module sequence-> #1/##	#2/##	#3/##	Test summary - Print reference - Chip information
266)	5735	TEST017P	3626	7757	BUT(IN&TR1), IR=(000522), CLASS-F(BRANCH)				
					7417	K3/89	../..	../..	FI; K3=E109-E110
					7652	K3/99	../..	../..	FI; K3=E114
					7717	K3/89	../..	../..	FI; K3=E89
					7752	K3/99	../..	../..	FI; K3=E96
267)	5737	TEST017A	3598	7405	BUT(IR9-6), IR=(000522); IR<9:6>="0101"				
					7401	K3/80	K5/10	../..	FI; K3=E80,E79,E83-E86,E88,E114; K5=E98
					7405	K3/89	../..	../..	FI; K3=E23
					7418	K6/65	K5/35	../..	FI; K6=E82-E83; K5=E59
268)	5741	TEST016A	3585	7757	BUT(IN&TR1), IR=(100000), CLASS-F(BRANCH)				
					7400	K3/99	../..	../..	FI; K3=E51
					7417	K3/75	K6/15	K5/10	FI; K3=E41,E95,E98,E110,E113; K6=E91-E92; K5=E98
					7513	K6/99	../..	../..	FI; K6=E74
					7557	K3/99	../..	../..	FI; K3=E99
					7657	K3/99	../..	../..	FI; K3=E58,E70,E100
					7717	K3/99	../..	../..	FI; K3=E100
					7760	K3/89	../..	../..	FI; K3=E100
					7770	K3/89	../..	../..	FI; K3=E95,E110,E115,E119
					7777	K3/99	../..	../..	FI; K3=E108
269)	5743	TEST015A	3473	7425	BUT(IN&TR5), IR=(000152); E78(1BI)=(25)				
					5075	K3/99	../..	../..	FI; K3=E10
					5673	K3/99	../..	../..	FI; K3=E3
					7305	K3/99	../..	../..	FI; K3=E20
					7400	K3/89	../..	../..	FI; K3=E33,E106
					7410	K4/80	K3/35	K6/15	FI; K4=E12,E14; K3=E75,E103; K6=E65
					7424	K3/89	../..	../..	FI; K3=E67,E78,E88
					7434	K3/99	../..	../..	FI; K3=E103,E111,E114
					7435	K3/89	../..	../..	FI; K3=E76,E78
					7477	K3/99	../..	../..	K308=IN&TR5-DECODE
					7655	K3/89	../..	../..	FI; K3=E36
270)	5745	TEST014A	3368	7432	BUT(IN&TR4), IR=(000125); E78(125)=(32)				
					7400	K3/89	../..	../..	FI; K3=E88
					7420	K3/50	K6/25	K5/20	K2/5 FI; K3=E66-E67,E77-E78,E84-E88,E93,E101, E103,E112; K6=E65-E66,E73-E74,E83,E92; K5=E67, E70,E106; K2=E112
					7422	K3/99	../..	../..	FI; K3=E78,E88
					7430	K3/99	../..	../..	FI; K3=E78,E88
					7434	K2/85	K7/15	K6/15	K3/15 FI; K2=E33,E71,E102,E108; K7=E17; K6=E91; K3=E43
					7436	K3/99	../..	../..	FI; K3=E75,E78
					7477	K3/99	../..	../..	K309=IN&TR5-DECODE (Conti

Module codes: K1/DCS K2/UWORD K3/IRDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

###	FPDR code	Symbolic label	Line number	ENUA	TNUA	->Module sequence-> #1/##	#2/##	#3/##	Test summary - Print reference - Chip information
	5745	TEST014A	(Continued)		BUT(IN&TR5), IR=(000125); E78(125)=(32)				
					7673	K3/99	../..	../..	FI; K3=E98
271)	5746	SFDFT08R	10415	7777	(NUA SEQUENCING LOGIC ERROR)				
					7777	K1/99	../..	../..	INTERNAL DC5 ERROR
272)	5747	TEST013C	3331	7417	BUT(IN&TR1), IR=(000000), CLASS-OTHER(NOT-A-THRU-G)				
					7400	K3/89	../..	../..	FI; K3=E97,E115
					7407	K3/89	../..	../..	FI; K3=E76,E87,E116
					7410	K3/89	../..	../..	FI; K3=E97-E98
					7413	K3/89	../..	../..	FI; K3=E75,E106
					7415	K3/99	../..	../..	FI; K3=E77,E107
					7416	K3/99	../..	../..	FI; K3=E67,E105
					7437	K3/89	../..	../..	FI; K3=E68,E108
					7440	K3/99	../..	../..	FI; K3=E90,E95,E99,E113,E115-E117,E121
					7457	K3/89	../..	../..	FI; K3=E70,E97,E100
					7460	K3/89	../..	../..	FI; K3=E47,E57,E90,E94,E97,E108,E110
					7510	K3/99	../..	../..	FI; K3=E120
					7540	K3/99	../..	../..	FI; K3=E108
					7650	K3/99	../..	../..	FI; K3=E96,E98
					7710	K3/99	../..	../..	FI; K3=E110
					7757	K3/99	../..	../..	FI; K3=E89,E110
					7776	K2/99	../..	../..	FI; K2=E32
273)	5750	TEST013F	3312	7434	BUT(IN&TR5), IR=(000000); E78(000)=(34)				
					7400	K3/85	K4/5	K2/5	FI; K3=E38,E59,E68,E88-E89,E96,E98-E99,E104-E107, E109-E110,E113-E114,E116-E119; K4=E4; K2=E25
					7402	K3/99	../..	../..	FI; K3=E104,E106,E113,E115-E117
					7414	K3/89	../..	../..	FI; K3=E89
					7420	K3/99	../..	../..	FI; K3=E78
					7421	K2/85	K1/35	K5/20	FI; K2=E91,E111; K3=E78,E93; K5=E94
					7422	K2/80	K3/30	K5/20	FI; K2=E91,E111; K3=E78,E93; K5=E94
					7424	K3/55	K2/35	K5/15	FI; K3=E76,E78,E88,E93; K2=E91,E103; K5=E94
					7426	K3/99	../..	../..	FI; K3=E78
					7427	K2/99	../..	../..	FI; K2=E71
					7430	K3/99	../..	../..	FI; K3=E75,E78,E88
					7435	K3/89	../..	../..	FI; K3=E67,E78
					7436	K3/99	../..	../..	FI; K3=E77-E78
					7437	K3/99	../..	../..	FI; K3=E78
274)	5751	TEST013F	3291	7400	BUT(MOV/DAT&IR5=3), IR=(000000); IR<5:3>="000", -MOV				
					7401	K2/80	K3/30	K5/15	K6/8 FI; K2=E84,E91,E103; K3=E67,E93; K5=E94; K6=E69
					7402	K3/45	K2/35	K5/15	K6/5 FI; K3=E77,E103,E107,E109; K2=E91,E114; K5=E105; K6=E73
					7404	K2/45	K3/35	K5/20	FI; K2=E91,E114; K3=E75,E103; K5=E105
					7410	K3/99	../..	../..	FI; K3=E47,E57,K76,E113,E115-E117,E119-E120

Module codes: K1/DCS K2/UWORD K3/IRDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

```
      ERROR                            Line              ->Module sequence->
###)  code  Symbolic label           number  ENUA  THUA  #1/%%  #2/%%  #3/%%  Test summary - Print reference - Chip information
---   ----- ---------------          ------  ----  ----  -----  -----  -----  -------------------------------------------------

275)  5752  TEST014B                  3394  7403  BUTR(DR6/7L), IR=(000125); DRB=-(DR6+7)
                                                  7402  K3/99  ../..  ../..  FI; K3=E37-E39,E64,E66,E79

276)  5753  TEST013D                  3272  7400  BUT(IR9-6), IR=(000000); IR<9:6>="0000"
                                                  7401  K3/99  ../..  ../..  FI; K3=E67
                                                  7402  K3/99  ../..  ../..  FI; K3=E77,E103
                                                  7404  K3/99  ../..  ../..  FI; K3=E78
                                                  7410  K3/99  ../..  ../..  FI; K3=E76

277)  5754  TEST621H                 15572  7402  CHECK "WR-CSP" LATCH CLEARED IN JAMMED U-WORD, WRITE NOT DONE
                                                  7401  K7/65  K3/35  ../..  FI; K7=E25,E41; K3=E32

278)  5755  TEST013C                  3253  7401  BUTR(IR11), IR=(000000); IR11="0"
                                                  7401  K3/99  ../..  ../..  FI; K3=E39,E86

279)  5756  TEST624D                 15844  7403  BUTR(PREFETCH+JAM) ASSERTED="1", GETS PREFETCH-H AT JAMUPP="1"
                                                  7407  K3/99  ../..  ../..  K3E8=PREFETCH+JAM-LOGIC
                                                  7747  K3/99  ../..  ../..  FI; K3=E17

280)  5757  TEST013B                  3233  7400  BUT(IR11+FLOAT), IR=(000000); IR<11>="0", FP+ROM(000)=(00)
                                                  7401  K2/45  K3/30  K5/15  K6/8   FI; K2=E97,E113; K3=E67,E86,E103; K5=E105;
                                                                                    K6=E66,E73
                                                  7402  K2/48  K3/35  K5/15  K6/8   FI; K2=E97-E98; K3=E77,E86,E114; K5=E97;
                                                                                    K6=E82
                                                  7404  K2/50  K3/35  K5/15  FI; K2=E97-E98; K3=E78,E86,E114; K5=E97
                                                  7410  K2/48  K3/40  K5/15  FI; K2=E97,E107; K3=E76,E86,E114; K5=E97
                                                  7417  K3/99  ../..  ../..  FI; K3=E86
                                                  7420  K2/50  K3/25  K5/15  K6/10  FI; K2=E97,E107; K3=E68,E114; K5=E97; K6=E82

281)  5761  TEST013A                  3205  7400  BUT(IR15-12), IR=(000000); IR<15:12>="0000"
                                                  7401  K2/55  K3/25  K5/20  FI; K2=E85,E102; K3=E67,E113; K5=E95
                                                  7402  K2/50  K3/30  K5/20  FI; K2=E85,E102; K3=E77,E113; K5=E95
                                                  7404  K2/45  K3/35  K5/20  FI; K2=E85,E108; K3=E78,E113; K5=E95
                                                  7405  K2/99  ../..  ../..  FI; K2=E102-E103,E108,E111,E114
                                                  7410  K2/45  K3/30  K5/15  K6/10  FI; K2=E85,E108; K3=E76,E113; K5=E95; K6=E91
                                                  7417  K2/60  K5/20  K7/10  K6/5   K4/5   FI; K2=E1,E17,E23,E34,E55-E56,E62,
                                                                                    E70-E71,E76,E82-E83,E85,E92,E100-E101,E105;
                                                                                    K5=E64,E84,E94-E95,E97,E105; K7=E28,E37,E54;
                                                                                    K6=E101; K4=E26,E29,E38,E87

282)  5762  TEST014C                  3414  7652  BUT(INSTR1), IR=(000125), CLASS=E(JMP)
                                                  7417  K3/99  ../..  ../..  FI; K3=E96,E98,E103,E107
                                                  7452  K3/99  ../..  ../..  FI; K3=E58,E70,E99
                                                  7612  K3/99  ../..  ../..  FI; K3=E100
                                                  7640  K3/99  ../..  ../..  FI; K3=E90                           [Continued]
```



```
      ERROR                            Line              ->Module sequence->
###)  code  Symbolic label           number  ENUA  THUA  #1/%%  #2/%%  #3/%%  Test summary - Print reference - Chip information
---   ----- ---------------          ------  ----  ----  -----  -----  -----  -------------------------------------------------

      5762  TEST014C               (Continued)    BUT(INSTR1), IR=(000125), CLASS=E(JMP)
                                                  7642  K3/99  ../..  ../..  FI; K3=E90,E98,E119
                                                  7650  K3/99  ../..  ../..  FI; K3=E103,E107
                                                  7653  K3/99  ../..  ../..  FI; K3=E58,E70,E93,E105,E109
                                                  7656  K3/99  ../..  ../..  FI; K3=E69,E75,E89,E103,E106,E109
                                                  7657  K3/99  ../..  ../..  FI; K3=E98,E105-E107
                                                  7740  K3/99  ../..  ../..  FI; K3=E100
                                                  7752  K3/99  ../..  ../..  FI; K3=E99,E120

283)  5763  TEST012G                  3169  7474  BUT(INSTR1), IR=(177777), CLASS-A(FLOATING-POINT), FLAG<4:5>="00"
                                                  ####  ../..  ../..  ../..  !!! WATCH FOR BUSDIN PULLED LOW BY OTHER DRIVER !!!
                                                  7434  K3/99  ../..  ../..  FI; K3=E37,E58,E70,E97-E98,E100
                                                  7437  K3/99  ../..  ../..  FI; K3=E90,E97,E105-E107,E118
                                                  7454  K3/99  ../..  ../..  FI; K3=E68,E109
                                                  7464  K3/99  ../..  ../..  FI; K3=E76,E113,E118
                                                  7470  K3/99  ../..  ../..  FI; K3=E75,E106
                                                  7475  K4/45  K3/40  K2/15  K7/8   FI; K4=E3,E33-E34,E51,E55,E71,E82,
                                                                                    E100-E101; K3=E3,E43,E51,E67,E82,E94,E102,E105;
                                                                                    K2=E77,E95,E106; K7=E43
                                                  7476  K4/50  K3/35  K2/15  K5/8   FI; K4=E3,E42-E43,E55,E59,E82,E109-E110;
                                                                                    K3=E41,E43,E57,E77,E82,E94,E107; K2=E96,E100,
                                                                                    E106; K5=E59
                                                  7477  K3/65  K4/40  ../..  FI; K3=E1,E41,E43,E51-E52,E61,E82,E102; K4=E3,E26,
                                                                                    E29,E55,E70-E71
                                                  7574  K3/99  ../..  ../..  FI; K3=E70,E97-E98,E100,E109,E116-E117,E120-E121
                                                  7674  K3/99  ../..  ../..  FI; K3=E70,E90,E97-E100
                                                  7774  K3/99  ../..  ../..  FI; K3=E95-E97,E99-E100,E104,E110,E119

284)  5764  TEST012F                  3147  7402  BUTR(DR6/7L), IR=(177777); DR7
                                                  ####  ../..  ../..  ../..  !!! WATCH FOR BUSDIN PULLED LOW BY OTHER DRIVER !!!
                                                  7403  K3/99  ../..  ../..  FI; K3=E64,E66

285)  5765  TEST012E                  3127  7417  BUT(MOV/DR7+IR5-3), IR=(177777); IR<5:3>="111", -MOV
                                                  ####  ../..  ../..  ../..  !!! WATCH FOR BUSDIN PULLED LOW BY OTHER DRIVER !!!
                                                  7400  K2/99  ../..  ../..  FI; K2=E26,E91
                                                  7406  K3/75  K5/25  ../..  FI; K3=E93; K5=E94
                                                  7407  K3/60  K5/20  K2/10  K4/10  K6/5   K7/5   FI; K3=E57,E66,E71,E76,
                                                                                    E78-E79,E81,E84,E92-E93,E95,E99,E101,E105-E107,
                                                                                    E110,E113,E115-E119; K5=E48-E49,E67-E68,E94,E106,
                                                                                    E113-E115; K2=E91,E103,E111-E112; K4=E12-E15,
                                                                                    E106; K6=E65-E66; K7=E73,E82
                                                  7413  K3/35  K5/30  K2/15  K6/5   K4/5   K7/5   FI; K3=E75,E78,E86-E87,E101,
                                                                                    E103; K5=E56,E61,E70,E92,E105-E106; K2=E91,E114;

                                                                                    K6=E73-E74; K4=E106; K7=E88
                                                  7415  K5/40  K3/30  K2/10  K7/5   K6/5.  K4/5   FI; K5=E56,E61,E67,E105-E106,
                                                                                    E112; K3=E77-E78,E86-E87,E101,E103; K2=E91,E114;
                                                                                    K7=E88; K6=E73-E74; K4=E106
                                                  7416  K5/35  K3/35  K2/15  K6/10  K4/5   K7/5   FI; K5=E49,E67-E68,E94,E106,
```

```
      ERROR                        Line              ->Module sequence->
(##)  code   Symbolic label        number  ENUA  TNUA  #1/%%  #2/%%  #3/%%  Test summary - Print reference - Chip information
----  -----  ---------------       ------  ----  ----  -----  -----  -----  ------------------------------------------------

      5765   TEST012E              [Continued]   BUT(MOV/DR7,IR5-3), IR=(177777); IR<5:3>="111", =MOV
                                                 7416  K5/35  K3/35  K2/15  E120; K3=E67,E76,E86,E93,E101; K2=E91,E103;
                                                                            K6=E65-E66; K4=E106; K7=E82
                                                 7437  K3/99  ../..  ../..  FI; K3=E68

286)  5766   TEST621G              15550  7401   BUTR(PREFETCH=JAM) NEGATED; GETS PREFETCH-H AT JAMUPP="0"
                                                 7403  K3/99  ../..  ../..  FI; K3=E44
                                                 7407  K3/99  ../..  ../..  K306=PREFETCH=JAM-LOGIC
                                                 7763  K2/75  K3/25  ../..  FI; K2=E10-E12,E37-E39,E64-E66; K3=E2,E4-E5

287)  5767   TEST012D              3108   7417   BUT(IR9-6), IR=(177777); IR<9:6>="1111"
                                                 ****  ../..  ../..  ../..  !!! WATCH FOR BUSDIN PULLED LOW BY OTHER DRIVER !!!
                                                 7407  K3/99  ../..  ../..  FI; K3=E76
                                                 7412  K3/99  ../..  ../..  FI; K3=E34,E54,E64,E67-E68,E74-E75,E98
                                                 7414  K3/75  K5/25  ../..  FI; K3=E103; K5=E105
                                                 7415  K3/45  K5/25  K2/10  K4/10  K6/5   K7/5   FI; K3=E34,E44,E50,E56,
                                                                            E76-E79,E84-E86,E88,E103,E112; K5=E51,E61,E70,
                                                                            E105-E106,E121; K2=E97,E113; K4=E12,E14,E105;
                                                                            K6=E73-E74; K7=E64
                                                 7416  K3/40  K5/30  K2/15  K7/5   K6/5   K4/5   FI; K3=E67,E78,E84-E86,E88,
                                                                            E96,E103,E112; K5=E51,E61,E70,E105-E106,E122;
                                                                            K2=E97,E113; K7=E44; K6=E73-E74; K4=E105
                                                 7437  K3/99  ../..  ../..  FI; K3=E68
                                                 7457  K3/99  ../..  ../..  FI; K3=E98

288)  5770   TEST046B              4950   7401   BUT(IR11=FLOAT), IR=(000200); FP-ROM(040)=(01)
                                                 7477  K3/99  ../..  ../..  K308=FP-INSTR-DECODE

289)  5771   TEST012C              3088   7403   BUTR(IR11), IR=(177777); IR11="1"
                                                 ****  ../..  ../..  ../..  !!! WATCH FOR BUSDIN PULLED LOW BY OTHER DRIVER !!!
                                                 5403  K2/99  ../..  ../..  FI; K2=E34
                                                 7401  K3/99  ../..  ../..  FI; K3=E23,E33-E34,E37-E39,E44,E56,E70,E76-E77
                                                 7407  K3/99  ../..  ../..  FI; K3=E74
                                                 7443  K3/99  ../..  ../..  FI; K3=E37,E98
                                                 7475  K2/90  K3/15  ../..  FI; K2=E34,E40,E59,E65,E68,E74,E80; K3=E39
                                                 7477  K3/99  ../..  ../..  FI; K3=E39
                                                 7767  K2/99  ../..  ../..  FI; K2=E40

290)  5772   TEST015P              3800   7403   BUTR(DR6/7L), IR=(000182); DR20=(DR6+7)
                                                 7402  K3/99  ../..  ../..  FI; K3=E66,E79

291)  5773   TEST012B              3069   7437   BUT(IR11=FLOAT), IR=(177777); IR<11>="1", FP-ROM(776)=(17)
                                                 ****  ../..  ../..  ../..  !!! WATCH FOR BUSDIN PULLED LOW BY OTHER DRIVER !!!
                                                 7400  K7/80  K6/15  K3/15  K2/15  FI; K7=E80; K6=E91; K3=E34; K2=E97
                                                 7401  K3/75  K5/25  ../..  FI; K3=E114; K5=E97
                                                 7403  K3/99  ../..  ../..  FI; K3=E34
                                                 7411  K7/99  ../..  ../..  FI; K7=E80
                                                 7413  K3/99  ../..  ../..  FI; K3=E34                      (Continued)
```



```
      ERROR                        Line              ->Module sequence->
(##)  code   Symbolic label        number  ENUA  TNUA  #1/%%  #2/%%  #3/%%  Test summary - Print reference - Chip information
----  -----  ---------------       ------  ----  ----  -----  -----  -----  ------------------------------------------------

      5773   TEST012B              (Continued)   BUT(IR11=FLOAT), IR=(177777); IR<11>="1", FP-ROM(776)=(17)
                                                 7416  K6/99  ../..  ../..  FI; K6=E91
                                                 7417  K3/65  K5/15  K2/10  K6/5   K7/5   K4/5   FI; K3=E34,E52,E54,E56,E64,
                                                                            E67-E68,E74-E75,E87-E88,E98-E99,E112,E114,
                                                                            E116-E117; K5=E20,E59-E60,E97-E98,E107; K2=E97,
                                                                            E107; K6=E82-E83; K7=E81; K4=E38,E105
                                                 7427  K3/85  K5/20  K2/10  K7/5   K6/5   K4/5   FI; K3=E34,E44,E56,E76-E77,
                                                                            E83,E85-E88,E89,E112,E114,E116; K5=E20,E59-E60,
                                                                            E97-E98,E109; K2=E97,E107; K7=E81; K6=E82-E83;
                                                                            K4=E105
                                                 7433  K3/99  ../..  ../..  FI; K3=E75,E86
                                                 7434  K3/40  K5/30  K2/15  K7/5   K6/5   K4/5   FI; K3=E76,E83,E85-E86,E88,
                                                                            E112,E114,E117; K5=E21,E59-E60,E97-E98,E108;
                                                                            K2=E97-E98; K7=E56; K6=E82-E83; K4=E105
                                                 7435  K3/99  ../..  ../..  FI; K3=E77,E86
                                                 7436  K3/80  K5/25  K2/10  K7/5   K6/5   K4/5   FI; K3=E50,E67,E75,E79,
                                                                            E83-E86,E88-E89,E112,E114; K5=E21,E59-E60,
                                                                            E97-E99; K2=E97-E98; K7=E56; K6=E82-E83; K4=E105

292)  5774   TEST021C              3797   7411   BUT(IR11=FLOAT), IR=(002315); FP-ROM(462)=(11)
                                                 7401  K3/99  ../..  ../..  FI; K3=E66
                                                 7416  K5/99  ../..  ../..  FI; K5=E59
                                                 7477  K3/99  ../..  ../..  K305=FP-INSTR-DECODE

293)  5775   TEST012P              3031   7417   BUT(IR15-12), IR=(177777); IR<15:12>="1111"
                                                 ****  ../..  ../..  ../..  !!! WATCH FOR BUSDIN PULLED LOW BY OTHER DRIVER !!!
                                                 3000  K7/99  ../..  ../..  FI; K7=E10,E94
                                                 3020  K7/99  ../..  ../..  FI; K7=E27
                                                 3220  K7/99  ../..  ../..  FI; K7=E10,E21,E27,E29,E42,E52
                                                 4102  K6/99  ../..  ../..  FI; K6=E70,E74,E92,E105
                                                 5760  K2/99  ../..  ../..  FI; K2=E61
                                                 7005  K2/65  K5/25  K3/25  FI; K2=E5,E29,E34,E38-E39,E50,E61,E68,E71,E76,
                                                                            E83-E85; K5=E20-E71,E48-E49,E51,E60; K3=E34,E37,
                                                                            E39,E81,E103,E112
                                                 7376  K2/99  ../..  ../..  FI; K2=E46
                                                 7400  K2/40  K3/25  K7/15  K5/15  K6/10  K4/5   FI; K2=E1,E5,E11,E17,E23,
                                                                            E25-E26,E29,E32-E34,E38-E39,E44-E46,E50-E51,
                                                                            E53-E54,E56,E59,E61-E62,E65,E70-E71,E74,E76-E77,
                                                                            E80,E82-E86,E91,E97-E98,E102-E103,E105,E107-E108,
                                                                            E111,E113-E114; K3=E23,E31-E34,E36-E39,E43,E45,
                                                                            E48,E51-E52,E57,E65-E66,E73,E92-E93,E101,E103,
                                                                            E113-E114; K7=E10,E13,E17,E20-E21,E27,E29,E33,
                                                                            E41,E52,E94; K5=E6,E20-E21,E31,E48-E49,E51,E54,

                                                                            E56,E61,E77,E95; K6=E65-E66,E70,E73-E74,E79-E80,
                                                                            E82-E83,E88,E95-E96,E98-E99,E101,E105,E108;
                                                                            K4=E29-E30,E38
                                                 7401  K3/99  ../..  ../..  FI; K3=E113
                                                 7402  K2/99  ../..  ../..  FI; K2=E29,E34,E61                 (Conti
```

###	ERROR code	Symbolic label	Line number	ENUA	TNUA	→Module sequence→ #1/##	#2/##	#3/##	Test summary = Print reference = Chip information
	5775	TE6T012A		[Continued]	BUT(IR15-12), IR=(177777); IR<15:12>="1111"				
					7403	K7/96	K2/5	../..	FI; K7=E20,E56,E64-E65,E73,E81-E82,E88; K2=E61
					7404	K6/50	K2/50	../..	FI; K6=E108; K2=E76
					7408	K3/55	K6/25	K3/15	K5/5 FI; K3=E34,E71,E83-E85,E90,E102-E103, E107-E108,E111,E113-E114; K6=E3,E5; K3=E34,E45; K5=E69
					7407	K3/50	K5/25	K2/10	K6/5 K7/5 K4/5 FI; K3=E21,E76,E83,E85, E87-E88,E108,E110,E113; K5=E31,E77,E91,E95,E98; K2=E85,E108; K6=E91-E92; K7=E80; K4=E95
					7412	K3/85	K5/10	K7/5	K2/5 FI; K3=E34,E54,E64,E67-E68,E74-E75; K5=E68, E92; K7=E33; K2=E61
					7413	K5/45	K3/30	K6/10	K2/10 K7/5 K4/5 FI; K5=E31,E59,E67,E70,E77, E84,E98,E98,E101; K3=E21,E75,E85,E113,E121; K6=E91-E92,E95,E105; K2=E85,E108; K7=E80; K4=E95
					7414	K5/99	../..	../..	FI; K5=E60-E61,E77
					7415	K3/40	K5/25	K2/15	K6/5 K4/5 K7/5 FI; K3=E21,E77,E85,E94,E104, E113; K5=E6,E67,E77,E90,E95,E98; K2=E85,E102; K6=E91-E92; K4=E95; K7=E65
					7416	K3/40	K5/30	K2/10	K6/5 K4/5 K7/5 FI; K3=E21,E67,E85,E94,E104, E113; K5=E6,E67,E77,E95,E98,E100; K2=E85,E102; K6=E91-E92; K4=E95; K7=E65
					7417	K3/99	../..	../..	FI; K3=E34,E38,E68
					7447	K3/99	../..	../..	FI; K3=E70,E98
294)	5776	TE6T024C	15839	7402	PROC=NUX(CUA=PORT)=(063851); CUA LOCKED, EXFLAG CLEAR, PREFETCH=JAM SET				
					7216	K3/99	../..	../..	FI; K3=E48
					7400	K3/75	K2/25	../..	FI; K3=E44,E48; K2=E111
295)	5777	TE6T011	2847	6532	BUTA(&USR-B) -> BUTA(RETURN) SEQUENCE, RETURN="1101 0101 1010"=(6532)				
					0777	K2/99	../..	../..	FI; K2=E55
					4646	K2/99	../..	../..	FI; K2=E46
					4747	K2/10	K4/20	K3/5	FI; K2=E8,E11,E23,E34,E46,E55; K4=E3; K3=E23
					5245	K6/99	../..	../..	FI; K6=E80
					6132	K2/99	../..	../..	FI; K2=E19,E41,E77-E78
					6432	K2/99	../..	../..	FI; K2=E7,E13,E25
					6812	K2/99	../..	../..	FI; K2=E7,E13,E25
					6532	K2/65	K4/35	../..	FI; K2=E26,E97; K4=E15
					6533	K2/99	../..	../..	FI; K2=E8,E14,E26
					6536	K2/99	../..	../..	FI; K2=E8,E14,E26
					7005	K3/55	K2/45	../..	FI; K3=E4,E6-E7,E11,E15,E43; K2=E70,E85
					7271	K3/60	K2/40	../..	FI; K3=E4,E6-E7,E11,E15,E43; K2=E61,E77,E85
					7274	K2/70	K3/30	../..	FI; K3=E1,E3,E6-E7,E12,E15,E18,E24,E30,E36; K3=E23, E58,E70
					7275	K2/99	../..	../..	FI; K2=E8,E25
					7301	K6/99	../..	../..	FI; K6=E104
					7417	K3/99	../..	../..	FI; K3=E83
					7532	K2/99	../..	../..	FI; K3=E35,E41,E77
					7577	K2/99	../..	../..	FI; K3=E42

Module codes: K1/DCS K2/UWORD K3/IRDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

###	ERROR code	Symbolic label	Line number	ENUA	TNUA	→Module sequence→ #1/##	#2/##	#3/##	Test summary = Print reference = Chip information
296)	6752	TE6T002	2854	6631	NUA SEQUENCING, NO "BUT"				
					2631	K2/99	../..	../..	FI; K2=E55
					6231	K3/55	K2/45	../..	FI; K3=E43,E111; K2=E7,E19
					6777	K3/50	K2/50	../..	FI; K3=E11; K2=E32
297)	6520	TPST720P	16788	7402	DATIB=-BYTE, ODD ADDR JAM; JAM=(101004)				
					7400	K6/99	../..	../..	FI; K6=E52
298)	6521	TPST713D	16689	7402	DATI=NOINT, I/O PAGE, 16. BIT I/O PAGE MODE; SERVICE=(101740)				
					7400	K6/99	../..	../..	FI; K6=E51
					7401	K2/99	../..	../..	FI; K2=E8,E14,E26
299)	6522	TE6T713C	16664	7403	BUTR(OTHER-JAM=FP), ILLEGAL INTERNAL ADR; OTHER-JAM="1"				
					7401	K7/99	../..	../..	FI; K7=E74
300)	6523	TE6T712D	16555	7402	DATI, I/O PAGE, 16. BIT I/O PAGE MODE; SERVICE=(101740)				
					7400	K7/99	../..	../..	FI; K7=E82
					7403	K7/65	K6/35	../..	FI; K7=E80; K6=E99
301)	6524	TE6T712C	16530	7401	BUTR(OTHER-JAM=FP), VALID INTERNAL ADDR; OTHER-JAM="0"				
					7402	K2/85	K5/15	../..	FI; K2=E26,E88,E104,E110,E116; K5=E105
302)	6526	TP6T711D	16427	7402	CLEAR JAM ERRORS/SSYN TIMEOUT; JAM=(001000)				
					7400	K6/90	K7/10	../..	FI; K6=E3,E51,E10-E81,E99; K7=E58
303)	6530	TE6T711C	16404	7402	DATOB=BYTE, I/O PAGE, 16. BIT I/O PAGE MODE; SERVICE=(005740)				
					7400	K2/70	K4/20	K7/5	K6/5 FI; K2=E1,E8,E14,E26,E31,E37,E43,E49,E52, E71,E82,E91,E105; K4=E13,E15; K7=E81,E97; K6=E14
					7401	K2/99	../..	../..	FI; K2=E26
304)	6534	TP6T711B	16374	7402	DATOB=BYTE, SSYN TIMEOUT; JAM=(021200)				
					7400	K6/55	K7/30	K5/5	FI; K6=E80-E82,E81,E99; K7=E58,E64-E65,E84; K5=E84
					7401	K6/80	K7/25	K5/20	K3/10 FI; K6=E47,E55,E76,E85; K7=E81,E84; K5=E84, E87,E93; K3=E14
305)	6535	TE6T512F2	13261	7412	KT-BSPKI(SF)=(06), NEGATED				
					7400	K3/99	../..	../..	FI; K3=E47
					7477	K3/99	../..	../..	K312=KT-SP-ADDR-DECODE
306)	6537	TE6T512D1	13185	7400	KT-ASPKI(SF)=(16), ASSERTED, =F2=SM6=PS15				
					7412	K3/99	../..	../..	FI; K3=E47
					7477	K3/99	../..	../..	K312=KT-SP-ADDR-DECODE
307)	6541	TP6T373A	9738	4777	BUTA(DIAGNOSE) CAUSES XFER TO 8,M, ROM/JAMUPP TO DCS RETURN				
					0000	K2/99	../..	../..	FI; K2=E24,E30,E34,E45,E48,E51,E54,E60,E66,E68-E69,

Module codes: K1/DCS K2/UWORD K3/IRDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

###	ERROR code	Symbolic label	Line number	ENUA	TNUA	→Module sequence→ #1/%%	#2/%%	#3/%%	Test summary + Print reference + Chip information
	6541	TEST373A	(Continued)		BUTA(DIAGNOSE) CAUSES XFER TO S.M. ROM/JAMUPP TO DCS RETURN				
					0000	K2/99	../..	../..	E78;E81
					0003	K3/99	../..	../..	FI; K3=E23
					0020	K2/99	../..	../..	FI; K2=E21
					0055	K2/99	../..	../..	FI; K2=E15
					0212	K2/99	../..	../..	FI; K2=E4,E10,E16,E22,E27-E28,E31,E36-E37,E42-E43, E49,E52,E58,E64,E67,E73,E79
					0777	K2/99	../..	../..	FI; K2=E24
					1020	K2/99	../..	../..	FI; K2=E80
					1025	K6/99	../..	../..	FI; K6=E42
					2000	K3/99	../..	../..	FI; K3=E40,E81,E57,E63
					2042	K2/99	../..	../..	FI; K2=E21,E74
					2153	K2/99	../..	../..	FI; K2=E75
					2211	K2/99	../..	../..	FI; K2=E75
					2247	K3/99	../..	../..	FI; K3=E22
					2310	K2/99	../..	../..	FI; K2=E79
					2314	K2/99	../..	../..	FI; K2=E75
					2408	K2/99	../..	../..	FI; K2=E75
					2442	K2/99	../..	../..	FI; K2=E21
					2626	K2/99	../..	../..	FI; K2=E24
					2530	K2/99	../..	../..	FI; K2=E24
					2577	K2/99	../..	../..	FI; K2=E48
					2671	K2/99	../..	../..	FI; K2=E87
					3007	K2/99	../..	../..	FI; K2=E87;E78
					3054	K2/99	../..	../..	FI; K2=E8,E11,E15,E17,E23,E29,E32,E38,E44,E50,E53, E69,E65,E60,E74,E80
					3353	K2/99	../..	../..	FI; K2=E75
					3462	K2/99	../..	../..	FI; K2=E42,E74
					3508	K2/99	../..	../..	FI; K2=E15,E21,E74
					3807	K2/99	../..	../..	FI; K2=E21
					3823	K2/99	../..	../..	FI; K2=E6,E11,E15,E17,E21,E27,E29,E32,E44,E50,E53, E60,E73-E74,E80
					3837	K2/99	../..	../..	FI; K2=E48,E60
					3777	K2/99	../..	../..	FI; K2=E42
					4621	K2/99	../..	../..	FI; K2=E60
					4747	K3/85	K2/15	../..	FI; K3=E0-E9; K2=E18
					5301	K2/99	../..	../..	FI; K2=E48,E57,E61,E78
					5558	K2/99	../..	../..	FI; K2=E48,E57,E63,E78
					5597	K2/99	../..	../..	FI; K2=E78
					6073	K2/99	../..	../..	FI; K2=E73
					7005	K2/99	../..	../..	FI; K2=E6,E12,E18,E24,E33,E45,E75
					7037	K4/60	K2/30	../..	FI; K4=E06; K2=E81
					7274	K2/99	../..	../..	FI; K2=E35,E47,E55
					7373	K2/99	../..	../..	FI; K2=E36,E73
					7617	K6/60	K7/40	../..	FI; K6=E39-E40,E42-E43,E58; K7=E34,E52
					7777	K2/99	../..	../..	FI; K2=E47

Module codes: K1/DCS K2/UNIBD K3/IRDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

###	ERROR code	Symbolic label	Line number	ENUA	TNUA	→Module sequence→ #1/%%	#2/%%	#3/%%	Test summary + Print reference + Chip information
308)	6547	TEST512A1	13026	7405	KT-A&PHI(&F)=(02), NEGATED, -SR6				
					7412	K3/99	../..	../..	FI; K3=E45,E47
					7477	K3/99	../..	../..	K312=KT-SP-ADDR-DECODE
309)	6544	TEST512A2	13053	7400	KT-B&PHI(DF)=(16), ASSERTED, DR6=PS15=FLOAT				
					7412	K3/88	K4/15	../..	FI; K3=E45,F47,E58,E69,E79; K4=E13,E15
					7477	K3/99	../..	../..	K312=KT-SP-ADDR-DFCODE
310)	6546	TEST512P2	13108	7400	KT-B&PHI(&F)=(16), ASSERTED, P2=DR6=SM0=PS13=-FLOAT				
					7412	K3/65	K4/35	../..	FI; K3=E48,E47,E89; K4=E13,E15
					7477	K3/89	../..	../..	K312=KT-SP-ADDR-DECODE
311)	6552	MFS805	12715	6545	BUTR(MF=SAME=STACK)=N, EXPECTED L				
					6547	K3/80	K2/20	../..	FI; K3=E56,E79,E89,E96; K2=E106,E118
312)	6553	MFS806	12725	6547	BUTR(MF=SAME=STACK)=L, EXPECTED H				
					6545	K3/75	K2/20	K5/5	FI; K3=E13,E47,E56,E63,E69,E79,E89,E96; K2=E62,E84, E106,E118; K5=E63
313)	6556	TEST512C1	13130	7400	KT-A&PHI(DF)=(16), ASSERTED, P1=DR6=SM0=PS13=-FLOAT				
					7412	K3/99	../..	../..	FI; K3=E119
					7477	K3/99	../..	../..	K312=KT-SP-ADDR=DECODE
314)	6557	TEST507D	12243	7407	BUTM(MASK=PS(T)), MASK=PS(T)=1=1="1"				
					7406	K2/99	../..	../..	FI; K2=E79,E101,E105,E112
315)	6561	TEST500	10968	6131	(NUA SEQUENCING LOGIC ERROR)				
					????	K1/99	../..	../..	INTERNAL DCS ERROR
316)	6563	TEST974	9898	NONE	(NUA SEQUENCING LOGIC/DCS ERROR)				
					????	K1/99	../..	../..	INTERNAL DCS ERROR
317)	6565	TEST512P1	13061	7412	KT-A&PHI(DF)=(06), NEGATED				
					7400	K3/99	../..	../..	FI; K3=E48,E47
					7477	K3/99	../..	../..	K312=KT-SP-ADDR=DECODE
318)	6566	TEST512C2	13157	7412	KT-B&PHI(&F)=(06), NEGATED				
					7400	K3/99	../..	../..	FI; K3=E47
					7406	K4/99	../..	../..	FI; K4=E59
					7477	K3/99	../..	../..	K312=KT-SP-ADDR-DFCODE
319)	6567	ERROR624A	15803	0005	MICROBREAK JAMUPP AT (6255) ATTEMPTED; DID NOT OCCUR				
					6567	K3/99	../..	../..	FI; K3=E81,E91,E101,E111-E112

Module codes: K1/DCS K2/UNIBD K3/IRDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

###	ERROR code	Symbolic label	Line number	ENDA	TNUA	→Module sequence→ #1/##	#2/##	#3/##	Test summary - Print reference - Chip information
320)	6571	TEST372A	9606	6222	RUTA(SUBR=A)/BUTA(RETURN) SEQUENCE; D<14:10>= => RETURN; "1100 1001 0010"=(6222				
					4022	K2/99	../..	../..	FI; K2=E92
					4222	K2/99	../..	../..	FI; K2=E48,E77,E89,E92-E93,E95,E102,E106,E118
					6000	K2/99	../..	../..	FI; K2=E90
					6022	K2/99	../..	../..	FI; K2=E3,E6,E12,E18,E24-E25,E30,E33,E39,E45,E51, E54,E60,E66,E69,E75,E81,E89,E92-E93,E98,E101,E107
					6076	K2/99	../..	../..	FI; K2=E84,E90
					6200	K6/50	K2/50	../..	FI; K6=E34; K2=E86
					6202	K2/99	../..	../..	FI; K2=E3,E6,E12,E18,E24,E30,E33,E39,E45,E51,E54, E60,E66,E69,E75,E81,E86-E88,E104,E110,E113,E116
					6220	K2/99	../..	../..	FI; K2=E4,E9,E12,E18,E24,E26,E30,E33,E39,E45,E51, E54,E60,E66,E69,E75,E81,E86-E88,E104,E110,E114, E116
					6222	K2/99	../..	../..	FI; K2=E9,E26,E86,E103
					6226	K2/99	../..	../..	FI; K2=E9,E26,E86,E114
					6232	K2/99	../..	../..	FI; K2=E9,E26,E86,E113
					6262	K2/99	../..	../..	FI; K2=E3,E25,E86,E98
					6322	K2/99	../..	../..	FI; K2=E3,E25,E92,E98
					6362	K2/99	../..	../..	FI; K2=E98
					6376	K2/70	K7/15	K5/15	FI; K2=E61,E71,E93; K7=E36,E51; K5=E15
					6622	K2/99	../..	../..	FI; K2=E62,E77,E92,E107
					7222	K2/99	../..	../..	FI; K2=E72,E77,E92,E102
					7700	K2/99	../..	../..	FI; K2=E86
321)	6572	TEST373B	9814	7425	EXEC B.M. ROM CODE FOR 'FLPADR' ASSERT TO ASPHI/READ				
					7400	K4/58	K2/35	K3/10	FI; K4=E1,E16,E20,E51,E54,E59-E60,E67,E69,E84,E93, E104,E114; K2=E4,E7,E10,E16,E18,E22,E25,E27-E28, E37,E43,E49,E52,E58,E64,E73; K3=E3,E7
					7402	K3/55	K2/25	K4/15	FI; K3=E1-E20,E22-E26,E48,E51,E66; K2=E4,E10,E16, E22,E27-E28,E31,E36-E37,E43,E49,E52,E58,E64,E67, E69,E73,E79; K4=E13,E15,E51,E54,E62,E69,E113-E114
					7403	K3/95	K4/5	../..	FI; K3=E1-E19,E22,E24-E26; K4=E29
					7417	K4/80	K2/20	../..	FI; K4=E69-E60,E84,E93; K2=E31,E69
					7420	K4/99	../..	../..	FI; K4=E1,E61,E66,E68,E70,E74-E77,E83,E85,E93-E94, E103-E104,E113-E114
					7426	K2/99	../..	../..	FI; K2=E43
					7432	K2/99	../..	../..	FI; K2=E4,E16,E22,E27-E28,E37,E43,E52,E58,E67,E79
					7433	K4/99	../..	../..	FI; K4=E84,E60,E66-E68,E70,E75-E76,E83,E103
					7434	K7/30	K6/30	K2/30	K3/15 FI; K7=E41,E66; K6=E78,E90; K2=E18,E75; K3=E7
322)	6573	TEST361A	8827	7434	BR=(052525), D[C]="1"; BR/XMUX=(100125) (FLTPT)				
					7400	K4/96	K3/5	../..	FI; K4=E27,E29,E37,E44,E46,E64,E69,E70; K3=E92
					7402	K4/99	../..	../..	FI; K4=E46
					7417	K4/99	../..	../..	FI; K4=E27 [Continued]

Module codes: K1/DCS K2/UWORD K3/IRDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

###	ERROR code	Symbolic label	Line number	ENDA	TNUA	→Module sequence→ #1/##	#2/##	#3/##	Test summary - Print reference - Chip information
	6573	TEST361A	[Continued]		BR=(052525), D[C]="1"; BR/XMUX=(100125) (FLTPT)				
					7420	K4/99	../..	../..	FI; K4=E37,E44
					7425	K4/99	../..	../..	FI; K4=E16
					7426	K4/99	../..	../..	FI; K4=E37
323)	6574	TEST710D	16265	7402	DATO, I/O PAGE, 16, BIT I/O PAGE MODE; SERVICE=(003740)				
					7400	K7/80	K6/30	K5/20	FI; K7=E44,E88,E7; K6=E51,E53-E54,E59,E105; K5=E7, E27,E30,E95
324)	6575	TEST710E	16294	7402	CLEAR JAM ERRORS/ODD ADDR; JAM=(001000)				
					7401	K7/99	../..	../..	FI; K7=E25,E59
325)	6576	TEST512D2	13212	7405	XT=DSPHI[DF1=(02), NEGATED, =DR6				
					7412	K3/99	../..	../..	FI; K3=E47,E79,E93
					7427	K3/99	../..	../..	K312=KT-SP-ADDR-DECODE
326)	6577	TEST512F1	13234	7400	XT=ASPHI[DF1=(16), ASSERTED, F1=DR6=-DK0=P815				
					7405	K3/99	../..	../..	FI; K3=E45
					7412	K3/99	../..	../..	FI; K3=E47
					7427	K3/99	../..	../..	K312=KT-SP-ADDR-DECODE
327)	6600	TEST510D	12438	7406	BUTM(MASK-PS[T]), MASK-PS[T]=1=0="0"				
					7407	K2/99	../..	../..	FI; K2=E72,E101,E105
328)	6601	TEST503DA	11359	7406	BUTM(FLTPT-INSTR), IR=(125122), -FLTPT				
					7407	K2/99	../..	../..	FI; K2=E83,E112
329)	6603	TEST510A	12363	7407	BUTR(BG-SERVICE-L), NEGATED WHEN PS PRIO<7:5>=(7)				
					7403	K7/80	K3/15	K2/5	FI; K7=E3,E5,E28,E76; K3=E50,E74; K2=E100
330)	6604	TEST510E	12497	7403	BUTR(SERVICE-H), ASSERTED="1" WHEN FLAG<7>H="1"				
					7402	K2/99	../..	../..	FI; K2=E78,E100
331)	6605	TEST506F	12099	7403	BUTR(INTR-HIGH), INTR-HIGH-H="1"; INTERNAL SERVICE CLEAR				
					4747	K6/99	../..	../..	FI; K6=K78
					7401	K7/85	K3/10	K2/5	FI; K7=E37,E43,E48,E50,E57-E58,E64-E67,E76-E77, E84; K3=E44,E50; K2=E72,E100
332)	6607	TEST511A	12755	NONE	(NDA SEQUENCING LOGIC ERROR)				
					7777	K1/99	../..	../..	INTERNAL DCS ERROR
333)	6610	TEST510F	12524	7402	BUTR(SERVICE-H), NEGATED="0", NO INPUTS ACTIVE				
					7403	K7/80	K2/15	K3/5	FI; K7=E3,E7-E8,E11,E16,E27; K2=E38,E78,E100; K3=E64
334)	6611	TEST507C	12221	7406	BUTM(PS03), PS<03>H="0"				
					7407	K2/99	../..	../..	FI; K2=E112
					7407	K2/80	K3/20	../..	K210=MULTIPLE-BUT; K306=PS(CC)

Module codes: K1/DCS K2/UWORD K3/IRDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

###	ERROR code	Symbolic label	Line number	ENUA	TNUA	->Module sequence-> #1/%%	#2/%%	#3/%%	Test summary - Print reference - Chip information
3351	6613	TEST507A	12144	7402					LOAD/READ PS=(140)25) PS-PORT PROC=MUX; UCON FUNCTION
					7400	K2/50	K3/45	K4/5	K7/5 FI; K2=E102-E103,E105,E111,E113-E114,E116, E118; K3=E35,E71-E73,E83; K4=E63,E72; K7=E3
					7401	K3/45	K2/45	K4/10	FI; K3=E28,E47,E54,E63; K2=E108,E116; K4=E72
3361	6615	TEST504A	11578	4777					NUA SEQUENCING ERROR, JAMUPP ERROR; BUTA(DIAGNOSE) TO B.M. 'LOADNZM4/LOADNZM5'
					4744	K2/99	../..	../..	FI; K2=E54
					4757	K2/99	../..	../..	FI; K2=E54
3371	6616	TEST510C	12418	7403					BUTR(SERVICE-H), ASSERTS0="1" WHEN INTR-HIGH-H="0"
					7345	K3/99	../..	../..	FI; K3=E76
					7402	K2/75	K3/25	../..	FI; K2=E100; K3=E84
3381	6617	TEST510DA	12475	7407					BUTM(D00), D=(040401)
					7406	K2/99	../..	../..	K210=MULTIPLE=BUT
3391	6621	TEST506A	11952	7402					LOAD/READ PS=(030252) PS/PROC=MUX; BUTA(CLR-...) SETS BUSDIN=EMIT; UCON FCN
					7400	K2/50	K3/40	K7/10	K5/5 FI; K2=E61-E62,E70-E71,E92-E93,E98, E102-E103,E107-E108,E111-E114,E116-E118; K3=E47, E52,E57,E62-E63,E72-E73,E83,E89; K7=E3,E27,E42, E44,E68; K5=E13,E54,E84
					7401	K2/50	K3/10	../..	FI; K2=E62,E92,E117; K3=E57
					7403	K2/99	../..	../..	FI; K2=E108
3401	6623	TEST505A	11847	7402					SET ALL FLAGS, CHECK BUTA(CLR-FLAG-RES-UCON) CLEARS SHORT TERM FLAGS
					7400	K2/99	../..	../..	FI; K2=E101,E108,E119
3411	6629	TEST533A	13528	7402					SHIFTER, AMUX/D[HI]=D[LO]=(052652), D=(0)(052652)
					7400	K4/99	../..	../..	FI; K4=E29,E51,E54,E60-E62,E66-E69,E77,E81-E85, E91-E94,E103-E104,E113-E114
3421	6627	TEST520A	13322	7403					INSTR-BRANCH-L="H", IR=(101004)=BNI, PS(NZVC)=(04)
					7402	K3/99	../..	../..	FI; K3=E64,E83
					7407	K3/99	../..	../..	K3=PS(CC)-BRANCH(F64,72-73,83)
3431	6630	TEST503A	11318	7406					BUTM(FLAG7), FLAG<7>H="0"
					7407	K2/85	K3/15	../..	FI; K2=E112; K3=E54
3441	6631	TEST003	2673	5525					NUA SEQUENCING, PAGE (6) -> (5), UBF=(35)
					4525	K2/50	K4/10	K3/10	FI; K2=E35,E47,E58,E62,E64,E67,E72-E73,E78,E91; K4=E3; K3=E111
					5425	K3/90	K2/90	../..	FI; K3=E58,E91; K2=E1,E7
					5520	K2/99	../..	../..	FI; K2=E76
					5544	K2/99	../..	../..	FI; K2=E90
					5546	K2/99	../..	../..	FI; K2=E71

[Continued]

Module codes: K1/DCS K2/UWORD K3/IRDECODE K4/DATAPATH K5/RTCACHE K6/TIMING K7/STATUS

###	ERROR code	Symbolic label	Line number	ENUA	TNUA	->Module sequence-> #1/%%	#2/%%	#3/%%	Test summary - Print reference - Chip information
	6631	TEST003	[Continued]						NUA SEQUENCING, PAGE (6) -> (5), UBF=(35)
					6525	K2/99	../..	../..	FI; K2=E48,E59,E65,E68,E74,E80
					7525	K2/99	../..	../..	FI; K2=E35,E47
3451	6632	TEST503C	11338	7402					BUTR(FLIPT-PROC-H), FLIPT-PROC-H="0"
					6402	K2/99	../..	../..	FI; K2=E34
					7403	K3/50	K3/50	../..	FI; K3=E64; K2=E119
					7406	K2/60	K3/40	../..	FI; K2=E2,E6,E66,E69,E75,E81; K3=E21,E81
3461	6633	TEST533B	13569	7402					SHIFTER, AMUX/D[HI]=D[LO]=(125125), D=(0)(125125)
					6367	K3/99	../..	../..	FI; K3=E38
					7400	K4/99	../..	../..	FI; K4=E51,E54,E58-E62,E66-E69,E74-E76,E81-E85, E93-E94,E103-E104,E113
					7401	K4/99	../..	../..	FI; K4=E78-E76
					7403	K4/99	../..	../..	FI; K4=E51,E54,E62,E69,E84,E103,E114
3471	6634	TEST534B	13650	7402					SHIFTER, AMUX/D[LO]=D[HI]=(052652), D=(0)(125125)
					7400	K4/99	../..	../..	FI; K4=E93-E94,E103-E104,E113
					7403	K4/99	../..	../..	FI; K4=E114
3481	6635	TEST534D	13725	7402					SHIFTER, AMUX/D[C]=D[HI]=(000177), D=(0)(177777)
					7400	K4/75	K2/20	K3/10	FI; K4=E59,E93,E104,E113-E114; K2=E65,E68; K3=E75
					7403	K4/99	../..	../..	FI; K4=E114
3491	6636	TEST534F	13805	7402					SHIFTER=COUNTER, AMUX/COUNTER[HI]=(052652), CTR=(125), D=(0)(125000)
					7400	K4/99	../..	../..	FI; K4=E87,E90-E91,E93,E104,E113
					7401	K4/99	../..	../..	FI; K4=E114
					7403	K4/99	../..	../..	FI; K4=E8,E91,E114
3501	6637	TEST535A	13926	7402					SHIFTER, BMUX/4=D[C]=AMUX=(002645), D=(0)(055132)
					7400	K4/99	../..	../..	FI; K4=E69,E81-E85
					7403	K4/99	../..	../..	FI; K4=E69
3511	6640	TEST536B	14019	7402					SHIFTER, CMUX/2=D[C]=BMUX=(034343), D=(0)(161616)
					7400	K4/99	../..	../..	FI; K4=E54,E60-E61,E67,E75
					7401	K5/99	../..	../..	FI; K5=E105
					7403	K4/99	../..	../..	FI; K4=E54
3521	6641	TEST536D	14103	7402					SHIFTER, CMUX/D[C]=BMUX=(063146), D=(0)(146314)
					7400	K4/90	K3/15	../..	FI; K4=E60-E62,E67-E69,E75-E76; K3=E42
					7403	K4/99	../..	../..	FI; K4=E54
3531	6642	TEST536F	14196	7402					SHIFTER, CMUX/BMUX=BENDMUX=(063146), D/SR=(0)(031463)(077777)
					7400	K4/99	../..	../..	FI; K4=E60-E62,E67-E68,E75-E77
					7403	K4/99	../..	../..	FI; K4=E54
3541	6643	NEWCTR537A	14291	7370					(NUA SEQUENCING LOGIC ERROR)
					????	K1/99	../..	../..	INTERNAL DCS ERROR

Module codes: K1/DCS K2/UWORD K3/IRDECODE K4/DATAPATH K5/RTCACHE K6/TIMING K7/STATUS

###	ERROR code	Symbolic label	Line number	ENUA	TNUA	->Module sequence-> #1/##	#2/##	#3/##	Test summary - Print reference - Chip information
355)	6644	TEST710B	16212	7402	DATO, ODD ADDR ERR; JAM=(101004)				
					7400	K7/45	K5/45	K6/10	F1; K7=E61,E65-E66,E80,E82,E84; K5=E11,E43-E45, E64; K6=E76
					7401	K6/80	K7/20	../..	F1; K6=E71,E80-E81,E99; K7=E80,E82
356)	6645	TEST710C	16236	7402	DATO, I/O PAGE, 18. BIT I/O PAGE MODE; SERVICE=(002340)				
					7400	K7/45	K5/15	K6/10	K4/5 K3/6 F1; K7=E14,E27,E32,E44,E64-E65,E69, E70,E73,E76-E77,E79-E82,E86,E98; K5=E13-E14,E39, E67,E93; K6=E43,E46-E47,E53-E54,E59,E81; K4=E20, E65; K3=E34
					7401	K2/65	K7/15	K6/15	F1; K3=E1,E82; K7=E80; K6=E43
					7403	K7/99	../..	../..	F1; K7=E80
357)	6646	TEST503K	11531	7407	BUTM(FLTPT+INSTR), IR=(175552), FLTPT				
					7406	K2/50	K3/50	../..	K210=MULTIPLE-BUT; K)04=FP=DECODE
358)	6647	TEST507F	12286	7407	CHECK P8[C]=P8<00>="1", D[C] INPUTS P8[C], CIN/P8[C], CIN/D[C]=P8[C] ALL SET				
					7400	K3/50	K2/50	../..	F1; K3=E93; K2=E112
					7401	K4/99	../..	../..	F1; K4=E96
					7406	K3/99	../..	../..	F1; K3=E38,E46
359)	6653	TEST536E	14154	7402	SHIFTER, CMUX/BMUX=SENDMUX=(114631), D/SR=(0)(146314)(100000)				
					7400	K4/99	../..	../..	F1; K4=E54,E60-E62,E67-E68,E75-E77,E80
360)	6655	TEST538C	14067	7402	SHIFTER, CMUX/D[C]=BMUX=(114631), D=(1)(031463)				
					7400	K4/95	K6/5	../..	F1; K4=E54,E60-E62,E67-E68,E70,E75-E76; K6=E108
361)	6657	TEST536A	13973	7402	SHIFTER, CMUX/2=D[C]=BMUX=(143434), D=(1)(016161)				
					7400	K4/99	../..	../..	F1; K4=E60-E62,E68,E76,E84
					7401	K4/99	../..	../..	F1; K4=E54,E60-E62,E67-E68,E70,E75-E76
362)	6661	TEST535A	13879	7402	SHIFTER, BMUX/4=D[C]=AMUX=(175132), D=(1)(122645)				
					7400	K4/99	../..	../..	F1; K4=E3,E60,E69,E75,E77,E83-E85
363)	6663	TEST534E	13765	7407	SHIFTER=COUNTER, AMUX/COUNT=D[LO]=(125125), CTR=(252), D=(0)(000125)				
					7400	K4/96	K3/5	../..	F1; K4=E87,E90-E91,E93,E104,E113-E114; K2=E55
					7401	K4/80	K6/15	K2/5	F1; K4=E87,E90-E91; K6=E103; K2=E55
					7403	K4/99	../..	../..	F1; K4=E91,E114
364)	6665	TEST534C	13691	7402	SHIFTER, AMUX/8=D[C]=D[HI]=(177400), D=(1)(000000)				
					7400	K4/99	../..	../..	F1; K4=E87-E88
					7401	K4/99	../..	../..	F1; K4=E28-E29,E93,E104,E113-E114
365)	6667	TEST534A	13615	7402	SHIFTER, AMUX/D[LO]=D[HI]=(125125), D=(0)(052652)				
					6356	K2/65	K3/35	../..	F1; K3=E2,E8-E9,E21,E27,E60,E66,E69,E75,E81;

Module codes: K1/DC5 K2/UWORD K3/IRDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

###	ERROR code	Symbolic label	Line number	ENUA	TNUA	->Module sequence-> #1/##	#2/##	#3/##	Test summary - Print reference - Chip information
	6667	TEST534A	(Continued)		SHIFTER, AMUX/D[LO]=D[HI]=(125125), D=(0)(052652)				
					6356	K2/65	K3/35	../..	K3=E21-E22,E44,E56,E77
					7341	K2/99	../..	../..	F1; K2=E34
					7343	K3/99	../..	../..	F1; K3=E37-E39,E44,E64
					7400	K4/80	K3/5	K2/5	F1; K4=E28-E29,E77,E93-E94,E103-E104,E113-E114; K3=E1; K2=E74
					7401	K4/99	../..	../..	F1; K4=E28-E29,E93,E104,E113-E114
366)	6670	TEST503D	11379	7476	BUT(INSTR1), IR=(178262), CLASS=A(FLTPT), FLAG<4:5>H="10"				
					7474	K3/85	K7/15	../..	F1; K2=E62,E100; K7=E44
					7477	K3/99	../..	../..	K308=FLAGS/FP=INSTR1
367)	6671	TEST0500	11008	6377	(NUA SEQUENCING LOGIC ERROR)				
					6173	K2/70	K3/30	../..	F1; K2=E2-E3,E6-E7,E12,E15,E18,E24,E30,E36; K3=E23, E58,E70
					7376	K3/99	../..	../..	F1; K3=E58
					7777	K1/99	../..	../..	INTERNAL DC8 ERROR
368)	6672	TEST504F	11742	7407	BUTM(EXFLAG2), EXFLAG<2>H="1"				
					7406	K2/99	../..	../..	K210=FLAGS
369)	6673	TEST500A	11070	7434	SP REWRITE FUNCTION WR(A,BI,A) DOESN'T WRITE B-SIDE				
					7477	K4/99	../..	../..	K405=SP=REWRITE-CNTL,K406/7=A/B-XPADS
370)	6674	TEST507F	12265	7401	BUTR(INTR-HIGH), INTR-HIGH=H="0"; INTERNAL SERVICE SET				
					7403	K2/65	K3/35	../..	F1; K2=E72; K3=E44
371)	6675	TEST51DR	12396	7407	BUTR(BG+FP/SERVICE), NEGATED="0" WHEN P8 PRIO<7:5>=(7)				
					7377	K2/99	../..	../..	F1; K2=E43
					7377	K2/65	K3/35	../..	F1; K2=E16-E18,E44-E45,E67-E69; K3=E3,E7-E8,E25
					7417	K3/99	../..	../..	F1; K3=E80,E76
372)	6677	TEST503A	11238	4332	NUA SEQUENCING ERROR; RUTA(DIAGNOSE) TO 0,M, 'LOADNEW4', 'LOADNEW5'				
					0000	K2/99	../..	../..	F1; K2=E11,E17,E23,E29,E32,E38,E44,E50,E53-E54,E57, E59,E63,E68,E69,E72,E74,E78,E80,E92-E93
					0127	K2/75	K3/25	../..	F1; K2=E35,E47,E72,E76; K3=E111
					0332	K2/99	../..	../..	F1; K2=E63,E80
					0732	K2/99	../..	../..	F1; K2=E8,E11,E15,E17,E23,E29,E32,E38,E44,E50,E53, E59,E65,E69,E74,E80
					4032	K2/99	../..	../..	F1; K2=E30
					4126	K2/99	../..	../..	F1; K2=E6,E12,E18,E24,E30,E33,E39,E45,E51,E54,E60, E66,E69,E75,E81
					4322	K2/99	../..	../..	F1; K2=E81
					4330	K2/99	../..	../..	F1; K2=E81
					4331	K2/99	../..	../..	F1; K2=E81
					4335	K2/99	../..	../..	F1; K2=E81
					4336	K2/99	../..	../..	F1; K2=E80
					4337	K2/99	../..	../..	F1; K2=E57,E79

(Continued

Module codes: K1/DC5 K2/UWORD K3/IRDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

#(4)	ERROR code	Symbolic label	Line number	ENUA	TNUA	>Module sequence< #1/##	#2/##	#3/##	Test summary + Print reference - Chip information
	6677	TEST503A	(continued)						NUA SEQUENCING ERROR; BUTA(DIAGNOSE) TO S,M, 'LOADN2W4', 'LOADN2W8'
					4352	K2/99	../..	../..	FI; K2=E30
					4372	K2/99	../..	../..	FI; K2=E30
					4375	K2/99	../..	../..	FI; K2=E92
					4732	K2/99	../..	../..	FI; K2=E5,E11,E17,E21,E23,E29,E32,E38,E44,E50,E53, E59,E65,E68,E74,E80
					4747	K2/99	../..	../..	FI; K2=E1-E20,E24-E26
					7005	K2/99	../..	../..	FI; K2=E5,E11,E15,E17,E29,E32,E50,E53-E54,E61,E68, E68,E74,E80
373)	6700	TEST503E	11407	7402					EXFLAGS<2:1>H="01", READ THRU CUA-PORT, PROC=MUX
					7400	K2/95	K3/15	../..	FI; K2=E92,E101,E103,E106,E108,E111-E112,E119; K3=E48
					7403	K2/99	../..	../..	FI; K2=E108
374)	6701	TEST377A	9658	5555					BUTA(SUBR-A)/BUTA(RETURN) SEQUENCE; D<14:03> => RETURN; "1011 0110 1101"=(5555
					4555	K2/99	../..	../..	FI; K2=E77,E89,E92,E95,E102,E106
					5155	K2/99	../..	../..	FI; K2=E6,E12,E18,E24,E30,E33,E39,E42,E45,E51,E54, E60,E66,E69,E75,E77,E81,E89,E92-E93,E96,E106-E107
					5355	K2/99	../..	../..	FI; K2=E92,E107
					5455	K2/99	../..	../..	FI; K2=E3,E6,E12,E18,E24-E25,E30,E33,E39,E45,E51, E54,E60,E66,E69,E75,E81,E89,E92-E93,E96,E98,E101
					5515	K2/99	../..	../..	FI; K2=E3,E6,E12,E18,E24-E25,E30,E33,E39,E45,E51, E54,E60,E66,E69,E75,E81,E86-E87,E89,E96,E98,E101
					5557	K2/99	../..	../..	FI; K2=E9,E26,E86,E114
					7555	K2/99	../..	../..	FI; K2=E72,E77,E92,E102
375)	6702	TEST504G	11762	7406					BUTM(EXFLAG1), EXFLAG<1>M="0"
					7407	K2/99	../..	../..	FI; K2=E112
376)	6704	TEST410B	10814	7434					IR PATTERN LOOP; D<14:00>=ZERO NEGATED CORRECT # TIMES
					7400	K4/65	K3/35	../..	FI; K4=E24,E33; K3=E84
377)	6705	TEST410F	10911	7402					IR PATTERN LOOP; "BYTE-CONSTANT/SECOND-1=OR-2" ASSERTED CORRECT # TIMES
					7400	K4/99	../..	../..	FI; K4=E33
					7401	K3/99	../..	../..	FI; K3=E49,E60
378)	6706	TEST504H	11783	7406					BUTR(FPS05), FPS<05>M="0"
					7407	K3/99	../..	../..	FI; K3=E54
379)	6707	TEST410C	10844	7402					IR PATTERN LOOP; "BYTE-M" ASSERTED/NEGATED CORRECT # TIMES
					7400	K3/65	K5/35	../..	FI; K3=E85; K5=E63
					7401	K6/95	K7/15	../..	FI; K6=E60,E78-E80; K7=E10
380)	6710	TEST503F	11433	7406					BUTM(EXFLAG2), EXFLAG<2>M="0"
					7407	K2/99	../..	../..	FI; K2=E112

Module codes: K1/DCS K2/UWORD K3/IRDECODE K6/DATAPATH K8/KTCACHE K6/TIMING K7/STATUS

#(4)	ERROR code	Symbolic label	Line number	ENUA	TNUA	>Module sequence< #1/##	#2/##	#3/##	Test summary + Print reference + Chip information
381)	6711	TEST410A	10790	7422					IR PATTERN LOOP; D<14:00>=ZERO ONLY ASSERTED 2, TIMES
					7420	K2/89	../..	../..	FI; K2=E40,E88-E89
					7421	K4/65	K3/35	../..	FI; K4=E99; K3=E86
					7424	K2/99	../..	../..	FI; K2=E77,E88-E89,E95,E99,E101,E106
					7436	K2/99	../..	../..	FI; K2=E89-E89,E106
382)	6712	TEST505A	11891	7407					EXFLAG<2> NOT CLEARED BY BUTA(CLR=FLAG-RES=UCON)
					7406	K2/99	../..	../..	K207=ACTIVE-BUT,K210=FLAGS
383)	6713	TEST371A	9538	7401					SR/HOP HAS NO EFFECT ON GUARD="0100"
					7477	K4/99	../..	../..	K40=SR/RES/GUARD-LOGIC
384)	6714	TEST503A	11292	7402					LOAD/READ FLAGS+FPS=(125252) FLAG-PORT PROC=MUX; UCON FUNCTION
					4777	K3/99	../..	../..	FI; K3=E41,E52,E111
					7400	K3/60	K2/40	K7/8	K5/5 FI; K3=E1-E20,E22-E27,E29,E33,E35,E41,E47, E51,E54,E96; K2=E4,E10,E16,E22,E28,E31,E36-E37, E43,E49,E52,E58,E62,E64,E67,E73,E79,E82,E98-E99, E101-E103,E106-E108,E110-E111,E113-E114,E117, E119; K7=E42,E44; K5=E84
					7401	K2/89	K7/20	K5/15	K3/10 FI; K2=E82,E71,E82-E83,E100-E101,E106,E108, E117; K7=E42,E44,E51,E79; K5=E15,E59,E67,E70; K3=E57,E61
					7403	K2/65	K3/25	K4/10	FI; K2=E95,E106,E108,E118; K3=E41; K4=E33
385)	6715	TEST371A	9490	7434					SR/HOP HAS NO EFFECT ON SR=(052525)
					7400	K4/99	../..	../..	FI; K4=E36,E80
					7433	K4/99	../..	../..	FI; K4=E35
386)	6716	TEST504I	11804	7403					BUTM(FLTPT-FD=H), FLTPT-FD=H="1"
					7402	K3/50	K2/50	../..	FI; K3=F64,E66; K2=E119
387)	6717	TEST370D	9467	7401					SR/LEFT-GUARD/ENB SHIFTS GUARD LEFT, GD<3:0>="0100" AFTER
					7400	K4/99	../..	../..	FI; K4=E53
388)	6720	TEST503G	11453	7407					BUTM(EXFLAG1), EXFLAG<1>M="1"
					4756	K3/99	../..	../..	FI; K3=E61
					7406	K2/65	K4/15	K3/15	FI; K2=E112; K4=E86; K3=E54
389)	6721	TEST370C	9429	7434					SR/LEFT-GUARD/ENB, GD<3>="1"; SR=(052525) AFTER
					7477	K4/99	../..	../..	K40=SR/RES/GUARD-LOGIC
390)	6722	TEST505C	11912	7406					EXFLAG<1> CLEARED BY BUTA(CLR-FLAG-RES-UCON)
					7406	K2/99	../..	../..	K207=ACTIVE-BUT,K210=FLAGS

Module codes: K1/DCS K2/UWORD K3/IRDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

```
      ERROR                             Line                  ->Module sequence->
###)  code  Symbolic label             number  ENUA  TNUA  #1/%%  #2/%%  #3/%%  Test summary - Print reference - Chip information
---   ----- ---------------            ------  ----  ----  -----  -----  -----  ---------------------------------------------------

391)  6723  TEST370B                     9405  7402  SR/LEFT-GUARD/ENB SHIFTS GUARD LEFT, GD<3:0>="1010" AFTER
                                                     7400  K4/99  ../..  ../..  FI; K4=E30
                                                     7401  K2/99  ../..  ../..  FI; K2=E83

392)  6724  TEST507B                    12201  7401  BUTR(PS15), PS<15>H="1"
                                                     7402  K5/99  ../..  ../..  FI; K5=E84

393)  6725  TEST370A                     9360  7434  SR LEFT 1, SR=(125252) AFTER; GUARD/ENB="1010" AFTER
                                                     7400  K2/85  K4/10  K7/5   FI; K2=E46,E55,E61,E63,E105; K4=E80,E87; K7=E28
                                                     7421  K6/99  ../..  ../..  FI; K4=E80,E86

394)  6726  TEST500C                    11100  7402  IR PATTERN LOOP; "OVERLAP-L" ASSERTED TO ASPLO CORRECT # TIMES
                                                     7400  K3/80  K4/10  K7/5   K2/5   FI; K3=E46,E59,E80,E84,E87,E89,E116-E117;
                                                                                       K4=E25,E96,E90; K7=E69; K2=E85,E105
                                                     7401  K3/99  ../..  ../..  FI; K3=E46,E87,E59,E80,E84,E87,E89,E121

395)  6727  TEST410D                    10879  7402  IR PATTERN LOOP; "BYTE-CONSTANT/FIRST-1-OR-2" ASSERTED CORRECT # TIMES
                                                     7400  K3/90  K5/5   K4/5   FI; K3=E50,E59-E60,E79,E89,E116-E117; K5=E63,E66;
                                                                                 K4=E15
                                                     7401  K3/96  K4/5   ../..  FI; K3=E32,E36,E46,E49-E50,E59-E60,E79,E89; K4=E38

396)  6730  TEST504B                    11647  7407  BUTM(FLAG7), FLAG<7>H="1"
                                                     7406  K2/99  ../..  ../..  FI; K2=E106

397)  6731  TEST367A                     9317  7401  SR/LOAD-GUARD/DISABLED SETUP BY BUTA(CLR+RES), GUARD NOT ALTERED
                                                     74??  K4/99  ../..  ../..  K408=SR/RES/GUARD-LOGIC

398)  6732  TEST503H                    11474  7407  BUTR(FP806), FPS<05>H="1"
                                                     7406  K2/99  ../..  ../..  FI; K2=E7,E13,E25,E110,E117

399)  6733  TEST366C                     9278  7434  SR LEFT 1, D[C]="1"; SR=(025251) AFTER; (GUARD/DIS)
                                                     7400  K4/99  ../..  ../..  FI; K4=E36
                                                     7402  K4/99  ../..  ../..  FI; K4=E45
                                                     7420  K4/99  ../..  ../..  FI; K4=E38
                                                     7421  K4/99  ../..  ../..  FI; K4=E52,E71,E86

400)  6734  TEST506D                    12076  7406  BUTM(MASK-PS[T]), MASK-PS[T]=0:H="0"
                                                     7407  K2/99  ../..  ../..  FI; K2=E72,E79,E108,E112

401)  6735  TEST366B                     9256  7401  CHECK GUARD NOT ALTERED ON SR/LEFT-GUARD/DISABLED SHIFT
                                                     7402  K4/99  ../..  ../..  FI; K4=E30,E80,E86

402)  6736  TEST500E                    11164  7402  IR PATTERN LOOP; BUTR(PREFETCH(0)H) ASSERTED CORRECT # OF TIMES
                                                     7400  K3/99  ../..  ../..  FI; K3=E32,E34,E38,E44,E48,E80
                                                     7401  K3/90  K2/10  ../..  FI; K3=E32,E39,E44,E48,E80; K2=E105
```



```
      ERROR                             Line                  ->Module sequence->
###)  code  Symbolic label             number  ENUA  TNUA  #1/%%  #2/%%  #3/%%  Test summary - Print reference - Chip information
---   ----- ---------------            ------  ----  ----  -----  -----  -----  ---------------------------------------------------

403)  6737  TEST366A                     9209  7434  SR LEFT 1, D[C]="0"; SR=(112524) AFTER; (GUARD/DIS)
                                                     7005  K4/99  ../..  ../..  FI; K4=E80
                                                     7400  K4/99  ../..  ../..  FI; K4=E45
                                                     7421  K4/99  ../..  ../..  FI; K4=E52,E71,E86

404)  6740  TEST504C                    11667  7403  BUTR(FLTPT-PROC-H), FLTPT-PROC-H="1"
                                                     7402  K2/65  K3/35  ../..  FI; K2=E101,E106,E119; K3=E64

405)  6741  TEST365B                     9187  7401  CHECK GUARD NOT ALTERED ON SR/RIGHT-GUARD/DISABLED SHIFT
                                                     7402  K4/99  ../..  ../..  FI; K4=E30,E53,E80

406)  6742  TEST503T                    11495  7402  BUTR(FLTPT-FD-H), FLTPT-FD-H="0"
                                                     7403  K3/99  ../..  ../..  FI; K3=E64
                                                     7756  K2/99  ../..  ../..  FI; K3=E40

407)  6743  TEST365A                     9146  7434  SR RIGHT 1, D<00>="1"; SR=(145252) AFTER
                                                     7400  K4/99  ../..  ../..  FI; K4=E10,E17,E36,E45
                                                     7420  K4/99  ../..  ../..  FI; K4=E52

408)  6744  TEST506C                    12054  7407  BUTM(PS03), PS<03>H="1"
                                                     7406  K2/80  K3/20  ../..  K210=MULTIPLE-BUT; K306=PS[CC]

409)  6745  TEST364B                     9119  7401  BUT(GD3-2), GD<3:0>="0101" AFTER SR/RIGHT, SR<00>="0"
                                                     74??  K4/99  ../..  ../..  K408=SR/RES/GUARD-LOGIC

410)  6746  TEST500D                    11136  7402  IR PATTERN LOOP; "OVERLAP-L" ASSERTED SAME # TIMES TO ASPLO AND BSPLO
                                                     7401  K4/99  ../..  ../..  FI; K4=E99

411)  6747  TEST364A                     9091  7402  BUT(GD3-2), GD<3:0>="1010" AFTER SR/RIGHT, SR<00>="1"
                                                     74??  K4/99  ../..  ../..  K408=SR/RES/GUARD-LOGIC

412)  6750  TEST504D                    11689  7475  BUT(INSTR1), IR=(172525), CLASS=A(FLTPT), FLAG<4:5>H="01"
                                                     74??  K3/99  ../..  ../..  K305=FLAGS/FP-INSTR1

413)  6751  TEST363B                     9069  7401  BUT(GD3-2), GD<3:0>="0100" AFTER SR/RIGHT, SR<00>="0"
                                                     7400  K3/60  K4/40  ../..  FI; K3=E54; K4=E53
                                                     7403  K4/99  ../..  ../..  FI; K4=E53

414)  6752  TEST534G                    13846  7402  M.M. COUNTER CLEARED TO ZEROS BY BUTA(LAST)
                                                     7400  K4/65  K2/35  ../..  FI; K4=E87,E90-E91; K2=E55
                                                     7637  K2/70  K3/30  ../..  FI; K2=E22-E24,E49-E51,E73-E75; K3=E6,E9,E11,E19

415)  6753  TEST363A                     9033  7434  SR RIGHT 1, D<00>="0"; SR=(052525) AFTER
                                                     7400  K4/99  ../..  ../..  FI; K4=E10,E17,E24,E45,E74,E85,E94
                                                     7410  K4/99  ../..  ../..  FI; K4=E47                            [Continued]
```

###	ERROR code	Symbolic label	Line number	ENUA	TNUA	->Module sequence-> #1/%%	#2/%%	#3/%%	Test summary + Print reference + Chip information
	6753	TFST3838	[Continued]		7417	K4/99	../..	../..	FI; K4=E36
					7420	K4/99	../..	../..	FI; K4=E52
					7426	K4/99	../..	../..	FI; K4=E35
416)	6754	TFST504B	12034	7402					BUTR(PS15), PS<15>H="0"
					7403	K3/99	../..	../..	FI; K3=E54
417)	6755	TEST3679	9010	7402					BUT(GD1-2), GD<3:0>="1000" AFTER SR/RIGHT, SR<00>="1"
					7400	K4/80	K3/15	K2/5	FI; K4=E26,E28-E30,E47,E53,E55,E71,E80,E86; K3=E1-E3,E13-E15,E56; K2=E39,E54
418)	6756	TFST503J	11513	7406					BUTN(D00), D=(000000)
					7407	K2/99	../..	../..	FI; K2=E112
419)	6757	TEST3620	8989	7405					BUTR(SR1-08COUNT), SP<11>="10"
					7417	K2/99	../..	../..	FI; K2=E46,E55
					7477	K4/99	../..	../..	K400=SR/REG/GUARD-LOGIC
420)	6760	TEST504E	11716	7402					EXFLAGS<2:1>H="10", READ THRU CUA-PORT, PROC-MUX
					7400	K2/99	../..	../..	FI; K2=E103,E106,E111-E112
421)	6761	TEST702R	15089	7402					LOAD BA<17:16>="10", 18, BIT MODE, READ THRU STATUS-MUX(SERVICE)<9:8>
					7400	K6/55	K4/35	K7/10	K5/10 FI; K6=E11,E51-E54; K4=E45; K7=E56; K5=E26, E60
422)	6762	TEST520D	13417	7402					INSTR-BRANCH-L="L", IR=(101401)=BLOS, PS[NZVC]=(01)
					7407	K3/99	../..	../..	K2=PS[CC]-BRANCH(E64,72-73,83)
423)	6763	TFST362R	8962	7434					SR=(125252), D[C]="0"; SR/XMUX=(000052) (FLTPT)
					7400	K4/99	../..	../..	FI; K4=E48,E55,E78
					7420	K4/99	../..	../..	FI; K4=E37
					7421	K4/99	../..	../..	FI; K4=E44
					7424	K4/99	../..	../..	FI; K4=E44
424)	6764	TEST520B	13366	7402					INSTR-BRANCH-L="L", IR=(063000)=BGT, PS[NZVC]=(00)
					7403	K3/99	../..	../..	FI; K3=E84,E83
					7407	K3/99	../..	../..	K2=PS[CC]-BRANCH(E64,72-73,83)
425)	6765	TFST3624	8917	7434					SR RIGHT 1, D<00>="1"; SR=(125252) AFTER
					7400	K4/99	../..	../..	FI; K4=E3,E13,E22,E24,E29,E35-E36,E45,E52,E55,E74, E76,E85,E94,E104
426)	6766	TERTA504A	11621	7402					LOAD/READ FLAG=FPS=(052525) FLAG-PORT PROC-MUX; UCON FUNCTION
					7400	K2/90	K3/10	../..	FI; K2=E78,E98-E99,E101-E103,E105-E108,E110-E114, E119; K3=E27,E47,E94,E96
					7401	K2/99	../..	../..	FI; K2=E106,E108

Module codes: K1/DCS K2/UWORD K3/IRDECODE K4/DATAPATH K5/RTCACHE K6/TIMING K7/STATUS

###	ERROR code	Symbolic label	Line number	ENUA	TNUA	->Module sequence-> #1/%%	#2/%%	#3/%%	Test summary + Print reference + Chip information
427)	6767	TEST3618	8894	7400					BUT(GD3-2), GD<3:0>="0080" AFTER ENABLED, SR/LOAD
					7401	K4/80	K2/15	K3/5	FI; K4=E17,E28,E30,E53,E60,E86-E87; K2=E33,E45, E51; K3=E54
					7402	K4/90	K3/10	../..	FI; K4=E26,E30,E53,E87; K3=E56
					7404	K3/99	../..	../..	FI; K3=E74
428)	6770	TEST701C	15981	7402					LOAD BA<19:00>=(125252), READ THRU BA/KT-ALU/PBA/STATUS-MUX(PBA)
					7400	K5/50	K4/15	K6/15	K7/15 FI; K5=E7,E9-E11,E27,E37-E38,E43,E45-E47, E80,E59,E69-E70,E72,E79-E82,E94; K4=E39,E48,E57; K6=E19,E25,E29,E33,E49,E59; K7=E56,E64-E65,E73, E80-E82,E88
					7401	K7/50	K4/50	../..	FI; K7=E88; K6=E29
					7403	K5/65	K6/20	K7/10	FI; K5=E7,E26-E27,E46-E47; K6=E25,E59; K7=E80
429)	6771	TEST361D	8873	7403					BUTR(SP1-08COUNT), SR<1:00>="01"
					6377	K2/65	K3/35	../..	FI; K2=E4-E5,E58-E59; K3=E1,E16
					7407	K3/99	../..	../..	FI; K3=E74
					7477	K4/99	../..	../..	K400=SR/REG/GUARD-LOGIC
430)	6772	TFST520C	13388	7403					INSTR-BRANCH-L="H", IR=(002416)=BLT, PS[NZVC]=(16)
					7402	K3/99	../..	../..	FI; K3=E72,E83
					7407	K3/99	../..	../..	K2=PS[CC]-BRANCH(E64,72-73,83)
431)	6773	TFNTD410	10707	7361					(NUA SEQUENCING LOGIC ERROR)
					6777	K2/99	../..	../..	FI; K2=E48
					7777	K1/99	../..	../..	INTERNAL DCS ERROR
432)	6775	TFST520F	13445	7403					INSTR-BRANCH-L="H", IR=(103406)=BLO, PS[NZVC]=(06)
					7402	K3/99	../..	../..	FI; K3=E83
					7407	K3/99	../..	../..	K2=PS[CC]-BRANCH(E64,72-73,83)

1
2
3

```
;
;
;                       I D E N T I F I C A T I O N
;                       ---------------------------
;
;
;       PRODUCT CODE:   MAINDEC-11-DQKUB-B0
;
;       PRODUCT NAME:   KD11-K Microdiagnostic
;
;       MAINTAINER:     Diagnostic Engineering
;
;       AUTHOR:         Don North
;
;       DATE CREATED:   14-January-1977
;
;       LAST REVISION:  15-June-1977,  Version /101/
;
;
;
;
;
;
;
;       COPYRIGHT (C) 1976, 1977;  DIGITAL EQUIPMENT CORPORATION
;                                  146 MAIN STREET
;                                  MAYNARD, MASSACHUSETTS, USA
;                                  01754      617-897-5111
;
;       THIS SOFTWARE IS FURNISHED TO THE PURCHASER UNDER A LICENSE FOR
;       USE ON A SINGLE COMPUTER SYSTEM, AND CAN BE COPIED (WITH INCLU-
;       SION OF DIGITAL'S COPYRIGHT NOTICE) ONLY FOR USE IN SUCH SYSTEM,
;       EXCEPT AS MAY OTHERWISE BE PROVIDED IN WRITING BY DIGITAL.
;
;       THE INFORMATION  IN THIS DOCUMENT  IS SUBJECT TO CHANGE WITHOUT
;       NOTICE,  AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL
;       EQUIPMENT CORPORATION, DIGITAL EQUIPMENT CORPORATION ASSUMES NO
;       RESPONSIBILITY FOR ANY ERRORS THAT MAY APPEAR IN THIS DOCUMENT.
;
;       DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF
;       ITS SOFTWARE ON EQUIPMENT NOT SUPPLIED BY DIGITAL.
;
;
;
;
;
;
;
;
;
;
;
```

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

```
!.PAGE=====================================================================================================

.TOC    * IDENTIFICATION
!
.TITLE KD11-K MICRODIAGNOSTIC




          !*****************************************************************************************!
          !                                                                                         !
          !                         K     K  DDDDDD         1     1          /  K     K             !
          !                         K    K   D     D       11    11         /   K    K              !
          !                         K   K    D     D      1 1   1 1        /    K   K               !
          !                         KKKK     D     D XXX    1     1       /     KKKK                !
          !                         K   K    D     D        1     1      /      K   K               !
          !                         K    K   D     D        1     1     /       K    K              !
          !                         K     K  DDDDDD       11111 11111  /        K     K             !
          !                                                                                         !
          !                                                                                         !
          !                DDDDDD   IIIII    AAA     GGGGG   N     N  OOOOO    SSSSS  TTTTTTT IIIII  CCCCC !
          !                D     D    I     A   A   G     G  NN    N O     O  S     S    T      I   C     C !
          !   U     U      D     D    I    A     A  G        N N   N O     O  S          T      I   C       !
          !   U     U      D     D    I    A     A  G  GGG   N  N  N O     O   SSSSS     T      I   C       !
          !   U     U      D     D    I    AAAAAAA  G     G  N   N N O     O        S    T      I   C       !
          !   U     U      D     D    I    A     A  G     G  N    NN O     O  S     S    T      I   C     C !
          !   U UUUUU U    DDDDDD   IIIII  A     A   GGGGG   N     N  OOOOO    SSSSS     T    IIIII  CCCCC !
          ! U                                                                                       !
          ! U                                                                                       !
          !                                                                                         !
          !*****************************************************************************************!



!.PAGE=====================================================================================================

.TOC    * REVISION HISTORY

.IDENT /V101A0/

        REV-NUMBER          ::=  000101         ;NO BIT15 DURING EXECUTION
        REV-NUMBER-AND-B15  ::=  100101         ;BUT SET W/REV. NUMBER AT EOP ONLY




!.PAGE=====================================================================================================

.TOC    * MICROWORD FIELD DEFINITIONS

!       NOTE:   THE FOLLOWING ARE THE ASSIGNED RANGES OF THE
```

```
                ;               MICROWORD FIELD BIT DEFINITIONS USED IN THIS
                ;               SOURCE LISTING:
                ;
                ;               BITS(NUMBER)            WHERE HELD
                ;               ------------            ----------
                ;
                ;               (47:00) 48/48           MAIN MACHINE ROM,
                ;                                       DCS MAIN ROM
                ;               (59:48) 12/12 *         MAIN MACHINE ROM EXTENSION,
                ;                                       USES 12/12 BITS
                ;               (54:48) 7/12 #          DCS 4-BIT ROM EXTENSION,
                ;                                       USES 7/12 BITS, (54:48)
                ;
                ;       * = NOTE OVERLAP OF SM EXTENSION/DCS EXTENSION BITS,
                ;               THESE BITS ARE MUTUALLY EXCLUSIVE.


1,PAGE==========================================================================

.TOC    * MICROWORD BIT LAYOUT
        BASE            1=EMIT                          6=RETURN                                                UCON
        MACHINE         2=SHIFT=TREE    4=UCON=DATA     7=PAGING        9=RES=BIT6      DCS             PROCESSOR
        CONTROL         3=RESIDUAL=CTL  5=CSP=ADDRESS   8=UCON=CONTROL  10=MULTIPLE     EXTENSION       CONTROL
        --------        --------------  -------------   ---------------  -----------    -------------   ------------
!U47    ALU3            1=EMITH15       4=UCONL10                                       DCS=CTR3        PS<3:0>=CLK
!U46    ALU2            1=EMITH14       4=UCON=I/O=SEL  6=RETR11        9=MISMUX=SEL1   DCS=CTR2
!U45    ALU1            1=EMITH13       4=UCON=WCS=SEL  6=RETR10        9=SMS1=L        DCS=CTR1
!U44    ALU0            1=EMITH12       4=UCON=KT=SEL   6=RETR09        9=SMS0=L        DCS=CTR0
!U43    SEN1
!U42    SEN0                            4=UCONL09                                                       UBREAK=CLK
!U41    BSEL1           1=EMITH11       4=UCONL08       6=RETR08        9=GUARD=EN=H    ENUA11            <MU>
!U40    BSEL0           1=EMITH10       4=UCONL07       6=RETR07                        ENUA10            <MU>
!U39    AEN1            1=EMITH09       4=UCONL06       6=RETR06                        ENUA09          SEL=MSMUX1L
!U38    AEN0            1=EMITH08       4=UCONL05       6=RETR05                        ENUA08          SEL=MSMUX0L
!U37    ASEL1           1=EMITL07                       6=RETR04                        ENUA07
!U36    ASEL0           1=EMITL06       4=UCON=PROC=SEL 6=RETR03                        ENUA06
!U35    RTF2            1=EMITL05       4=UCONH12       6=RETR02                        ENUA05          FPS<7:4>=CLK
!U34    RTF1            1=EMITL04       4=UCONH11       6=RETR01                        ENUA04          PS<7:4>=CLK
!U33    RTF0            1=EMITL03       4=UCON=FP=SEL   6=RETR00                        ENUA03
!U32    COUT2           1=EMITL02       4=UCONH15       7=NEXT=PAGE2    10=MULT=SEL2    ENUA02          IR=CLOCK
!U31    COUT1           1=EMITL01       4=UCONH14       7=NEXT=PAGE1    10=MULT=SEL1    ENUA01          PS<15:12>=CLK
!U30    COUT0           1=EMITL00       4=UCONH13       7=NEXT=PAGE0    10=MULT=SEL0    ENUA00          FLAG<8:0>=CLK
!U29 * WMEM
!U28 * CLK=D
!U27 * CLK=SR
!U26 * CLK=RA
!U25 * SET=CC
!U24 * BEGIN
!U23    SELECT(=0)      2=BMUX          5=CSPADR3       8=SELECT(=1)
!U22    BUSCOD2         2=AMUX2         5=CSPADR2       8=FLPGO
!U21    BUSCOD1         2=AMUX1         5=CSPADR1       8=UCON=ENER
!U20    BUSCOD0         2=AMUX0         5=CSPADR0       8=UCON=LOAD
!U19 * WRCSP
```



```
!U18    HI/LO           3=LOAD=RES
!U17    WRSEL
!U16    WRB             3=LOAD=COUNT
!U15    WRA
!U14 * MOD(=0)          3=MOD(=1)
!U13 * UPF4
!U12 * UPF3
!U11 * UPF2
!U10 * UPF1
!U09 * UPF0
!U08 * UPF8
!U07 * UPF7
!U06 * UPF6
!U05 * UPF5
!U04 * UPF4
!U03 * UPF3
!U02 * UPF2
!U01 * UPF1
!U00 * UPF0             ( * = DEDICATED TO THE CORRESPONDING SINGLE FUNCTION )


1,PAGE==========================================================================

.TOC    * A & B & C SCRATCHPAD LAYOUT, DCS SPECIFIC
;
;       THE USE OF THE A, B, & C SCRATCHPADS AS TEMPORARY STORAGE AREAS
;       IS OUTLINED BELOW.   NOTE THAT IN MOST CASES, THE REGISTERS
;       EMPLOYED HAVE NO RESEMBLANCE  TO ANY SIMILAR NAMED REGISTER
;       IN THE KD11-F BASE MACHINE MICROCODE, EITHER LIVING OR DEAD.
;
;       THE BELOW DEFINITIONS ARE -DCS SPECIFIC- ONLY.
;
;
;       OCTAL           ASPHI           ASPLO           BSPHI           BSPLO           CSP
;       ADDRESS
;
;         00            . . .           . . .           . . .           . . .           <RETURN>
;
;         01            (000000)        . . .           (000000)        . . .           TEMP/(000152)
;
;         02            . . .           . . .           . . .           . . .           TEMP/(000125)
;
;         03            (177777)        . . .           (177777)        . . .           TEMP
;
;         04            . . .           . . .           . . .           . . .           TEMP/(125200)
;
;         05            (125252)        . . .           (125252)        . . .           TEMP/(052522)/(170360)
;
;         06            . . .           . . .           . . .           . . .           TEMP/(053433)/(007417)
;
;         07            (052525)        . . .           (052525)        . . .           TEMP
;
;         10            . . .           . . .           . . .           . . .           (052525)
```

```
!
!         11          (000001)      . . .       (000001)      . . .       (125252)
!
!         12          . . .         . . .       . . .         . . .       (177777)
!
!         13          (100000)      . . .       (100000)      . . .       (000000)
!
!         14          . . .         . . .       . . .         . . .       TEMP/(000100)/(087077)
!
!         15          (000200)      . . .       (000200)      . . .       TEMP/(000077)/(177067)
!
!         16          . . .         . . .       . . .         . . .       TEMP/(170000)
!
!         17          TEMPAHI       TEMPALO     TEMPBHI       TEMPBLO     TEMP/(007700)
!
!
!
!       NOTES:  (XXXXXX) ::= A REGISTER W/ A CONSTANT VALUE, NAMED:
!               CXXXXXX-A, IF ON THE A-SIDE, OR
!               CXXXXXX-B, IF ON THE B-SIDE
!



!.PAGE=========================================================================================================

.TOC    * MICROWORD FIELD SPECIFICATION


!----------------------------------------------------------------------------------

.TOC    *   MICROWORD FIELD FORMAT

.RADIX 8                        ! ALL NUMBERS ARE OCTAL, UNLESS OTHERWISE NOTED

.WIDTH 64R                      ! MICROWORD IS 64\10 BITS WIDE, BIT <00> IS RIGHTMOST BIT

.BOUNDS         [4000:7777]             ! ADDRESSES ARE 12 BITS, ON PAGES 4:7

!.OBJECT  <15:00>"<31:16>"<47:32>"<63:48>              ! OUTPUT FORMAT (DEFAULT SPEC)

!----------------------------------------------------------------------------------

.TOC    *   NULL FIELD/MACRO SPECIFICATION

.FIELD N::=<63>
.MACRO NULL::=N/0


.TOC    *   ALU AND INTERNAL DATA BUS CONTROL



.TOC    *     <ALU>-ALU FUNCTION CONTROL BITS
```



```
!SPECIFIES ALU FUNCTION CODE AND CINMUX SELECT, ALWAYS IN EFFECT.
.FIELD ALU::=<47:44>
!                                   ---FUNCTION---     LOGIC/ARITH  ALUS<3:0> H    CINMUX L
        NOT-A::=00                  !COMPLEMENT A,          L           0000        =1
        A-PLUS-B-PLUS-PS(C)::=01    !ADD,                   A           1001        =PS(C)
        NOT-A-AND-B::=02            !AND,                   L           0010        =PS(C)
        ZERO::=03                   !ZERO,                  L           0011        =PS(C)
        A-PLUS-B-PLUS-D(C)::=04     !PLUS,                  A           1001        =D(C)
        A-PLUS-NOT-B-PLUS-D(C)::=05 !PLUS,                  A           0110        =D(C)
        A-XOR-B::=06                !XOR,                   L           0110        =D(C)
        A-AND-NOT-B::=07            !AND,                   L           0111        =D(C)
        DIVIDE::=10                 !DIVIDE STEP,
                                    !SUB,   IF D(C)H=1      A           0110        =D(C)=-1
                                    !ADD,   IF D(C)H=0      A           1001        =D(C)=-0
        A-PLUS-B::=11               !PLUS,                  A           1001        =0
        B::=12                      !SELECT B,              L           1010        =0
        A-AND-B::=13                !AND,                   L           1011        =0
        A-PLUS-B-PLUS-1::=14        !PLUS,                  A           1001        =1
        A-MINUS-B::=15              !MINUS,                 A           0110        =1
        A-IOR-B::=16                !IOR,                   L           1110        =1
        A::=17                      !SELECT A,              L           1111        =1



.TOC    *     <BEN>-B-BUS DATA SOURCE
!SPECIFIES GATING OF DATA ONTO B-BUS, ALWAYS IN EFFECT.
.FIELD BEN::=<43:42>
        BSPLO::=0                   !DIRECT BSP LOCATIONS 00-17
        BSPHI::=1                   !DIRECT BSP LOCATIONS 20-37
        CSP::=2                     !USE <CSPADDR> [SIC] AS ADDRESS (4 BIT)
        BASCON::=3                  !1 OF 4 BASE CONSTANTS IN CSP17 TO CSP14 (2 BIT)



.TOC    *     <BSEL>-B-BUS SOURCE SELECTION CONTROL
!SPECIFIES CONTROL OF INDIVIDUAL B-BUS SOURCES, ALWAYS IN EFFECT.
.FIELD BSEL::=<41:40>
!NOT USED WHEN BEN/CSP
!CSP17 TO CSP14 IMMEDIATE ADDRESS WHEN BEN/BASCON
        B17::=0                     !
        B16::=1                     !
        B15::=2                     !
        B14::=3                     !
!USED IN CONJUNCTION WITH <RIF> FOR SP ADDRESS WHEN BEN/BSPLO OR BEN/BSPHI
        DF::=0                      !DESTINATION FIELD
        SF::=1                      !SOURCE FIELD

        IMMED0::=2                  !DIRECT ADDRESS, LOW BIT=0
          R00::=2                   !FOR JOINT USE W/ RIF FIELD
          R02::=2                   !
          R04::=2                   !
          R06::=2                   !
          R10::=2                   !
          R12::=2                   !
          R14::=2                   !
```

```
            R16;;=2                 !
            IMMED1;;=3              !DIRECT ADDRESS, LOW BIT=1
              R01;;=3               !FOR JOINT USE W/ RIF FIELD
              R03;;=3               !
              R05;;=3               !
              R07;;=3               !
              R11;;=3               !
              R13;;=3               !
              R15;;=3               !
              R17;;=3               !
            C000000;;=3             !ASPHI/BSPHI CONSTANTS
            C177777;;=3             !
            C125252;;=3             !
            C052525;;=3             !
            C000001;;=3             !
            C100000;;=3             !
            C000200;;=3             !


.TOC    *     <AEN>-A-BUS DATA SOURCE
!SPECIFIES GATING OF DATA ONTO A-BUS, ALWAYS IN EFFECT.
.FIELD AEN;;=<39:38>
            XMUX;;=0                !XMUX=BR OR FLTPT ASSEMBLE
            CMUX;;=1                !SHIFT TREE
            ASPLO;;=2               !DIRECT ASP LOCATIONS 00-17
            ASPHI;;=3               !DIRECT ASP LOCATIONS 20-37


.TOC    *     <ASEL>-A-BUS SOURCE SELECTION CONTROL
!SPECIFIES CONTROL OF INDIVIDUAL A-BUS SOURCES, ALWAYS IN EFFECT.
.FIELD ASEL0;;=<36>
!XMUX CONTROL WHEN AEN/XMUX  [USES ASEL0 ONLY]
            SR;;=0                  !SR OUTPUT ONTO BUS-A
            FLTPT;;=1               !FLTPT-ASSEMBLE ONTO BUS-A
.FIELD ASEL;;=<37:36>
!CMUX CONTROL WHEN AEN/CMUX.  SHIFTS CMUX INPUT APPROPRIATE AMOUNT
            LEFT-1;;=0              !LOW BIT GETS SEN=MUX OUTPUT
            DIRECT;;=1              !OUTPUT=INPUT
            RIGHT-1;;=2             !HIGH BIT GETS D[C]
            RIGHT-2;;=3             !HIGH BITS BOTH GET D[C]
!USED IN CONJUNCTION WITH <RIF> FOR SP ADDRESS WHEN AEN/ASPLO OR AEN/ASPHI
            IMMED0;;=0              !DIRECT ADDRESS, LOW BIT=0
              R00;;=0               !FOR JOINT USE W/ RIF FIELD
              R02;;=0               !
              R04;;=0               !
              R06;;=0               !
              R10;;=0               !
              R12;;=0               !
              R14;;=0               !
              R16;;=0               !
            IMMED1;;=1              !DIRECT ADDRESS, LOW BIT=1
              R01;;=1               !FOR JOINT USE W/ RIF FIELD
```



```
              R03;;=1               !
              R05;;=1               !
              R07;;=1               !
              R11;;=1               !
              R13;;=1               !
              R15;;=1               !
              R17;;=1               !
            C000000;;=1             !ASPHI/BSPHI CONSTANTS
            C177777;;=1             !
            C125252;;=1             !
            C052525;;=1             !
            C000001;;=1             !
            C100000;;=1             !
            C000200;;=1             !
            DF;;=2                  !DESTINATION FIELD
            SF;;=3                  !SOURCE FIELD


.TOC    *     <RIF>-ASP, BSP REGISTER IMMEDIATE FIELD
!SPECIFIES ADDRESSES WITH ASP, BSP ALONG WITH AEN, ASEL & BEN, BSEL
.FIELD RIF;;=<35:33>
            R00-OR-01;;=4           !LOW BIT IS 0/1, SPECIFIED BY
              R00;;=4               !
              R01;;=4               !
            R02-OR-03;;=5           !USING EITHER IMMED0/IMMED1 MODES
              R02;;=5               !
              R03;;=5               !
            R04-OR-05;;=6           !
              R04;;=6               !
              R05;;=6               !
            R06-OR-07;;=7           !
              R06;;=7               !
              R07;;=7               !
            R10-OR-11;;=0           !
              R10;;=0               !
              R11;;=0               !
            R12-OR-13;;=1           !ADDR<3:0>H = =RIF<2>H # RIF<1:0>H # A/BSEL<0>H
              R12;;=1               !
              R13;;=1               !
            R14-OR-15;;=2           !
              R14;;=2               !
              R15;;=2               !
            R16-OR-17;;=3           !
              R16;;=3               !
              R17;;=3               !

            C000000;;=4             !ASPHI/BSPHI CONSTANTS
            C177777;;=5             !
            C125252;;=6             !

            C052525;;=7             !
            C000001;;=0             !
            C100000;;=1             !
            C000200;;=2             !
```

```
433
434
435     .TOC    *       <COUT>-CARRY OUT BIT MUX SELECTION
436     ;SPECIFY INPUT TO D[C] REGISTER, LOADED WHEN D REGISTER LOADED,  ALWAYS IN EFFECT.
437     .FIELD COUT[:]=<32:30>
438             CIN[:]=0                ;OUTPUT OF CINMUX [SIC]
439             PS[C][:]=1              ;PS C-BIT
440             ALU00[:]=2              ;ALU OUTPUT BIT 00
441             ALU07[:]=3              ;ALU OUTPUT BIT 07
442             ALU15[:]=4              ;ALU OUTPUT BIT 15
443             COUT07[:]=5             ;BYTE CARRY BIT
444             COUT15[:]=6             ;WORD CARRY BIT
445             D[C][:]=7               ;PROPOGATE [SAVE] LAST D[C]
446
447     ;-----------------------------------------------------------------------------
448
449
450
451     .TOC    *   CLOCKS
452
453
454
455     .TOC    *       <WHEN>-D/SR WHEN TO CLOCK
456     ;SPECIFY CLOCK D/SR REGISTERS AT P2 T OR P3 T, ALWAYS IN EFFECT.
457     .FIELD WHEN[:]=<29>,0
458             AT-P2-T[:]=0            ;[CLOCK D AND/OR SR AT P2 T[100 NS],
459             AT-P3-T[:]=1            ;[CLOCK D AND/OR SR AT P3 T[150 NS],
460
461
462
463     .TOC    *       <CLKD>-ENABLE D-REGISTER CLOCKING
464     ;ENABLES CLOCKING OF D-REGISTER, ALWAYS IN EFFECT.
465     .FIELD CLKD[:]=<28>,0
466             NO[:]=0                 ;NOP
467             YES[:]=1                ;CLOCK D[C], D-REGISTER AT <WHEN>
468
469
470
471     .TOC    *       <CLKSR>-ENABLE SR-REGISTER CLOCKING
472     ;ENABLES CLOCKING OF SR-REGISTER, ALWAYS IN EFFECT.
473     .FIELD CLKSR[:]=<27>,0
474             NO[:]=0                 ;NOP
475             YES[:]=1                ;CLOCK SR-REGISTER AT <WHEN>
476
477
478
479     .TOC    *       <CLKBA>-ENABLE CLOCKING OF BA-REGISTER
480     ;ENABLES CLOCKING OF BA-REGISTER AT P1 T[60 NS], ALWAYS IN EFFECT.
481     .FIELD CLKBA[:]=<26>,0
482             NO[:]=0                 ;NOP
483             YES[:]=1                ;CLOCK BA-REGISTER AT P1 T[60 NS],
484
485
486
```



```
487     .TOC    *       <SCC>-ENABLE SETTING OF PS CONDITION CODES
488     ;ENABLE CLOCKING OF PS CONDITION CODES AT P2 T[100 NS] OF NEXT UWORD,  D MUST
489     ;BE CLOCKED AT P2 T OR EARLIER OF PREVIOUS MICROWORD,   ALWAYS IN EFFECT.
490     .FIELD SCC[:]=<25>,0
491             NO[:]=0                 ;NOP
492             YES[:]=1                ;ENABLE CLOCKING IN NEXT UWORD
493
494     ;-----------------------------------------------------------------------------
495
496
497
498     .TOC    *   BUS/UCON & CSP-ADDRESS & SHIFT-TREE CONTROL
499
500
501
502     .TOC    *       BUS/UCON CONTROL
503
504
505     .TOC    *           <BEGIN>-BEGIN BUS/UCON OPERATION
506     ;INITIATE BUS XOR UCON OPERATION, ALWAYS IN EFFECT.
507     .FIELD BEGIN[:]=<24>,0
508             NO[:]=0                 ;NOP FOR BUS AND UCON OPERATIONS
509             YES[:]=1                ;BEGIN OPERATION SPECIFIED
510
511
512
513     .TOC    *           <SELECT>-SELECT BUS OR UCON
514     ;SELECT BUS XOR UCON, ONLY USED IF BEGIN/YES,
515     .FIELD SELECT[:]=<23>
516             BUS[:]=0                ;SELECT BUS
517             UCON[:]=1               ;SELECT UCON
518
519
520
521     .TOC    *           BUS CONTROL
522
523
524     .TOC    *             <BUSCODE>-BUS CODE ACTION FIELD
525     ;BUS ACTION CODES, ONLY USED IF BEGIN/YES & SELECT/BUS,
526     .FIELD BUSCODE[:]=<22:20>
527             DATI-CLKIR[:]=0         ;DATA IN, LOAD IR
528             DATI-NOINT[:]=1         ;DATA IN, NO INTERNAL ADDRESS
529             DATO[:]=2               ;DATA OUT
530             DATIB[:]=3              ;DATA IN, ALLOW; ODD ADDRESS
531             DATIB[P][:]=3           ;DATA IN, ALLOW; ODD ADDRESS, FORCE TO PAUSE
532             DATIP[:]=4              ;DATA IN, NO CACHE, LOCK BUS

533             DATOB[:]=5              ;DATA OUT, ALLOW; ODD ADDRESS
534             DATI[:]=6               ;DATA IN
535             DATI[P][:]=6            ;DATA IN, ALLOW; FORCE TO PAUSE
536             INVALIDATE[:]=7         ;INVALIDATE CACHE  LOCATION FUNCTION
537
538
539
540     .TOC    *       UCON CONTROL
```

```
541
542
543
544     .TOC    *         <FLPGO>-START MOT FLOATING POINT
545     !INITIATES MOT FLOATING POINT FUNCTION,  ONLY USED IF BEGIN/YES & SELECT/UCON,
546     .FIELD FLPGO::=<22>
547             NO::=0                  !NOP
548             YES::=1                 !YELL GO
549
550
551
552     .TOC    *         <UCON-XFER>-UCON OPERATION
553     !EXECUTE A UCON FUNCTION, ONLY USED IF BEGIN/YES & SELECT/UCON,
554     .FIELD UCON-XFER::=<21>
555             NO::=0                  !NOP
556             YES::=1                 !START UCON OPERATION
557
558
559
560     .TOC    *         <UCON-LOAD>-LOAD UCON REGISTER
561     !LOAD UCON CONTROL REGISTER, ONLY USED IF BEGIN/YES & SELECT/UCON,
562     .FIELD UCON-LOAD::=<20>
563             NO::=0                  !NOP
564             YES::=1                 !LOAD UCON CONTROL REGISTER
565
566
567
568     .TOC    *     CSP ADDRESS SPECIFICATION
569
570
571     .TOC    *       <CSPADDR>-CSP IMMEDIATE ADDRESS
572     !SPECIFY CSP 4 BIT ADDRESS, ONLY USED IF BEN/CSP,
573     .FIELD CSPADDR::=<23:20>
574             D00::=17                !NOTE INVERSION
575             D01::=16                !
576             D02::=15                !
577             D03::=14                !
578             D04::=13                !
579             D05::=12                !
580             D06::=11                !
581             D07::=10                !
582             D10::=07                !
583             D11::=06                !
584             D12::=05                !
585             D13::=04                !
586             D14::=03                !
587             D15::=02                !
588             D16::=01                !
589             D17::=00                !
590             C052525::=07            !CSP/DCS CONSTANTS
591             C125252::=06            !
592             C177777::=05            !
593             C000000::=04            !
594
```



```
595
596
597     .TOC    *     SHIFT CONTROL
598
599
600     .TOC    *       <BMUX>-SECOND LEVEL OF SHIFT TREE
601     !BMUX CONTROLS SHIFT RIGHT OF 0 OR 4, ALWAYS IN EFFECT,
602     .FIELD BMUX::=<23>
603             DIRECT::=0              !AMUX<15:00>
604             RIGHT-4::=1             !4*D(C) # AMUX <15:04>
605
606
607
608     .TOC    *       <AMUX>-FIRST LEVEL OF SHIFT TREE
609     !AMUX CONTROLS INPUT OF D-REG/COUNTER TO TREE, ALWAYS IN EFFECT,
610     .FIELD AMUX::=<22:20>
611             DIRECT::=0              !D<HI> # D<LO>
612             D(LO)#D(LO)::=1         !D<LO> # D<LO>
613             SIGNEXT::=2             !8*D(C) # D<LO>
614             COUNTER#D(LO)::=3       !COUNTER # D<LO>
615             COUNTER::=3             !SAME
616             D(HI)#D(HI)::=4         !D<HI> # D<HI>
617             SWAB::=5                !D<LO> # D<HI>
618             RIGHT-8::=6             !8*D(C) # D<HI>
619             COUNTER#D(HI)::=7       !COUNTER # D<HI>
620
621     !-----------------------------------------------------------------------------
622
623
624
625     .TOC    *   SP REWRITE & REGISTER CLOCKS
626
627
628
629     .TOC    *     <WRCSP>-WRITE TO CSP
630     !WRITE CSP FROM DMUX (BUSDIN/CACHE), ALWAYS IN EFFECT,
631     .FIELD WRCSP::=<19>,0
632             NO::=0                  !NOP
633             YES::=1                 !ON P3, 120-150 NS,
634
635
636
637     .TOC    *     <MOD>-MODE CONTROL OF FOLLOWING BITS
638     !CONTROLS REDEFINITION OF SP REWRITE/REGISTER CLOCK BITS, ALWAYS IN EFFECT,
639     .FIELD MOD::=<14>,0
640             CLKSP::=0               !CONTROL ASP/ASP CLOCKING

641             LOADREG::=1             !CONTROL REG-REG/COUNTER LOADING
642
643
644
645     .TOC    *     SP REWRITE [A,B] CONTROL
646     !WHEN MOD/CLKSP
647
648
```

```
649     .TOC    *       <HILO>-SP HI/LO SELECT
650     !WHICH HALF OF SP'S TO REWRITE, ONLY IF MOD/CLKSP.
651     .FIELD HILO<:=<18>
652             LO<:=0                  !REWRITE ENABLE A/B SP LO [00-17]
653               L<:=0                 !
654             HI<:=1                  !REWRITE ENABLE A/B SP HI [20-37]
655               H<:=1                 !
656
657
658
659     .TOC    *       <RASEL>-REWRITE ADDRESS SELECT
660     !WHICH WRITE ADDRESS TO USE ON REWRITE, ONLY IF MOD/CLKSP.
661     .FIELD RPSEL<:=<17>
662             A-ADDR<:=0              !USE A ADDRESS ON REWRITE
663               A<:=0                 !
664             B-ADDR<:=1              !USE B ADDRESS ON REWRITE
665               B<:=1
666
667
668
669     .TOC    *       <WRSP>-REWRITE A/B SELECT
670     !ENABLE REWRITE OF SPECIFIC SP'S, ONLY IF MOD/CLKSP.
671     .FIELD WRSP<:=<16:15>
672             NOP<:=0                 !NO ASP/BSP REWRITE
673             WR-A<:=1                !WRITE ASP ONLY, ON P3 120-150 NS.
674               A<:=1                 !
675               ASP<:=1               !
676             WR-B<:=2                !WRITE BSP ONLY, ON P3 120-150 NS.
677               B<:=2                 !
678               BSP<:=2               !
679             WR-A-AND-B<:=3          !WRITE BOTH ON P3
680               AB<:=3                !
681               RA<:=3                !
682               ABSP<:=3              !
683               RASP<:=3              !
684               BOTH<:=3              !
685
686
687
688     .TOC    *       REGISTER LOADING
689     !WHEN MOD/LOADREG
690
691
692     .TOC    *       <LOADRES>-LOAD RESIDUAL CONTROL REGISTER
693     !ENABLE LOAD OF RESIDUAL CONTROL REGISTER FROM B-BUS, ONLY IF MOD/LOADREG.
694     .FIELD LOADRES<:=<18>
695             NO<:=0                  !NOP
696             YES<:=1                 !LOAD RES WITH B-BUS<14:11>
697                                     !AT P2 T[100 NS], B-BUS<14> COMPLEMENTED
698
699
700
701     .TOC    *       <LOADCOUNT>-LOAD COUNTER
702     !ENABLE LOAD OF COUNTER FROM B-BUS <7:0>, ONLY IF MOD/LOADREG.
```



```
703     .FIELD LOADCOUNT<:=<16>
704             NO<:=0                  !NOP
705             YES<:=1                 !LOAD COUNTER AT P2 T[100 NS].
706
707     !-----------------------------------------------------------------------------
708
709
710
711     .TOC    *   SEQUENCING FIELD
712
713
714
715     .TOC    *       <UPF>-BUT MICROBRANCH FIELD
716     !SPECIFIES CONDITIONS TO MODIFY <UPF>/<J> FIELD DURING BRANCH.  ALWAYS IN EFFECT.
717     .FIELD UPF<:=<13:9>,30
718
719     .TOC    *       NO BUT
720             NULL<:=30                       !SPECIFY NO MODIFICATION = DEFAULT
721
722     .TOC    *       ACTIVE ONLY
723     !PURELY ACTIVE BUTS GENERATE SIDE AFFECTS; THEY DO NOT MODIFY THE <UPF> FIELD
724     !BY THE "OR-ING"-IN-OF-CONDITIONS METHOD.  THEY MAY MODIFY EXPLICITLY THE ENTIRE <UPF> FIELD,
725     !AS IN BUT(RETURN)
726             P-OP-F<:=22                     !FORM R(SF)-ION-*001
727             CUA-TRACK<:=31                  !RESUME/RESTART CUA TRACKING
728             CLR-FLAG-RES-UCON<:=32          !CLEAR FLAGS<2:0>, EX-FLAG<1>, RES-REGISTER, UCON-REGISTER
729             DIAGNOSE<:=33                   !SPECIAL DIAGNOSTIC BUT
730             SUBRB<:=34                      !RETURN <- EMIT<14:03>, PAGE <- EMIT<02:00>
731               SUBR-B<:=34                   !SYNONYMS ARE:
732               GOTO<:=34                     !
733               GO-TO<:=34                    !
734             SUBRA<:=35                      !RETURN <- D<14:03>, PAGE <- EMIT<02:00>
735               SUBR-A<:=35                   !SYNONYM
736             B#36<:=36                       !TBD
737             RETURN<:=37                     !PAGE <- RETURN<11:09>, MUA <- RETURN<08:00>
738
739     .TOC    *       INACTIVE ONLY
740     !INACTIVE BUTS ONLY CAUSE MODIFICATION OF THE <UPF> FIELD BY THE "OR-ING"-
741     !IN-OF-CONDITIONS METHOD.
742                                             !<---UPF MASK-----
743                                             !876 543 210 OCTAL  **NOT AFFECTED
744             SR3-0<:=00                      !*** **0 000 (000)
745               CASE<:=00                     !
746               SR01<:=00                     !*** **0 111 (007)
747               SR02<:=00                     !*** **1 011 (013)
748               SP01<:=00                     !*** **1 101 (015)
749               SP00<:=00                     !*** **1 110 (016)
750             IR15-12<:=01                    !*** **0 000 (000)
751               DOP<:=01                      !
752             INSTR5<:=02                     !*** *00 000 (000)
753               IN5TH-5<:=02                  !
754             IR11#WILTPT3-0<:=03             !*** *00 000 (000)
755               IN11-A<:=03                   !*** *01 111 (017)
756             IR9-6<:=04                      !*** *** 000 (000)
```

```
        BOP::=04                        ;
        MOV-DR7!IR5-3::=05              ;*** **0 000 (000)
          MOV-DR7::=05                  ;*** **0 111 (007)
          IR5-3::=05                    ;*** **1 000 (010)
        BGSERV-FPSERV!D[C]!FPRET::=07   ;*** **0 000 (000)
          BGSERV-FPSERV::=07            ;*** **0 111 (007)
          D[C]-C::=07                   ;*** **1 011 (013)
          FPRET1-0::=07                 ;*** **1 100 (014)
        COUT07!DOUT07!FP805::=10        ;*** *** 000 (000)
          COUT07::=10                   ;*** *** 011 (003)
          DOUT07::=10                   ;*** *** 101 (005)
          COUT07!DOUT07::=10            ;*** *** 001 (001)
          FP805::=10                    ;*** *** 110 (006)
        DMO!SMO!BYTE::=11               ;*** *** 000 (000)
          DMO::=11                      ;*** *** 011 (003)
          SMO::=11                      ;*** *** 101 (005)
          BYTE::=11                     ;*** *** 110 (006)
        GD3-2::=12                      ;*** *** *00 (000)
        BG-SERVCE-L!MFSS!MULTIPLE::=14  ;*** *** 000 (000)
          BG-SERVCE-L::=14              ;*** *** 011 (003)
          MFSS::=14                     ;*** *** 101 (005)
          MULTIPLE::=14                 ;*** *** 110 (006)
            MASKED-PS(T)::=14           ;
            DF01::=14                   ;
            PS(W)::=14                  ;
            FLAG7::=14                  ;
            EXFLAG1::=14                ;
            FLTPTS::=14                 ;
            EXFLAG2::=14                ;
            INIT-JAM::=14               ;
        D14-00EQ0!D15::=15              ;*** *** *00 (000)
          D14-00-EQ-0!D15::=15          ;
          D14-00-EQ-0::=15              ;*** *** *01 (001)
          D15::=15                      ;*** *** *10 (002)
        IR11!PS15::=16                  ;*** *** *00 (000)
          IR11-B::=16                   ;*** *** *01 (001)
          PS15::=16                     ;*** *** *10 (002)
        VECTOR-LOAD!DR6-7L::=21         ;*** *** *00 (000)
          VECTOR-LOAD::=21              ;*** *** *01 (001)
          DR6-7L::=21                   ;*** *** *10 (002)
        D[C]!BA00::=23                  ;*** *** *00 (000)
          D(C)-B::=23                   ;*** *** *01 (001)
          BA00::=23                     ;*** *** *10 (002)
        OTHER-JAM!FP-PROC::=24          ;*** *** *00 (000)
          OTHER-JAM::=24                ;*** *** *01 (001)
          FP-PROC::=24                  ;*** *** *10 (002)
        INTR-HIGH!INSTR-BRANCH-L::=26   ;*** *** *00 (000)
          INTR-HIGH::=26                ;*** *** *01 (001)
          INSTR-BRANCH-L::=26           ;*** *** *10 (002)
        PREFETCH-JAM!FP-FD::=27         ;*** *** *00 (000)
          PREFETCH-JAM::=27             ;*** *** *01 (001)
          FP-FD::=27                    ;*** *** *10 (002)

.TOC    *       BOTH ACTIVE AND INACTIVE
```



```
;THESE BUTS HAVE BOTH ACTIVE AND INACTIVE EFFECTS
                                        ;----UPF MASK-----
                                        ;876 543 210 OCTAL  **NOT AFFECTED
        INSTR::=06                      ;*00 000 000 (000)  BUS CONTROL, SP REWRITE DEFEAT
          INSTR-1::=06                  ;
        SP1-0!COUNT-IS-377::=13         ;*** *** 000 (000)  BUMP COUNTER
          SP1-0::=13                    ;*** *** 001 (001)  STILL BUMP COUNTER
          COUNT-IS-377-A::=13           ;*** *** 110 (006)  BUMP COUNTER
        COUNT-IS-377!D[C]::=17          ;*** *** *00 (000)  BUMP COUNTER
          COUNT-IS-377-B::=17           ;*** *** *01 (001)  BUMP COUNTER
          D[C]-A::=17                   ;*** *** *10 (002)  STILL BUMP COUNTER
        COUNT-IS-377::=25               ;*** *** **0 (000)  BUMP COUNTER
        PREFETCH-L!SERVICE::=20         ;*** *** *00 (000)  TIMING
          PREFETCH-L::=20               ;*** *** *01 (001)  TIMING
          SERVICE::=20                  ;*** *** *10 (002)  TIMING
          LAST::=20                     ;*** *** *11 (003)  TIMING




.TOC    *       <UPF>-MICRO POINTER FIELD
;SPECIFIES EITHER NEXT MICROINSTRUCTION ADDRESS OR BASE TARGET
;ADDRESS TO BE USED "UNDER" THE BUT-CODE IN <UBF>.
.FIELD          UPF::=<8:0>,000         ;ACTUAL MICROWORD POINTER FIELD
.ADDRESS        J::=<8:0>               ;THIS FIELD ALSO HAS MICROADDRESS QUALITIES

        ;BASE MACHINE MICROCODE ENTRY POINTS:

        ;THESE ENTRY POINTS HAVE BEEN FIXED AS OF 31-AUGUST-1976,
        INIT01  ::=  3412       ;INITIALIZATION SUBROUTINE
        CON99   ::=  1040       ;FORCE "CONSOLE-MODE HALT"
        FET01   ::=  0702       ;INSTR FETCH, NO OVERLAP
        FET03   ::=  0700       ;INSTR FETCH, OVERLAP
        SER01   ::=  0701       ;SERVICE ENTRY, OVERLAP
        SER02   ::=  0703       ;SERVICE ENTRY, NO OVERLAP

        ;ENTRY POINTS INTO BASE MACHINE FOR "BUTA(DIAGNOSE)":
        ;THESE ENTRY POINTS FIXED AS OF 26-OCT-76:
        MED23   ::=  3200       ;FOR FLPADR:  D_ASPLO(DF)-TOP
        MED25   ::=  3020       ;FOR FLPADR:  D_ASPHI(DF)-TOP
        MED27   ::=  3210       ;FOR FLPADR:  ASPLO(DF)-TOP_D
        MED29   ::=  3214       ;FOR FLPADR:  ASPHI(DF)-TOP_D
        MED31   ::=  3044       ;FOR FLPADR:  D_BSPLO(DF)-TOP
        MED33   ::=  3230       ;FOR FLPADR:  D_BSPHI(DF)-TOP
        MED35   ::=  3234       ;FOR FLPADR:  BSPLO(DF)-TOP_D
        MED37   ::=  3064       ;FOR FLPADR:  BSPHI(DF)-TOP_D

        BYTE01  ::=  0032       ;FOR KJENAB:  DATOB*KJENAB

        RTS02   ::=  4034       ;FOR SPRYKT:  DATI, BA_SP-A, SP_SP+2

        DAT01   ::=  0511       ;FOR ALTER:   DATIB(P)
        DST02   ::=  0512       ;
        DST20   ::=  0527       ;
```

```
865             DST22   =:=  0525       !
866
867             LOADNZW4 =:= 4330       !FOR FPASCC&CLKFPSCC)  D_CSP(MD)
868             LOADNZW5 =:= 4332       !                FPS<3:0>_D<3:0>
869
870     !-----------------------------------------------------------------------------
871
872
873
874     .TOC    *   MISCELLANEOUS FIELDS
875
876
877
878     .TOC    *       <NEXT-PAGE>-NEW PAGE ADDRESS LOADED DURING BUT[SUBROUTINE]
879     !THESE 3 BITS ARE CLOCKED INTO PAGE REGISTER DURING A BUT[SUBRA] OR
880     !BUT[SUBRB],  ONLY USED WHEN URF/BUT[SUBRA] OR URF/BUT[SUBRB],
881     .FIELD NEXT-PAGE=:=<12:30>
882
883
884     .TOC    *       <MULTIPLE>-SELECT CODE FOR BUT[MULTIPLE]
885     !MUST BE SET IN BOTH PREVIOUS AND CURRENT MICROWORDS WHEN BUT[MULTIPLE] IS TO BE EMPLOYED,
886     .FIELD MULTIPLE=:=<32:30>
887             MASKED=PS[I]=:=0        !
888             D#0=:=1                 !
889             PS[N]=:=2               !
890             FLAG7=:=3               !
891             EXFLAG1=:=4             !
892             FLTPT#=:=5              !
893             EXFLAG2=:=6             !
894             INIT=JAM=:=7            !
895
896
897
898     .TOC    *       EMIT FIELD = IMMEDIATE DATA FROM MICROWORD
899     !USED WHENEVER LOADING IMMEDIATE DATA FROM MICROWORD
900     .FIELD EMIT=:=<47:44>*<41:30>
901     .FIELD EMITH=:=<47:44>
902     .FIELD EMITM=:=<41:38>
903     .FIELD EMITL=:=<37:30>
904     .FIELD EMITML=:=<41:30>
905     .FIELD EMIT9-8=:=<39:36>
906     .FIELD EMIT15=:=<47>
907     .FIELD EMIT14=:=<46>
908     .FIELD EMIT13=:=<45>
909     .FIELD EMIT12=:=<44>
910     .FIELD EMIT11=:=<43>
911     .FIELD EMIT10=:=<42>
912     .FIELD EMIT09=:=<39>
913     .FIELD EMIT08=:=<38>
914     .FIELD EMIT07=:=<37>
915     .FIELD EMIT06=:=<36>
916     .FIELD EMIT05=:=<35>
917     .FIELD EMIT04=:=<34>
918     .FIELD EMIT03=:=<33>
```



```
919     .FIELD EMIT02=:=<32>
920     .FIELD EMIT01=:=<31>
921     .FIELD EMIT00=:=<30>
922
923
924
925     .TOC    *       RETURN ADDRESS = FOR MICROSUBROUTINE CALLS
926     !USED WITH BUT(SUBRB) AND BUT(SUBRA)
927     .FIELD RETURN=:=<46:44>*<41:33>          !PAGE # D.I.P.
928
929
930
931     .TOC    *       UCON SELECTION AND CONTROL FIELDS
932
933
934     .TOC    *           SELECTION
935     !SELECT PARTICULAR UCON, ONLY USED IF BEGIN/YES & SELECT/UCON,
936     !FIELD UCON-SEL-EMIT=:=<43>              !SELECT EMIT CAN ONLY BE DONE BY USING
937     !       NO=:=0                           !  BUTA(CLR-FLAG-RES-UCON) TO ASSERT UCON-SEL-EMIT-L
938     !       YES=:=1                          !
939     .FIELD UCON-SEL-I-O=:=<46>               !SELECT I-O [BUS] CONTROL
940             NO=:=0
941             YES=:=1
942     .FIELD UCON-SEL-MCS=:=<45>               !SELECT MCS/ECS/DCS
943             NO=:=0
944             YES=:=1
945     .FIELD UCON-SEL-CACHEKT=:=<44>           !SELECT CACHE/KT
946             NO=:=0
947             YES=:=1
948     .FIELD UCON-SEL-PROC=:=<36>              !SELECT PROCESSOR CONTROL
949             NO=:=0
950             YES=:=1
951     .FIELD UCON-SEL-FLTPT=:=<33>             !SELECT NOT FLOATING POINT
952             NO=:=0
953             YES=:=1
954
955
956
957     .TOC    *           CONTROL
958     !AFTER UCON(S) SELECTED FROM ABOVE, CONTROL COMES FROM HERE,
959     .FIELD UCON=:=<32:30>*<35:34>*<47>*<42:38>
960     .FIELD UCONH=:=<32:30>
961     .FIELD UCONM=:=<35:34>
962     .FIELD UCONL=:=<47>*<42:38>
963     .FIELD UCON15=:=<32>
964     .FIELD UCON14=:=<31>

965     .FIELD UCON13=:=<30>
966     .FIELD UCON12=:=<35>
967     .FIELD UCON11=:=<34>
968     .FIELD UCON10=:=<47>
969     .FIELD UCON09=:=<42>
970     .FIELD UCON08=:=<41>
971     .FIELD UCON07=:=<40>
972     .FIELD UCON06=:=<39>
```

```
.FIELD UCONOS/=<38>

;--------------------------------------------------------------------------------------------------



.TOC	*	BASE MACHINE EXTENSION BITS
;
;	LAYOUT IN BASE MACHINE (NOT DCS) ADDRESS SPACE:
;
;				-------------- PAGE --------------
;	--- NAME ---		0	1	2	3	4
;	ROMEX 00 H		X				X
;	ROMEX 01 H				X	X
;	ROMEX 03 H					X
;
;	FPSEL L			X		X		X
;	SFCC L			X		X		X
;	FLPADR L		X		X	X	X
;
;	SPBYKT L		X				X
;	UDAD01 L		X				X
;	UDAD00 L		X				X
;	UKJCONT L		X				X
;	UKT01 H			X				X
;	UKT00 H			X				X
;	UALTER L		X				X
;	UKTEN L			X				X
;
;
;	NULL BIT DEFINITIONS:
;
.FIELD ROMEX00/=<60>			;ACTIVE HIGH
	ZERO/=0				;
	ONE/=1				;
.FIELD ROMEX01/=<61>			;ACTIVE HIGH
	ZERO/=0				;
	ONE/=1				;
.FIELD ROMEX03/=<62>			;ACTIVE HIGH
	ZERO/=0				;
	ONE/=1				;
.FIELD FPSEL/=<57>			;ACTIVE LOW
	ZERO/=1				;
	ONE/=0				;
.FIELD SFCC/=<58>			;ACTIVE LOW
	ZERO/=1				;
	ONE/=0				;
.FIELD FLPADR/=<59>			;ACTIVE LOW
	ZERO/=1				;
	ONE/=0				;
.FIELD SPBYKT/=<55>			;ACTIVE LOW
	ZERO/=1				;
	ONE/=0				;
.FIELD UDAD01/=<53>			;ACTIVE LOW
```



```
	ZERO/=1				;
	ONE/=0				;
.FIELD UDAD00/=<52>			;ACTIVE LOW
	ZERO/=1				;
	ONE/=0				;
.FIELD UKJCONT/=<54>			;ACTIVE LOW
	ZERO/=1				;
	ONE/=0				;
.FIELD UKT01/=<49>			;ACTIVE HIGH
	ZERO/=0				;
	ONE/=1				;
.FIELD UKT00/=<48>			;ACTIVE HIGH
	ZERO/=0				;
	ONE/=1				;
.FIELD UALTER/=<51>			;ACTIVE LOW
	ZERO/=1				;
	ONE/=0				;
.FIELD UKTEN/=<50>			;ACTIVE LOW
	ZERO/=1				;
	ONE/=0				;

;--------------------------------------------------------------------------------------------------



.TOC	*	SPECIAL DCS FIELDS



.TOC	*	FIELDS USED IN PAGES 4, 5, OR 6 OF DCS


.TOC	*	<LOAD-DCS-CTR>-LOAD DIAGNOSTIC COUNTER FROM EMITH
;THIS CODE LOADS THE 4-BIT DCS COUNTER FROM THE CURRENT
;MICROWORD'S EMITH FIELD.  THIS COUNTER IS CLOCKED AT EVERY PO
;FOLLOWING, UNTIL THE COUNTER REACHES ZERO.  AT THIS POINT, THE
;COMPARE IS ENABLED, CLOCKING THE RESULT OF THE CURRENT ENUA:TNUA
;COMPARE INTO THE ERROR LATCH.
;ONLY USED IN PAGES 4, 5, OR 6 OF DCS.
.FIELD LOAD-DCS-CTR/=<51>,0
	NOP/=0				;NOP
	YES/=1				;LOAD, COUNT, AND ENABLE COMPARE



.TOC	*	<CTR>-4 BIT DCS COUNTER VALUE FROM EMIT

;THIS FOUR BIT VALUE IS LOADED INTO THE COUNTER (DIAGNOSTICS).
;WHEN LOAD COUNTDOWN/YES,  COMPLEMENT OF ACTUAL VALUE IS USED, FOR COUNT DOWN.
;LOADING COUNTER VALUE OF 17(8) CAUSES COMPARE AT END OF THIS UWORD.
.FIELD CTR/=<47:44>
	C0,/=17
	C1,/=16
	C2,/=15
	C3,/=14
```

```
1081            C4,1:=13
1082            C5,1:=12
1083            C6,1:=11
1084            C7,1:=10
1085            C8,1:=07
1086            C9,1:=06
1087            C10,1:=05
1088            C11,1:=04
1089            C12,1:=03
1090            C13,1:=02
1091            C14,1:=01
1092            C15,1:=00
1093
1094
1095
1096    .TOC    *       <LOAD-ENUA-ERRCOD>-LOAD THE ENUA AND ERRCOD REGISTERS
1097    !THIS CODE LOADS THE 12-BIT ENUA REGISTER FROM THE <EMITH,EMITL> FIELD
1098    !OF THE CURRENT MICROWORD, AND LATCHES THE NUA INTO THE ERRCOD REGISTER.
1099    !ONLY USED IN PAGES 4, 5, OR 6 OF DCS.
1100    .FIELD LOAD-ENUA-ERRCOD:,=<54>,0
1101            NOP:,=0                 !NOP
1102            YES:,=1                 !LOAD REGISTERS AT P0
1103
1104
1105
1106    .TOC    *       <ENUA>-ENUA VALUE FROM EMIT
1107    !THIS 12 BIT FIELD IS LOADED FROM <EMIT> TO THE ENUA REGISTER
1108    !WHEN LOAD ENUA-ERRCOD/YES.
1109    .FIELD ENUA:,=<41:30>
1110
1111
1112
1113    .TOC    *       <VERIFY>-VERIFY BIT FOR SELF CHECK TEST
1114    !WHEN IN SELF TEST MODE OF DCS, SETTING THIS BIT CAUSES THE VERIFY COUNTER TO BE
1115    !BUMPED AT THE START OF THIS MICROWORD.  THE VERIFY BIT IS IMPLICITLY SET FOR
1116    !ANY REFERENCE TO PAGE 7 (IE. THE COUNTER IS AUTOMATICALLY BUMPED ON A REFERENCE
1117    !TO PAGE 7).
1118    !ONLY (EXPLICITLY) USED IN PAGES 4, 5, OR 6 OF DCS, WHEN IN SELF TEST MODE.
1119    .FIELD VERIFY:,=<48>,0
1120            NOP:,=0                 !NO ACTION
1121            BUMP:,=1                !BUMP VERIFY COUNTER AT P0, WHEN IN SELF TEST MODE
1122
1123
1124
1125    .TOC    *       FIELDS USED IN PAGE 7 OF DCS EXTENSION
1126
1127
1128    .TOC    *       <EOP>-SIGNAL SUCCESSFUL END OF PASS
1129    !THIS CODE SETS THE END OF PASS LATCH, LIGHTING THE EOP LED
1130    !ONLY USED IN PAGE 7 OF DCS.
1131    .FIELD EOP:,=<49>,0
1132            NO:,=0                  !NO EOP
1133            SIGNAL:,=1              !SIGNAL SUCCESSFUL EOP AT P0
1134
```



```
1135
1136
1137    .TOC    *       <DAD>-DCS CONTROL OF BASE MACHINE EXTENSION DAD BITS
1138    !THESE BITS ARE WIRE-ANDED INTO THE BASE MACHINE DAD<11:0> BITS.
1139    !MUST BE SPECIFIED IN UWORD BEFORE C&P REFERENCE.
1140    !ONLY USED IN PAGE 7 OF DCS.
1141    .FIELD DAD:,=<53:52>,0
1142            NO-DAD:,=0              !
1143            FIRST-1-OR-2:,=1        !SETUP BYTE-CONST NON INPUT
1144            SECOND-1-OR-2:,=2       !  "
1145            WRITE-BYTE:,=3          !SETUP FOR BYTE WRITE TO ASP/SSP
1146
1147
1148
1149    .TOC    *       FIELDS USED IN ALL PAGES OF DCS EXTENSION
1150
1151
1152    .TOC    *       <SCOPE>-SCOPE ON ERROR, DIAGNOSTIC BIT
1153    !THIS CODE IS A SPECIAL BIT; THAT, WHEN ENABLED, CHECKS THE ERROR
1154    !LATCH TO SEE IF IT IS SET.  IF IT IS, NUADO IS FORCED TO A ZERO;
1155    !ELSE IT IS LEFT UNCHANGED.  USED TO IMPLEMENT FORCED SCOPE LOOP ON ERROR.
1156    !USED IN ALL PAGES OF DCS.
1157    .FIELD SCOPE:,=<50>,0
1158            NOP:,=0                 !NOP
1159            ENABLED:,=1             !ENABLE SCOPE LOOPING FACILITY
1160
1161
1162    !-------------------------------------------------------------------------------
1163    !END OF MICROWORD FIELD DEFINITIONS
1164    !-------------------------------------------------------------------------------
1165
1166
1167
1168    !.PAGE=========================================================================
1169
1170    .TOC    * MACRO DEFINITIONS
1171
1172
1173    .TOC    *   PRIMITIVE OPERATIONS
1174
1175
1176    .TOC    *     TIMING
1177    .MACRO P0       ;:=  NULL       !0 NS,, UP3 VIEWED AS THE START OF A MICROCYCLE
1178
1179    .MACRO P1       ;:=  NULL       !60 NS,, AT P1
1180    .MACRO P1-L     ;:=  NULL       !30 NS,, AT P1 LEADING EDGE
1181    .MACRO P1-T     ;:=  NULL       !60 NS,, AT P1 TRAILING EDGE
1182
1183    .MACRO P2       ;:=  NULL       !100 NS,, AT P2
1184    .MACRO P2-L     ;:=  NULL       !70 NS,, AT P2 LEADING EDGE
1185    .MACRO P2-T     ;:=  WHEN/AT-P2-T  !100 NS,, AT P2 TRAILING EDGE
1186    .MACRO P2-U     ;:=  NULL       !UNSUPPRESSED P2, CLOCK CONTINUOUSLY
1187
1188    .MACRO P3       ;:=  NULL       !150 NS,, 120-150 NS,, AT P3
```

```
1189    .MACRO P3-L     ;;=  NULL       ;120 NS., AT P3 LEADING EDGE
1190    .MACRO P3-T     ;;=  WHEN/AT-P3-T  ;150 NS., AT P3 TRAILING EDGE
1191    .MACRO P3-U     ;;=  NULL       ;UNSUPPRESSED P3, CLOCK CONTINUOUSLY
1192
1193    .MACRO UP3      ;;=  NULL       ;P3 DELAYED BY 5 NS., PO VIEWED AS THE END OF A
1194                                    ;MICROCYCLE. LATCHES NEW MICROINSTRUCTION INTO
1195                                    ;THE MICROWORD BUFFER REGISTER,
1196
1197    .MACRO DEFER    ;;=  NULL       ;CONTROL IS ISSUED AT THIS TIME,
1198                                    ;  ANY REQUIRED CLOCKING OCCURS LATER
1199    .MACRO NEXT     ;;=  NULL       ;WHERE TO GO NEXT, CLOCKED AT UP3
1200    .MACRO SETUP    ;;=  NULL       ;SETUP DATA/CONTROL
1201    .MACRO SELECT   ;;=  NULL       ;MAKE A HOT-BOX SELECTION
1202    .MACRO ISSUE    ;;=  NULL       ;SET/CLEAR HOT-BOX FLAG
1203    .MACRO ENABLE   ;;=  NULL       ;  DITTO
1204    .MACRO EMITC    ;;=  NULL       ;SPECIFY AN EMIT-CONSTANT VALUE
1205
1206
1207
1208    ;.PAGE**********************************************************************************************************
1209
1210    .TOC    *       WRITING THE A AND B SCRATCH PADS
1211    !
1212    !  WRITING THE APPROPRIATE SCRATCH PADS:
1213    !
1214    !                                        (NOP       )
1215    !                                        (A    L    A)
1216    !                                     WR (B  , H  , B)
1217    !                                        (AB        )
1218    !                                       /!\ /!\  /!\
1219    !                                       |   |    |
1220    !       ASP, BSP, BOTH, NEITHER----------   |    |
1221    !       LO(00-17), OR HI(20-37)--------------    |
1222    !       USE "A" SIDE OR "B" SIDE ADDRESS----------
1223    !
1224    !  WRITES CONTENTS OF D-REGISTER INTO ADDRESSED SCRATCH PADS (SEE
1225    !       BELOW) DURING P3
1226    !
1227    .MACRO WP(AB,HL,ADDR)  ;;=  MOD/CLKSP,          ;CLOCK SP MODE
1228                                WRSP/PAB,           ;NOP, A, ASP, B, BSP, AB, ABSP, BA, BASP, BOTH ARE CHOICES
1229                                HILO/SHL,           ;HI, LO, H, L ARE CHOICES
1230                                WRSEL/SADDR         ;A, B, A=ADDR, B=ADDR ARE CHOICES
1231
1232
1233
1234    .TOC    *       ASP AND BSP PHYSICAL REGISTER ADDRESSES
1235    !
1236    !  ENABLE INPUT/OUTPUT (FOR READ AND/OR WRITE) OF THE APPROPRIATE SCRATCH PAD ONTO
1237    !  EITHER BUS-A OR BUS-B VIA EXACT PHYSICAL ADDRESS
1238    !
1239    .MACRO ASPLO(XX)        ;;=  AEN/ASPLO,         ;SELECT
1240                                 ASEL/0XX,          ;REGISTER &
1241                                 RIF/0XX            ;ENABLE ON BUS-A
1242
```



```
1243    .MACRO ASPHI(XX)        ;;=  AEN/ASPHI,         ;SELECT
1244                                 ASEL/0XX,          ;REGISTER &
1245                                 RIF/0XX            ;ENABLE ON BUS-A
1246
1247    .MACRO ASP(XX)          ;;=  ASEL/0XX,          ;SELECT REGISTER,
1248                                 RIF/0XX            ;NO ENABLE
1249
1250
1251    .MACRO BSPLO(XX)        ;;=  BEN/BSPLO,         ;SELECT
1252                                 BSEL/0XX,          ;REGISTER &
1253                                 RIF/0XX            ;ENABLE ON BUS-B
1254
1255    .MACRO BSPHI(XX)        ;;=  BEN/BSPHI,         ;SELECT
1256                                 BSEL/0XX,          ;REGISTER &
1257                                 RIF/0XX            ;ENABLE ON BUS-B
1258
1259    .MACRO BSP(XX)          ;;=  BSEL/0XX,          ;SELECT REGISTER,
1260                                 RIF/0XX            ;NO ENABLE
1261
1262
1263
1264
1265
1266    .TOC    *       ASP AND BSP BASE MACHINE FUNCTIONAL REGISTER ADDRESSES
1267    !
1268    !  ENABLE INPUT/OUTPUT (FOR READ AND/OR WRITE) OF THE APPROPRIATE SCRATCH PAD ONTO
1269    !  EITHER BUS-A "-A" OR BUS-B "-B" VIA FUNCTIONAL REGISTER DESIGNATION
1270    !
1271    .MACRO R0-A             ;;=  ASPLO(R00)
1272    .MACRO R0-B             ;;=  BSPLO(R00)
1273    .MACRO R1-A             ;;=  ASPLO(R01)
1274    .MACRO R1-B             ;;=  BSPLO(R01)
1275    .MACRO R2-A             ;;=  ASPLO(R02)
1276    .MACRO R2-B             ;;=  BSPLO(R02)
1277    .MACRO R3-A             ;;=  ASPLO(R03)
1278    .MACRO R3-B             ;;=  BSPLO(R03)
1279    .MACRO R4-A             ;;=  ASPLO(R04)
1280    .MACRO R4-B             ;;=  BSPLO(R04)
1281    .MACRO R5-A             ;;=  ASPLO(R05)
1282    .MACRO R5-B             ;;=  BSPLO(R05)
1283    .MACRO SP-A             ;;=  ASPLO(R06)
1284    .MACRO SP-B             ;;=  BSPLO(R06)
1285    .MACRO PC-A             ;;=  ASPLO(R07)
1286    .MACRO PC-B             ;;=  BSPLO(R07)
1287    .MACRO FACA[0]-B        ;;=  BSPHI(R10)
1288    .MACRO FACB[0]-A        ;;=  ASPHI(R10)

1289    .MACRO FACC[0]-B        ;;=  BSPLO(R10)
1290    .MACRO FACD[0]-A        ;;=  ASPLO(R10)
1291    .MACRO FACA[1]-B        ;;=  BSPHI(R11)
1292    .MACRO FACB[1]-A        ;;=  ASPHI(R11)
1293    .MACRO FACC[1]-B        ;;=  BSPLO(R11)
1294    .MACRO FACD[1]-A        ;;=  ASPLO(R11)
1295    .MACRO FACA[2]-B        ;;=  BSPHI(R12)
1296    .MACRO FACB[2]-A        ;;=  ASPHI(R12)
```

```
1297    .MACRO FACC[2]=B        ;;=  BSPLO(R12)
1298    .MACRO FACD[2]=A        ;;=  ASPLO(R12)
1299    .MACRO FACA[3]=B        ;;=  BSPHI(R13)
1300    .MACRO FACB[3]=A        ;;=  ASPHI(R13)
1301    .MACRO FACC[3]=B        ;;=  BSPLO(R13)
1302    .MACRO FACD[3]=A        ;;=  ASPLO(R13)
1303    .MACRO FACA[4]=B        ;;=  BSPHI(R14)
1304    .MACRO FACB[4]=A        ;;=  ASPHI(R14)
1305    .MACRO FACC[4]=B        ;;=  BSPLO(R14)
1306    .MACRO FACD[4]=A        ;;=  ASPLO(R14)
1307    .MACRO FACA[5]=B        ;;=  BSPHI(R15)
1308    .MACRO FACB[5]=A        ;;=  ASPHI(R15)
1309    .MACRO FACC[5]=B        ;;=  BSPLO(R15)
1310    .MACRO FACD[5]=A        ;;=  ASPLO(R15)
1311    .MACRO FDSTA=B          ;;=  BSPHI(R17)
1312    .MACRO FDSTB=A          ;;=  ASPHI(R17)
1313    .MACRO FDSTC=B          ;;=  BSPLO(R17)
1314    .MACRO FDSTD=A          ;;=  ASPLO(R17)
1315    .MACRO FPSH1#FFC=A      ;;=  ASPHI(R16)
1316    .MACRO FEA=B            ;;=  BSPHI(R16)
1317    .MACRO USER=SP=A        ;;=  ASPLO(R16)
1318    .MACRO USER=SP=B        ;;=  BSPLO(R16)
1319    .MACRO WHAMI=A          ;;=  ASPHI(R02)
1320    .MACRO P[ZERO]=B        ;;=  BSPHI(R03)
1321    .MACRO R[IR]=A          ;;=  ASPHI(R17)
1322    .MACRO R[SRC]=B         ;;=  BSPHI(R04)
1323    .MACRO R[SRC]=A         ;;=  ASPHI(R04)
1324    .MACRO R[DST]=B         ;;=  BSPHI(R05)
1325    .MACRO R[DST]=A         ;;=  ASPHI(R05)
1326    .MACRO R[VECT]=B        ;;=  BSPHI(R02)
1327    .MACRO WCSR[0]=B        ;;=  BSPHI(R00)
1328    .MACRO WCSB[1]=B        ;;=  BSPHI(R01)
1329    .MACRO WCSA[0]=A        ;;=  ASPHI(R00)
1330    .MACRO WCSA[1]=A        ;;=  ASPHI(R01)
1331    .MACRO FPA=B            ;;=  BSPHI(R06)
1332    .MACRO CNSL=CNTL=B      ;;=  BSPHI(R07)
1333    .MACRO CNSL=CADR=A      ;;=  ASPHI(R07)
1334    .MACRO CNSL=SW=A        ;;=  ASPHI(R06)
1335    .MACRO CNSL=TMPSW=A     ;;=  ASPHI(R03)
1336
1337
1338
1339    .TOC    *       ASP AND BSP INDIRECT REGISTER ADDRESSES
1340    !
1341    !  ENABLE INPUT/OUTPUT (FOR READ AND/OR WRITE) OF THE APPROPRIATE SCRATCH PAD
1342    !  ON BUS=A (A) OR BUS=B (B) USING INDIRECT ADDRESSING WITH THE IR,
1343    !  WHERE:
1344    !
1345    !       SF<3:0>H = (FLPADR H + KTSPCADR83 H) + (FLPT L * IR8 H) + (IR7 H) + (IR6 H + ROR1 H)
1346    !
1347    !       DF<3:0>H = (FLPADR H + KTDSTADR83 H) + (IR2 H) + (IR1 H) + (IR0 H)
1348    !
1349    .MACRO R[SF]=LO=A       ;;=  AEN/ASPLD,ASEL/SF
1350    .MACRO R[SF]=LO=B       ;;=  BEN/BSPLD,BSEL/SF
```



```
1351    .MACRO R[SF]=HI=A       ;;=  AEN/ASPHI,ASEL/SF
1352    .MACRO R[SF]=HI=B       ;;=  BEN/BSPHI,BSEL/SF
1353    .MACRO R[DF]=LO=A       ;;=  AEN/ASPLO,ASEL/DF
1354    .MACRO R[DF]=LO=B       ;;=  BEN/BSPLO,BSEL/DF
1355    .MACRO R[DF]=HI=A       ;;=  AEN/ASPHI,ASEL/DF
1356    .MACRO R[DF]=HI=B       ;;=  BEN/BSPHI,BSEL/DF
1357    .MACRO R[SF]=A          ;;=  R[SF]=LO=A
1358    .MACRO R[SF]=B          ;;=  R[SF]=LO=B
1359    .MACRO R[DF]=A          ;;=  R[DF]=LO=A
1360    .MACRO R[DF]=B          ;;=  R[DF]=LO=B
1361
1362
1363
1364    .TOC    *       ASP, BSP INDIRECT ADDRESSING
1365    !
1366    !       THESE MACROS ONLY SELECT THE ADDRESS MODE FOR THE ASP AND BSP)
1367    !       THE SELECTED SP IS NOT ENABLED ONTO THE BUS
1368    !
1369    .MACRO ASP=ADDRS=R[DF]  ;;=  ASEL/DF
1370    .MACRO ASP=ADDRS=R[SF]  ;;=  ASEL/SF
1371    .MACRO BSP=ADDRS=R[DF]  ;;=  BSEL/DF
1372    .MACRO BSP=ADDRS=R[SF]  ;;=  BSEL/SF
1373
1374
1375
1376
1377    .TOC    *       ASP AND BSP DCS SPECIFIC FUNCTIONAL REGISTER ADDRESSES
1378    !
1379    !  ENABLE INPUT/OUTPUT (FOR READ AND/OR WRITE) OF THE APPROPRIATE SCRATCH PAD ONTO
1380    !  EITHER BUS=A "=A" OR BUS=B "=B" VIA FUNCTIONAL REGISTER DESIGNATION
1381    !
1382    .MACRO C000000=A        ;;=  ASPHI(C000000)     !IN R01
1383    .MACRO C000000=B        ;;=  BSPHI(C000000)     !IN R01
1384    .MACRO C177777=A        ;;=  ASPHI(C177777)     !IN R03
1385    .MACRO C177777=B        ;;=  BSPHI(C177777)     !IN R03
1386    .MACRO C125252=A        ;;=  ASPHI(C125252)     !IN R05
1387    .MACRO C125252=B        ;;=  BSPHI(C125252)     !IN R05
1388    .MACRO C052525=A        ;;=  ASPHI(C052525)     !IN R07
1389    .MACRO C052525=B        ;;=  BSPHI(C052525)     !IN R07
1390    .MACRO C000001=A        ;;=  ASPHI(C000001)     !IN R11
1391    .MACRO C000001=B        ;;=  BSPHI(C000001)     !IN R11
1392    .MACRO C100000=A        ;;=  ASPHI(C100000)     !IN R13
1393    .MACRO C100000=B        ;;=  BSPHI(C100000)     !IN R13
1394    .MACRO C000200=A        ;;=  ASPHI(C000200)     !IN R15
1395    .MACRO C000200=B        ;;=  BSPHI(C000200)     !IN R15
1396

1397
1398
1399    !.PAGE=========================================================================================================
1400
1401    .TOC    *       WRITING THE C SCRATCH PAD
1402    !
1403    !  WRITE DATA ON BUSDIN (ACTUALLY DMUX OUTPUT) INTO ADDRESSED CSP LOCATION
1404    !       (SEE BELOW) DURING P3
```

```
1405    ;
1406    .MACRO WR-CSP           ;;=  WRCSP/YES
1407
1408
1409
1410    .TOC    *     CSP IMPLIED ADDRESSING
1411    ;
1412    ;  ENABLE FOR INPUT/OUTPUT (READ AND/OR WRITE) ONTO BUS-B ONLY A SPECIFIC CSP LOCATION,
1413    ;  WHERE THE ADDRESS IS DETERMINED AS FOLLOWS:
1414    ;
1415    ;       CSPADDR<3:0>H = <1 0 0 0 BSEL<1>H BSEL<0>H >
1416    ;
1417    .MACRO CSPB(XX)         ;;=  SEN/BASCON,          ;USE IMMEDIATE MODE
1418                                 BSEL/@XX             ;WHICH ONE
1419
1420
1421
1422    .TOC    *     CSP DIRECT ADDRESSING
1423    ;
1424    ;  ENABLE FOR INPUT/OUTPUT (READ AND/OR WRITE) ONTO BUS-B ONLY A SPECIFIC CSP LOCATION,
1425    ;  WHERE THE ADDRESS IS DETERMINED AS FOLLOWS:
1426    ;
1427    ;       CSPADDR<3:0>H = .UNIBUS<21:20> H
1428    ;
1429    .MACRO CSPD(XX)         ;;=  SEN/CSP,             ;USE CSP-ADDR MODE
1430                                 CSPADDR/@XX          ;WHICH ONE
1431
1432
1433
1434    ;.PAGE*****************************************************************************************************
1435
1436    .TOC    *     SHIFT TREE SPECIFICATION
1437    ;N.B. MAY REQUIRE PRIOR SETUP OF REG-REGISTER FOR SHIFT AND MUX SELECTION CONTROL
1438    ;  (EG, WHEN ASEL/LEFT=1 IS USED).
1439
1440    .TOC    *       ENABLED ONTO BUS A
1441    .MACRO D=RIGHT=14       ;;=  AEN/CMUX,AMUX/RIGHT=8,BMUX/RIGHT=4,ASEL/RIGHT=2
1442    .MACRO D=RIGHT=13       ;;=  AEN/CMUX,AMUX/RIGHT=8,BMUX/RIGHT=4,ASEL/RIGHT=1
1443    .MACRO D=RIGHT=12       ;;=  AEN/CMUX,AMUX/RIGHT=8,BMUX/RIGHT=4,ASEL/DIRECT
1444    .MACRO D=RIGHT=11       ;;=  AEN/CMUX,AMUX/RIGHT=8,BMUX/RIGHT=4,ASEL/LEFT=1      ;SENDMUX SETUP
1445    .MACRO D=RIGHT=10       ;;=  AEN/CMUX,AMUX/RIGHT=8,BMUX/DIRECT,ASEL/RIGHT=2
1446    .MACRO D=RIGHT=9        ;;=  AEN/CMUX,AMUX/RIGHT=8,BMUX/DIRECT,ASEL/RIGHT=1
1447    .MACRO D=RIGHT=8        ;;=  AEN/CMUX,AMUX/RIGHT=8,BMUX/DIRECT,ASEL/DIRECT
1448    .MACRO D=RIGHT=7        ;;=  AEN/CMUX,AMUX/RIGHT=8,BMUX/DIRECT,ASEL/LEFT=1      ;SENDMUX SETUP
1449    .MACRO D=RIGHT=6        ;;=  AEN/CMUX,AMUX/DIRECT,BMUX/RIGHT=4,ASEL/RIGHT=2
1450    .MACRO D=RIGHT=5        ;;=  AEN/CMUX,AMUX/DIRECT,BMUX/RIGHT=4,ASEL/RIGHT=1
1451    .MACRO D=RIGHT=4        ;;=  AEN/CMUX,AMUX/DIRECT,BMUX/RIGHT=4,ASEL/DIRECT
1452    .MACRO D=RIGHT=3        ;;=  AEN/CMUX,AMUX/DIRECT,BMUX/RIGHT=4,ASEL/LEFT=1      ;SENDMUX SETUP
1453    .MACRO D=RIGHT=2        ;;=  AEN/CMUX,AMUX/DIRECT,BMUX/DIRECT,ASEL/RIGHT=2
1454    .MACRO D=RIGHT=1        ;;=  AEN/CMUX,AMUX/DIRECT,BMUX/DIRECT,ASEL/RIGHT=1
1455    .MACRO D=NO=SHIFT       ;;=  AEN/CMUX,AMUX/DIRECT,BMUX/DIRECT,ASEL/DIRECT
1456    .MACRO D=DIRECT         ;;=  D=NO=SHIFT
1457    .MACRO D=LEFT=1         ;;=  AEN/CMUX,AMUX/DIRECT,BMUX/DIRECT,ASEL/LEFT=1       ;SENDMUX SETUP
1458    .MACRO D=SWAB           ;;=  AEN/CMUX,AMUX/SWAB,BMUX/DIRECT,ASEL/DIRECT
```



```
1459    .MACRO D=SWAB=RIGHT=3   ;;=  AEN/CMUX,AMUX/SWAB,BMUX/RIGHT=4,ASEL/LEFT=1       ;SENDMUX SETUP
1460    .MACRO D=SWAB=LEFT=1    ;;=  AEN/CMUX,AMUX/SWAB,BMUX/DIRECT,ASEL/LEFT=1       ;SENDMUX SETUP
1461    .MACRO D=SIGNEXT        ;;=  AEN/CMUX,AMUX/SIGNEXT,BMUX/DIRECT,ASEL/DIRECT
1462    .MACRO D=SIGNEXT=RIGHT=1;;=  AEN/CMUX,AMUX/SIGNEXT,BMUX/DIRECT,ASEL/RIGHT=1
1463    .MACRO D=SIGNEXT=LEFT=1 ;;=  AEN/CMUX,AMUX/SIGNEXT,BMUX/DIRECT,ASEL/LEFT=1     ;SENDMUX SETUP
1464    .MACRO NO=SHIFT         ;;=  AEN/CMUX,BMUX/DIRECT,ASEL/DIRECT
1465    .MACRO DIRECT           ;;=  NO=SHIFT
1466    .MACRO COUNTRD[HI]      ;;=  AEN/CMUX,AMUX/COUNTER#D[HI],BMUX/DIRECT,ASEL/DIRECT
1467    .MACRO COUNT#D[LO]      ;;=  AEN/CMUX,AMUX/COUNTER#D[LO],BMUX/DIRECT,ASEL/DIRECT
1468
1469
1470
1471    .TOC    *       FIRST TWO LEVELS ONLY (AMUX,BMUX)
1472    ;N.B.:  FOR USE WHEN SHIFTING AR RIGHT, AR<15> <= BMUX<00>
1473    .MACRO D=DIRECT[AMUX]   ;;=  AMUX/DIRECT,BMUX/DIRECT
1474
1475
1476
1477    ;.PAGE*****************************************************************************************************
1478
1479    .TOC    *     ALU FUNCTIONS
1480    ;[SEE FIELD DESCRIPTION OF "ALU" FOR FULL DESCRIPTION]
1481    .MACRO ZERO             ;;=  ALU/ZERO
1482    .MACRO A=XOR=B          ;;=  ALU/A=XOR=B
1483    .MACRO B                ;;=  ALU/B
1484    .MACRO A=AND=B          ;;=  ALU/A=AND=B
1485    .MACRO A=IOR=B          ;;=  ALU/A=IOR=B
1486    .MACRO A                ;;=  ALU/A
1487    .MACRO NOT=A            ;;=  ALU/NOT=A
1488    .MACRO NOT=A=AND=B      ;;=  ALU/NOT=A=AND=B
1489    .MACRO A=AND=NOT=B      ;;=  ALU/A=AND=NOT=B
1490
1491    .MACRO DIVIDE           ;;=  ALU/DIVIDE
1492    .MACRO A=PLUS=B         ;;=  ALU/A=PLUS=B
1493    .MACRO A=MINUS=B        ;;=  ALU/A=MINUS=B
1494    .MACRO A=PLUS=B=PLUS=PS[C]      ;;=  ALU/A=PLUS=B=PLUS=PS[C]
1495    .MACRO A=PLUS=B=PLUS=D[C]       ;;=  ALU/A=PLUS=B=PLUS=D[C]
1496    .MACRO A=PLUS=NOT=B=PLUS=D[C]   ;;=  ALU/A=PLUS=NOT=B=PLUS=D[C]
1497    .MACRO A=PLUS=B=PLUS=1          ;;=  ALU/A=PLUS=B=PLUS=1
1498
1499
1500    .TOC    *     COUT GENERATION
1501    ;[SEE FIELD DESCRIPTION OF "COUT" FOR FULL DESCRIPTION]
1502    .MACRO COUT_CIN         ;;=  COUT/CIN
1503    .MACRO COUT_PS[C]       ;;=  COUT/PS[C]
1504    .MACRO COUT_ALU00       ;;=  COUT/ALU00

1505    .MACRO COUT_ALU07       ;;=  COUT/ALU07
1506    .MACRO COUT_ALU15       ;;=  COUT/ALU15
1507    .MACRO COUT_COUT07      ;;=  COUT/COUT07
1508    .MACRO COUT_COUT15      ;;=  COUT/COUT15
1509    .MACRO COUT_D[C]        ;;=  COUT/D[C]
1510
1511
1512
```

```
1513    ;.PAGE********************************************************************************************************************
1514
1515    .TOC    *       CLOCKS
1516
1517
1518
1519    .TOC    *       BASIC REGISTER CLOCKS (D, SR, BA, CC)
1520    .MACRO CLK-D                    ;;=  CLKD/YES   ;MUST SPECIFY P2 T OR P3 T
1521    .MACRO CLK-SR                   ;;=  CLKSR/YES  ;MUST SPECIFY P2 T OR P3 T
1522    .MACRO CLK-BA                   ;;=  CLKBA/YES  ;AT P1 T ONLY
1523    .MACRO SET-CC                   ;;=  SCC/YES    ;SETUP HERE, CLOCKED AT P2 T  **OF NEXT UWORD**  ONLY
1524    .MACRO CLK-CC                   ;;=  NULL       ;IN NEXT UWORD, FOR DOCUMENTATION
1525
1526
1527
1528    .TOC    *       REDEFINED FROM SP REWRITE FIELD (RES, COUNTER)
1529    .MACRO LOAD-RES                 ;;=  MOD/LOADREG,LOADRES/YES    ;AT P2 T ONLY, FROM B-BUS<14:11>
1530    .MACRO LOAD-COUNTER             ;;=  MOD/LOADREG,LOADCOUNT/YES  ;DURING ENTIRE UWORD, FROM B-BUS<9:0>
1531
1532
1533
1534    .TOC    *       RES REGISTER CONTROL VALUES (FROM EMIT)
1535    ;LOADED VIA;  EMIT<14:11> -> CSP(XX)<14:11> -> B-BUS<14:11> -> RES<3:0>
1536    .MACRO SENDMUX-0123-SEL         ;;=  EMIT14/1                   ;FOR SHIFT TREE
1537    .MACRO SENDMUX-4567-SEL         ;;=  EMIT14/0                   ;FOR SHIFT TREE
1538    .MACRO SR-LOAD                  ;;=  EMIT13/0,EMIT12/0          ;FOR SR/GUARD
1539    .MACRO SR-LEFT                  ;;=  EMIT13/0,EMIT12/1          ;FOR SR/GUARD
1540    .MACRO SR-RIGHT                 ;;=  EMIT13/1,EMIT12/0          ;FOR SR/GUARD
1541    .MACRO SR-NOP                   ;;=  EMIT13/1,EMIT12/1          ;FOR SR/GUARD
1542    .MACRO GUARD-EN                 ;;=  EMIT11/1                   ;FOR SR/GUARD
1543    .MACRO GUARD-DIS                ;;=  EMIT11/0                   ;FOR SR/GUARD
1544
1545
1546
1547
1548    .TOC    *       CC CONTROL  (FROM EMIT)
1549    ;USED VIA; BUS-U3T-H -> EMIT07-H -> MODIFY-V(1)-H
1550    .MACRO MODIFY-VBIT              ;;=  EMIT07/1
1551    .MACRO NOT-MODIFY-VBIT          ;;=  EMIT07/0
1552
1553
1554
1555    ;.PAGE*********************************************************************************************************************
1556
1557    .TOC    *     BUS CONTROL MACROS
1558    .MACRO DATI-CLKIR               ;;=  BEGIN/YES,SELECT/BUS,BUSCODE/DATI-CLKIR
1559    .MACRO DATI-NOINT               ;;=  BEGIN/YES,SELECT/BUS,BUSCODE/DATI-NOINT
1560    .MACRO DATI                     ;;=  BEGIN/YES,SELECT/BUS,BUSCODE/DATI
1561    .MACRO DATI(P)                  ;;=  BEGIN/YES,SELECT/BUS,BUSCODE/DATI(P)       ;WITH ALTER/ALLOWED
1562    .MACRO DATO                     ;;=  BEGIN/YES,SELECT/BUS,BUSCODE/DATO
1563    .MACRO DATIB                    ;;=  BEGIN/YES,SELECT/BUS,BUSCODE/DATIB
1564    .MACRO DATIB(P)                 ;;=  BEGIN/YES,SELECT/BUS,BUSCODE/DATIB(P)      ;WITH ALTER/ALLOWED
1565    .MACRO DATIP                    ;;=  BEGIN/YES,SELECT/BUS,BUSCODE/DATIP
1566    .MACRO DATOB                    ;;=  BEGIN/YES,SELECT/BUS,BUSCODE/DATOB
```



```
1567    .MACRO INVALIDATE               ;;=  BEGIN/YES,SELECT/BUS,BUSCODE/INVALIDATE
1568
1569
1570
1571    .TOC    *     KT/KJ CONTROL FUNCTIONS
1572
1573          ;THESE BITS ACTUALLY ARISE OUT OF THE BASE MACHINE EXTENSION ROMS,
1574          ;AND AS SUCH AREN'T DIRECTLY ACCESSIBLE FROM THE DCS.  THEY ARE
1575          ;INCLUDED HERE ONLY FOR DOCUMENTATION PURPOSES.
1576
1577    .MACRO KJ-ENABLE                ;;=  KJ/ONE
1578
1579    .MACRO MAINTENANCE              ;;=  UKTEN/ONE
1580    .MACRO CURRENT-MODE             ;;=  UKT01/ONE,UKT00/ZERO
1581    .MACRO KERNAL-MODE              ;;=  UKT01/ONE,UKT00/ONE
1582    .MACRO MT-MODE                  ;;=  UKT01/ZERO,UKT00/ONE
1583    .MACRO MP-MODE                  ;;=  UKT01/ZERO,UKT00/ZERO
1584
1585
1586
1587    ;.PAGE*********************************************************************************************************************
1588
1589    .TOC    *     UCON CONTROL MACROS
1590    .MACRO SET-UCON-CONTROL         ;;=  BEGIN/YES,SELECT/UCON,UCON-LOAD/YES  ;LOAD UCON CONTROL REGISTER AT P0
1591    .MACRO UCON-OPERATION           ;;=  BEGIN/YES,SELECT/UCON,UCON-XFER/YES  ;PERFORM UCON OPERATION
1592
1593
1594
1595    .TOC    *       PROCESSOR UCON CONTROL SETUP
1596    .MACRO UCON-PROC                ;;=  UCON-SEL-PROC/YES                    ;SELECT PROCESSOR
1597    .MACRO EN-CLK-IR[15-00]         ;;=  UCON15/1                             ;ENABLE OPERATIONS
1598    .MACRO EN-CLK-PS[15-12]         ;;=  UCON14/1                             ;
1599    .MACRO EN-CLK-FLAG[8-0]         ;;=  UCON13/1                             ;
1600    .MACRO EN-CLK-FPS[7-4]          ;;=  UCON12/1                             ;
1601    .MACRO EN-CLK-PS[7-4]           ;;=  UCON11/1                             ;
1602    .MACRO EN-CLK-PS[3-0]           ;;=  UCON10/1                             ;
1603    .MACRO EN-CLK-UBPERR[1-00]      ;;=  UCON09/1                             ;
          ;UCON<8:7> ARE NOT USED IN PROCESSOR CONTROL
1605    .MACRO BUSDIN_EMIT[15-00]       ;;=  UCON06/0,UCON05/0                    ;MBMUX SELECT
1606    .MACRO BUSDIN_CUA[14-03]        ;;=  UCON06/0,UCON05/1                    ;
1607    .MACRO BUSDIN_PS[15-00]         ;;=  UCON06/1,UCON05/0                    ;
1608    .MACRO BUSDIN_FLAG[8-0]&FPS[7-0] ;;=  UCON06/1,UCON05/1                   ;
1609
1610
1611
1612

1613    .TOC    *       DCS/WCS/ECS CONTROL
1614    .MACRO UCON-DCS                 ;;=  UCON-SEL-WCS/YES                     ;SELECT DCS
1615    .MACRO BUSDIN_TWUA[11-00]       ;;=  UCON14/0                             ;DCS BUSDIN MUX SEL
1616    .MACRO BUSDIN_ERR&OPS&ERRCOD[11-00]    ;;=  UCON14/1                      ;
1617    .MACRO START-DCS                ;;=  UCON15/1                             ;
1618
1619
1620
```

```
1621    .TOC    *       CACHE/KT UCON CONTROL
1622    .MACRO UCON-CACHE-KT                    ;;=  UCON-SEL-CACHEKT/YES       ;SELECT CACHE / KT UCON FUNCTION
1623          ;UCON<15> NOT USED HERE
1624    .MACRO EN-KT-NO-RELOCATE                ;;=  UCON14/1                   ;INHIBIT KT FROM ANY RELOCATION OF BA -> PBA
1625    .MACRO BUSDIN_BUS-INTERNAL-ADDR[15-00]  ;;=  UCON13/0,UCON12/1          ;FROM INTERNAL ADDR ROM
1626    .MACRO BUSDIN_CPU-INTERNAL-ADDR[15-00]  ;;=  UCON13/1,UCON12/1          ; DITTO ...
1627    .MACRO BUSDIN_MMR2[15-00]               ;;=  UCON11/1,UCON09/0          ;VIRTUAL PC
1628    .MACRO BUSDIN_CACHE-STATUS[15-00]       ;;=  UCON11/1,UCON09/1          ;CACHE INFO
1629    .MACRO BUSDIN_KT-SEL                    ;;=  UCON10/1                   ;FOR PAR-8, PDR-8 ETC
1630    .MACRO KT-WRITE-HIGH                    ;;=  UCON08/1                   ;WRITE REGISTER <15:08>
1631    .MACRO KT-WRITE-LOW                     ;;=  UCON07/1                   ;WRITE REGISTER <07:00>
1632    .MACRO KT-WRITE                         ;;=  UCON08/1,UCON07/1          ;WRITE REGISTER <15:00>
1633    .MACRO KT-SEL-SLR#CCP                   ;;=  UCON06/0,UCON05/0          ;SELECT KT-MUX OUTPUT
1634    .MACRO KT-SEL-MMR0                      ;;=  UCON06/0,UCON05/1          ;
1635    .MACRO KT-SEL-PDR                       ;;=  UCON06/1,UCON05/0          ;
1636    .MACRO KT-SEL-PAR                       ;;=  UCON06/1,UCON05/1          ;
1637
1638
1639
1640
1641    .TOC    *       I/O UCON CONTROL
1642    .MACRO UCON-I-O                 ;;=  UCON-SEL-I-O/YES                   ;SELECT I-O CONTROL
1643
1644
1645
1646    .TOC    *       BUS CONTROL
1647    .MACRO EN-LOAD-DBUF[15-00]      ;;=  UCON15/1                           ;EN LOAD DBUF AT P1
1648    .MACRO BUSDIN_DBUF[15-00]       ;;=  UCON15/1                           ;DBUF ON BUSDIN
1649    .MACRO EN-STATUS-MUX            ;;=  UCON15/0                           ;STATUS-MUX ENABLE ON BUSDIN
1650          ;UCON<14:11> ARE NOT USED IN UCON BUS CONTROL
1651    .MACRO BUSDIN_SERVICE[15-00]    ;;=  UCON10/0,UCON09/1                  ;
1652    .MACRO BUSDIN_JAM[15-00]        ;;=  UCON10/1,UCON09/0                  ;
1653    .MACRO BUSDIN_PBA[15-00]        ;;=  UCON10/1,UCON09/1                  ;
1654    .MACRO DMUX_CACHEDATA[15-00]    ;;=  UCON08/1                           ;
1655    .MACRO EN-BC-FCN-0              ;;=  UCON07/0,UCON06/0,UCON05/0         ;SELECT BUS CONTROL FUNCTION
1656    .MACRO EN-START-DELAY           ;;=  UCON07/0,UCON06/0,UCON05/1         ;
1657    .MACRO EN-CLR-JAM-ERRORS        ;;=  UCON07/0,UCON06/1,UCON05/0         ;
1658    .MACRO EN-CLR-NXM-TIMEOUT       ;;=  UCON07/0,UCON06/1,UCON05/1         ;
1659    .MACRO EN-CLR-PWR-FAIL          ;;=  UCON07/1,UCON06/0,UCON05/0         ;
1660    .MACRO EN-CLR-YELLOW-ZONE       ;;=  UCON07/1,UCON06/0,UCON05/1         ;
1661    .MACRO EN-ALLOW-BG[1]H          ;;=  UCON07/1,UCON06/1,UCON05/0         ;
1662    .MACRO EN-BUS-INIT-UCON         ;;=  UCON07/1,UCON06/1,UCON05/1         ;
1663
1664
1665
1666    .TOC    *       CONSOLE I-O
1667    .MACRO EN-CONSOLE-COMMAND       ;;=  UCON13/0,UCON14/0                  ;SETS UP UCON I-O BITS FOR CONSOLE COMMANDS
1668                                                                            ;ALSO SELECTS STATUS-MUX ON BUSDIN
1669    .MACRO EN-CNSL-NOP              ;;=  UCON13/0,UCON12/0,UCON11/0         ;ENABLE CONSOLE NO OPERATION
1670    .MACRO EN-CLR-COUNTR            ;;=  UCON13/0,UCON12/0,UCON11/1         ;ENABLE CLEAR DIGIT PAIR COUNTER
1671    .MACRO EN-INCR-COUNTR           ;;=  UCON13/0,UCON12/1,UCON11/0         ;ENABLE BUMP TO NEXT DIGIT PAIR
1672    .MACRO EN-CLR-CNSL-SRVC         ;;=  UCON13/0,UCON12/1,UCON11/1         ;ENABLE CLEAR CONSOLE SERVICE RQST FLOP
1673    .MACRO EN-STRB-DISP             ;;=  UCON13/1,UCON12/0,UCON11/0         ;ENABLE WRITE DIGIT PAIR TO DISPLAY LATCH
1674    .MACRO EN-CLR-CNSL              ;;=  UCON13/1,UCON12/0,UCON11/1         ;ENABLE CLEAR CONSOLE LED
```



```
1675    .MACRO EN-SET-CNSL              ;;=  UCON13/1,UCON12/1,UCON11/0         ;ENABLE SET CONSOLE LED
1676    .MACRO EN-SET-DP                ;;=  UCON13/1,UCON12/1,UCON11/1         ;ENABLE SET ALL DP LEDS
1677    .MACRO BUSDIN_CONSOLE[06-00]    ;;=  UCON10/0,UCON09/0                  ;STATUS-MUX SELECT
1678          ;UCON<8:5> ARE NOT USED IN UCON CONSOLE CONTROL
1679
1680
1681
1682    .TOC    *       REMOTE CONSOLE INTERFACE
1683    IN P.S  "EN CONSOLE COMMAND" DOES NOT APPLY TO REMOTE CONSOLE
1684    .MACRO EN-REMSTRB               ;;=  UCON14/1                           ;EN REMOTE CONSOLE STROBE
1685    .MACRO EN-REMCODE1              ;;=  UCON12/1                           ;EN SPECIAL CODE 1
1686    .MACRO EN-REMCODE0              ;;=  UCON11/1                           ;EN SPECIAL CODE 0
1687
1688
1689
1690    !.PAGE*************************************************************************************************************
1691
1692    .TOC    *       DCS ROM EXTENSION MACROS
1693
1694    .TOC    *       GENERAL FUNCTIONS
1695    .MACRO LOAD-ENUA(MICROADDR)     ;;=  LOAD-ENUA-ERRCOD/YES,      ;SPECIFY LOAD
1696                                            ENUA/#MICROADDR         ;AND VALUE
1697    .MACRO LOAD-ERROR(MICROADDR)    ;;=  LOAD-ENUA-ERRCOD/YES       ;FOR EFFECT ONLY - ALWAYS ACCOMPANIES ABOVE
1698    .MACRO BUMP-VERIFY              ;;=  VERIFY/BUMP                ;BUMP VERIFY COUNTER WHEN IN SELF TEST MODE
1699    .MACRO SIGNAL-EOP               ;;=  EOP/SIGNAL                 ;SIGNAL END OF PASS
1700    .MACRO DCS-CTR[XX]              ;;=  LOAD-DCS-CTR/YES,          ;SPECIFY LOAD COUNTER (DCS1)
1701                                            CTR/#XX                 ;AND VALUE
1702
1703
1704
1705    .TOC    *       DAD<11:0> BIT FUNCTIONS
1706    .MACRO NO-DAD                   ;;=  DAD/NO-DAD                 ;DON'T ASSERT DAD BITS
1707    .MACRO FIRST-1-OF-2             ;;=  DAD/FIRST-1-OF-2           ;SELECT CSP CONSTANT 1/2, FIRST USE
1708    .MACRO SECOND-1-OF-2            ;;=  DAD/SECOND-1-OF-2          ;SELECT CSP CONSTANT 1/2, SECOND USE
1709    .MACRO WRITE-BYTE               ;;=  DAD/WRITE-BYTE             ;BYTE WRITE ENABLE TO ASP/BSP
1710    .MACRO BYTE-WRITE               ;;=  DAD/WRITE-BYTE             ;  "
1711
1712
1713
1714    .TOC    *       DIAGNOSTIC MODE BUT ENABLES
1715
1716    .MACRO BUTD[SCOPE]              ;;=  SCOPE/ENABLED              ;ENABLE SCOPE LOOPING CHECK
1717    .MACRO BUTD[ERROR]              ;;=  SCOPE/ENABLED              ;FORCES NUA00=0 IF ERROR[1]H SET
1718
1719    .MACRO BUTD[VERIFY-MODE]        ;;=  NULL                       ;CHECK IMPLICITLY FOR VERIFY MODE
1720                                                                    ;"SIGNAL EOP" MUST BE PRESENT IN SAME WORD
1721                                                                    ;FORCES NUA01=0 IF VERIFY SWITCH SET
1722
1723    .MACRO BUTD[EOP-OVERFLOW]       ;;=  NULL                       ;AFTER "SIGNAL EOP" GIVEN, ADDRESS IS FORCED
1724                                                                    ;TO (4000) IF HARDWARE EOP/VERIFY COUNTER
1725                                                                    ;HAS NOT YET OVERFLOWED, FORCING ANOTHER
1726                                                                    ;PASS,  ELSE ADDRESS IN UPF IS TAKEN UNMODIFIED.
1727
1728
```

```

;.PAGE**************************************************************************************************************

.TOC    *       MICROBRANCH FIELD MACROS
;(SEE <UBF> FIELD DESCRIPTION FOR FULL INFO)

.MACRO BUT(XX)                  ;:=  UBF/#XX                     ;INACTIVE, FULL WIDTH
.MACRO BUTR(XX)                 ;:=  UBF/#XX                     ;INACTIVE, RESTRICTED WIDTH

.MACRO BUTA(XX)                 ;:=  UBF/#XX                     ;ACTIVE, FULL WIDTH
.MACRO BUTRA(XX)                ;:=  UBF/#XX                     ;ACTIVE, RESTRICTED WIDTH

.MACRO TEST(XX)                 ;:=  MULTIPLE/#XX                ;FOR BUTR(MULTIPLE) SETUP
.MACRO BUTM(XX)                 ;:=  MULTIPLE/#XX,UBF/#XX        ;A MULTIPLE BUTR



;.PAGE**************************************************************************************************************

.TOC    *       MISCELLANEOUS

.TOC    *       OTHER SOURCES ENABLED FOR A-BUS
.MACRO SR                       ;:=  AEN/XMUX,ASELO/SR
.MACRO FLTPT                    ;:=  AEN/XMUX,ASELO/FLTPT



.TOC    *       PAGING, RETURN REGISTER

        ;PAGE FIELD ONLY:
.MACRO PAGE(X)                  ;:=  NEXT-PAGE/#X

        ;PAGE FIELD AND BUT(SUBR B):
.MACRO GOTO-PAGE(X)             ;:=  NEXT-PAGE/#X,UBF/SUBR-B

        ;RETURN REGISTER <- D<14:03>, PAGE <- EMIT<02:00> ON BUTA(SUBR-A)
.MACRO RETURN_D[14-03]          ;:=  UBF/SUBR-A



;.PAGE**************************************************************************************************************

.TOC    *   ADVANCED OPERATIONS



.TOC    *       DATA INTO CSP, AT P3 ONLY

        ;N.B.:  BUSDIN IS ANY BUT EMIT (OVERLAPS BSEL<1:0>)
.MACRO CSPB[14]_BUSDIN          ;:=  CSPB(B14),WR-CSP
.MACRO CSPB[15]_BUSDIN          ;:=  CSPB(B15),WR-CSP
.MACRO CSPB[16]_BUSDIN          ;:=  CSPB(B16),WR-CSP
.MACRO CSPB[17]_BUSDIN          ;:=  CSPB(B17),WR-CSP

```



```
        ;N.B.:  GETS WHATEVER IS ON BUSDIN
.MACRO CSPD[00]_BUSDIN          ;:=  CSPD(D00),WR-CSP
.MACRO CSPD[01]_BUSDIN          ;:=  CSPD(D01),WR-CSP
.MACRO CSPD[02]_BUSDIN          ;:=  CSPD(D02),WR-CSP
.MACRO CSPD[03]_BUSDIN          ;:=  CSPD(D03),WR-CSP
.MACRO CSPD[04]_BUSDIN          ;:=  CSPD(D04),WR-CSP
.MACRO CSPD[05]_BUSDIN          ;:=  CSPD(D05),WR-CSP
.MACRO CSPD[06]_BUSDIN          ;:=  CSPD(D06),WR-CSP
.MACRO CSPD[07]_BUSDIN          ;:=  CSPD(D07),WR-CSP
.MACRO CSPD[10]_BUSDIN          ;:=  CSPD(D10),WR-CSP
.MACRO CSPD[11]_BUSDIN          ;:=  CSPD(D11),WR-CSP
.MACRO CSPD[12]_BUSDIN          ;:=  CSPD(D12),WR-CSP
.MACRO CSPD[13]_BUSDIN          ;:=  CSPD(D13),WR-CSP
.MACRO CSPD[14]_BUSDIN          ;:=  CSPD(D14),WR-CSP
.MACRO CSPD[15]_BUSDIN          ;:=  CSPD(D15),WR-CSP
.MACRO CSPD[16]_BUSDIN          ;:=  CSPD(D16),WR-CSP
.MACRO CSPD[17]_BUSDIN          ;:=  CSPD(D17),WR-CSP

        ;N.B.:  REQUIRED THAT BUSDIN_EMIT[15-00] PREVIOUSLY SET UP
.MACRO CSPD[00]_EMIT            ;:=  CSPD(D00),WR-CSP
.MACRO CSPD[01]_EMIT            ;:=  CSPD(D01),WR-CSP
.MACRO CSPD[02]_EMIT            ;:=  CSPD(D02),WR-CSP
.MACRO CSPD[03]_EMIT            ;:=  CSPD(D03),WR-CSP
.MACRO CSPD[04]_EMIT            ;:=  CSPD(D04),WR-CSP
.MACRO CSPD[05]_EMIT            ;:=  CSPD(D05),WR-CSP
.MACRO CSPD[06]_EMIT            ;:=  CSPD(D06),WR-CSP
.MACRO CSPD[07]_EMIT            ;:=  CSPD(D07),WR-CSP
.MACRO CSPD[10]_EMIT            ;:=  CSPD(D10),WR-CSP
.MACRO CSPD[11]_EMIT            ;:=  CSPD(D11),WR-CSP
.MACRO CSPD[12]_EMIT            ;:=  CSPD(D12),WR-CSP
.MACRO CSPD[13]_EMIT            ;:=  CSPD(D13),WR-CSP
.MACRO CSPD[14]_EMIT            ;:=  CSPD(D14),WR-CSP
.MACRO CSPD[15]_EMIT            ;:=  CSPD(D15),WR-CSP
.MACRO CSPD[16]_EMIT            ;:=  CSPD(D16),WR-CSP
.MACRO CSPD[17]_EMIT            ;:=  CSPD(D17),WR-CSP



.TOC    *       MISC CONSTANTS INTO ASP, BSP, AT P2-T * P3

.MACRO A&BSPHI[C100000]_D       ;:=  ASP(C100000),BSP(C100000),WR(AB,H,B)
.MACRO A&BSPHI[C000200]_D       ;:=  ASP(C000200),BSP(C000200),WR(AB,H,B)
.MACRO A&BSPHI[C000000]_D       ;:=  ASP(C000000),BSP(C000000),WR(AB,H,B)
.MACRO A&BSPHI[C177777]_D       ;:=  ASP(C177777),BSP(C177777),WR(AB,H,B)
.MACRO A&BSPHI[C000001]_D       ;:=  ASP(C000001),BSP(C000001),WR(AB,H,B)
.MACRO A&BSPHI[C052525]_D       ;:=  ASP(C052525),BSP(C052525),WR(AB,H,B)

.MACRO A&BSPHI[C125252]_D       ;:=  ASP(C125252),BSP(C125252),WR(AB,H,B)

.MACRO A&BSPHI[C100000]_D-(B)   ;:=  BSP(C100000),WR(AB,H,B)
.MACRO A&BSPHI[C000200]_D-(B)   ;:=  BSP(C000200),WR(AB,H,B)
.MACRO A&BSPHI[C000000]_D-(B)   ;:=  BSP(C000000),WR(AB,H,B)
.MACRO A&BSPHI[C177777]_D-(B)   ;:=  BSP(C177777),WR(AB,H,B)
.MACRO A&BSPHI[C000001]_D-(B)   ;:=  BSP(C000001),WR(AB,H,B)
.MACRO A&BSPHI[C052525]_D-(B)   ;:=  BSP(C052525),WR(AB,H,B)
```

```
1837    .MACRO A#BSPHI[C125252]_D=[B]   ;;=  BSP(C125252),WR(AB,H,B)
1838
1839    .MACRO A#BSPHI[C100000]_D=[A]   ;;=  ASP(C100000),WR(AB,H,A)
1840    .MACRO A#BSPHI[C000200]_D=[A]   ;;=  ASP(C000200),WR(AB,H,A)
1841    .MACRO A#BSPHI[C000000]_D=[A]   ;;=  ASP(C000000),WR(AB,H,A)
1842    .MACRO A#BSPHI[C177777]_D=[A]   ;;=  ASP(C177777),WR(AB,H,A)
1843    .MACRO A#BSPHI[C000001]_D=[A]   ;;=  ASP(C000001),WR(AB,H,A)
1844    .MACRO A#BSPHI[C052525]_D=[A]   ;;=  ASP(C052525),WR(AB,H,A)
1845    .MACRO A#BSPHI[C125252]_D=[A]   ;;=  ASP(C125252),WR(AB,H,A)
1846
1847
1848
1849    .TOC    *       DATA INTO ASP, BSP, AT P2=T + P3
1850
1851    .MACRO ASPLO[17]_CSPB(XX)       ;;=  B,ASPLO(R17),CSPB(0XX),CLK=D,P2=T,WR(A,L,A)
1852    .MACRO ASPLO[17]_CSPD(XX)       ;;=  B,ASPLO(P17),CSPD(0XX),CLK=D,P2=T,WR(A,L,A)
1853    .MACRO PC_D                     ;;=  PC=A,NR(AB,L,A)
1854    .MACRO RS_D                     ;;=  RS=A,WR(AB,L,A)
1855
1856    .MACRO ASPLO[00]_D              ;;=  ASP(R00),WR(A,L,A)
1857    .MACRO ASPLO[01]_D              ;;=  ASP(R01),WR(A,L,A)
1858    .MACRO ASPLO[02]_D              ;;=  ASP(R02),WR(A,L,A)
1859    .MACRO ASPLO[03]_D              ;;=  ASP(R03),WR(A,L,A)
1860    .MACRO ASPLO[04]_D              ;;=  ASP(R04),WR(A,L,A)
1861    .MACRO ASPLO[05]_D              ;;=  ASP(R05),WR(A,L,A)
1862    .MACRO ASPLO[06]_D              ;;=  ASP(R06),WR(A,L,A)
1863    .MACRO ASPLO[07]_D              ;;=  ASP(R07),WR(A,L,A)
1864    .MACRO ASPLO[10]_D              ;;=  ASP(R10),WR(A,L,A)
1865    .MACRO ASPLO[11]_D              ;;=  ASP(R11),WR(A,L,A)
1866    .MACRO ASPLO[12]_D              ;;=  ASP(R12),WR(A,L,A)
1867    .MACRO ASPLO[13]_D              ;;=  ASP(R13),WR(A,L,A)
1868    .MACRO ASPLO[14]_D              ;;=  ASP(R14),WR(A,L,A)
1869    .MACRO ASPLO[15]_D              ;;=  ASP(R15),WR(A,L,A)
1870    .MACRO ASPLO[16]_D              ;;=  ASP(R16),WR(A,L,A)
1871    .MACRO ASPLO[17]_D              ;;=  ASP(R17),WR(A,L,A)
1872
1873    .MACRO ASPHI[00]_D              ;;=  ASP(R00),WR(A,H,A)
1874    .MACRO ASPHI[01]_D              ;;=  ASP(R01),WR(A,H,A)
1875    .MACRO ASPHI[02]_D              ;;=  ASP(R02),WR(A,H,A)
1876    .MACRO ASPHI[03]_D              ;;=  ASP(R03),WR(A,H,A)
1877    .MACRO ASPHI[04]_D              ;;=  ASP(R04),WR(A,H,A)
1878    .MACRO ASPHI[05]_D              ;;=  ASP(R05),WR(A,H,A)
1879    .MACRO ASPHI[06]_D              ;;=  ASP(R06),WR(A,H,A)
1880    .MACRO ASPHI[07]_D              ;;=  ASP(R07),WR(A,H,A)
1881    .MACRO ASPHI[10]_D              ;;=  ASP(P10),WR(A,H,A)
1882    .MACRO ASPHI[11]_D              ;;=  ASP(R11),WR(A,H,A)
1883    .MACRO ASPHI[12]_D              ;;=  ASP(R12),WR(A,H,A)
1884    .MACRO ASPHI[13]_D              ;;=  ASP(P13),WR(A,H,A)
1885    .MACRO ASPHI[14]_D              ;;=  ASP(R14),WR(A,H,A)
1886    .MACRO ASPHI[15]_D              ;;=  ASP(R15),WR(A,H,A)
1887    .MACRO ASPHI[16]_D              ;;=  ASP(R16),WR(A,H,A)
1888    .MACRO ASPHI[17]_D              ;;=  ASP(R17),WR(A,H,A)
1889
1890    .MACRO BSPLO[00]_D              ;;=  BSP(R00),WR(B,L,B)
```



```
1891    .MACRO BSPLO[01]_D              ;;=  BSP(R01),WR(B,L,B)
1892    .MACRO BSPLO[02]_D              ;;=  BSP(R02),WR(B,L,B)
1893    .MACRO BSPLO[03]_D              ;;=  BSP(R03),WR(B,L,B)
1894    .MACRO BSPLO[04]_D              ;;=  BSP(R04),WR(B,L,B)
1895    .MACRO BSPLO[05]_D              ;;=  BSP(R05),WR(B,L,B)
1896    .MACRO BSPLO[06]_D              ;;=  BSP(R06),WR(B,L,B)
1897    .MACRO BSPLO[07]_D              ;;=  BSP(R07),WR(B,L,B)
1898    .MACRO BSPLO[10]_D              ;;=  BSP(R10),WR(B,L,B)
1899    .MACRO BSPLO[11]_D              ;;=  BSP(R11),WR(B,L,B)
1900    .MACRO BSPLO[12]_D              ;;=  BSP(R12),WR(B,L,B)
1901    .MACRO BSPLO[13]_D              ;;=  BSP(R13),WR(B,L,B)
1902    .MACRO BSPLO[14]_D              ;;=  BSP(R14),WR(B,L,B)
1903    .MACRO BSPLO[15]_D              ;;=  BSP(R15),WR(B,L,B)
1904    .MACRO BSPLO[16]_D              ;;=  BSP(R16),WR(B,L,B)
1905    .MACRO BSPLO[17]_D              ;;=  BSP(P17),WR(B,L,B)
1906
1907    .MACRO BSPHI[00]_D              ;;=  BSP(R00),WR(B,H,B)
1908    .MACRO BSPHI[01]_D              ;;=  BSP(R01),WR(B,H,B)
1909    .MACRO BSPHI[02]_D              ;;=  BSP(R02),WR(B,H,B)
1910    .MACRO BSPHI[03]_D              ;;=  BSP(R03),WR(B,H,B)
1911    .MACRO BSPHI[04]_D              ;;=  BSP(R04),WR(B,H,B)
1912    .MACRO BSPHI[05]_D              ;;=  BSP(R05),WR(B,H,B)
1913    .MACRO BSPHI[06]_D              ;;=  BSP(R06),WR(B,H,B)
1914    .MACRO BSPHI[07]_D              ;;=  BSP(R07),WR(B,H,B)
1915    .MACRO BSPHI[10]_D              ;;=  BSP(R10),WR(B,H,B)
1916    .MACRO BSPHI[11]_D              ;;=  BSP(R11),WR(B,H,B)
1917    .MACRO BSPHI[12]_D              ;;=  BSP(R12),WR(B,H,B)
1918    .MACRO BSPHI[13]_D              ;;=  BSP(R13),WR(B,H,B)
1919    .MACRO BSPHI[14]_D              ;;=  BSP(R14),WR(B,H,B)
1920    .MACRO BSPHI[15]_D              ;;=  BSP(R15),WR(B,H,B)
1921    .MACRO BSPHI[16]_D              ;;=  BSP(R16),WR(B,H,B)
1922    .MACRO BSPHI[17]_D              ;;=  BSP(R17),WR(B,H,B)
1923
1924    .MACRO A#BSPLO[00]_D            ;;=  ASP(R00),BSP(R00),WR(AB,L,A)
1925    .MACRO A#BSPLO[01]_D            ;;=  ASP(R01),BSP(R01),WR(AB,L,A)
1926    .MACRO A#BSPLO[02]_D            ;;=  ASP(R02),BSP(R02),WR(AB,L,A)
1927    .MACRO A#BSPLO[03]_D            ;;=  ASP(R03),BSP(R03),WR(AB,L,A)
1928    .MACRO A#BSPLO[04]_D            ;;=  ASP(R04),BSP(R04),WR(AB,L,A)
1929    .MACRO A#BSPLO[05]_D            ;;=  ASP(R05),BSP(R05),WR(AB,L,A)
1930    .MACRO A#BSPLO[06]_D            ;;=  ASP(R06),BSP(R06),WR(AB,L,A)
1931    .MACRO A#BSPLO[07]_D            ;;=  ASP(R07),BSP(R07),WR(AB,L,A)
1932    .MACRO A#BSPLO[10]_D            ;;=  ASP(R10),BSP(R10),WR(AB,L,A)
1933    .MACRO A#BSPLO[11]_D            ;;=  ASP(R11),BSP(R11),WR(AB,L,A)
1934    .MACRO A#BSPLO[12]_D            ;;=  ASP(R12),BSP(R12),WR(AB,L,A)
1935    .MACRO A#BSPLO[13]_D            ;;=  ASP(R13),BSP(R13),WR(AB,L,A)
1936    .MACRO A#BSPLO[14]_D            ;;=  ASP(P14),BSP(R14),WR(AB,L,A)

1937    .MACRO A#BSPLO[15]_D            ;;=  ASP(P15),BSP(R15),WR(AB,L,A)
1938    .MACRO A#BSPLO[16]_D            ;;=  ASP(R16),BSP(R16),WR(AB,L,A)
1939    .MACRO A#BSPLO[17]_D            ;;=  ASP(P17),BSP(R17),WR(AB,L,A)
1940
1941    .MACRO A#BSPHI[00]_D            ;;=  ASP(R00),BSP(R00),WR(AB,H,A)
1942    .MACRO A#BSPHI[01]_D            ;;=  ASP(P01),BSP(R01),WR(AB,H,A)
1943    .MACRO A#BSPHI[02]_D            ;;=  ASP(R021,BSP(R02),WR(AB,H,A)
1944    .MACRO A#BSPHI[03]_D            ;;=  ASP(R03),BSP(R03),WR(AB,H,A)
```

```
1945    .MACRO A=BSPHI[04]_D            ;;=  ASP(P04),BSP(R04),WR(AB,H,A)
1946    .MACRO A=BSPHI[05]_D            ;;=  ASP(R05),BSP(R05),WR(AB,H,A)
1947    .MACRO A=BSPHI[06]_D            ;;=  ASP(R06),BSP(R06),WR(AB,H,A)
1948    .MACRO A=BSPHI[07]_D            ;;=  ASP(P07),BSP(R07),WR(AB,H,A)
1949    .MACRO A=BSPHI[10]_D            ;;=  ASP(R10),BSP(R10),WR(AB,H,A)
1950    .MACRO A=BSPHI[11]_D            ;;=  ASP(R11),BSP(R11),WR(AB,H,A)
1951    .MACRO A=BSPHI[12]_D            ;;=  ASP(R12),BSP(R12),WR(AB,H,A)
1952    .MACRO A=BSPHI[13]_D            ;;=  ASP(R13),BSP(R13),WR(AB,H,A)
1953    .MACRO A=BSPHI[14]_D            ;;=  ASP(R14),BSP(R14),WR(AB,H,A)
1954    .MACRO A=BSPHI[15]_D            ;;=  ASP(R15),BSP(R15),WR(AB,H,A)
1955    .MACRO A=BSPHI[16]_D            ;;=  ASP(R16),BSP(R16),WR(AB,H,A)
1956    .MACRO A=BSPHI[17]_D            ;;=  ASP(P17),BSP(R17),WR(AB,H,A)
1957
1958    .MACRO A=BSPLO[00]_D=[A]        ;;=  ASP(R00),WR(AB,L,A)
1959    .MACRO A=BSPLO[01]_D=[A]        ;;=  ASP(R01),WR(AB,L,A)
1960    .MACRO A=BSPLO[02]_D=[A]        ;;=  ASP(R02),WR(AB,L,A)
1961    .MACRO A=BSPLO[03]_D=[A]        ;;=  ASP(R03),WR(AB,L,A)
1962    .MACRO A=BSPLO[04]_D=[A]        ;;=  ASP(R04),WR(AB,L,A)
1963    .MACRO A=BSPLO[05]_D=[A]        ;;=  ASP(R05),WR(AB,L,A)
1964    .MACRO A=BSPLO[06]_D=[A]        ;;=  ASP(R06),WR(AB,L,A)
1965    .MACRO A=BSPLO[07]_D=[A]        ;;=  ASP(P07),WR(AB,L,A)
1966    .MACRO A=BSPLO[10]_D=[A]        ;;=  ASP(R10),WR(AB,L,A)
1967    .MACRO A=BSPLO[11]_D=[A]        ;;=  ASP(R11),WR(AB,L,A)
1968    .MACRO A=BSPLO[12]_D=[A]        ;;=  ASP(R12),WR(AB,L,A)
1969    .MACRO A=BSPLO[13]_D=[A]        ;;=  ASP(R13),WR(AB,L,A)
1970    .MACRO A=BSPLO[14]_D=[A]        ;;=  ASP(P14),WR(AB,L,A)
1971    .MACRO A=BSPLO[15]_D=[A]        ;;=  ASP(R15),WR(AB,L,A)
1972    .MACRO A=BSPLO[16]_D=[A]        ;;=  ASP(R16),WR(AB,L,A)
1973    .MACRO A=BSPLO[17]_D=[A]        ;;=  ASP(R17),WR(AB,L,A)
1974
1975    .MACRO A=BSPHI[00]_D=[A]        ;;=  ASP(R00),WR(AB,H,A)
1976    .MACRO A=BSPHI[01]_D=[A]        ;;=  ASP(P01),WR(AB,H,A)
1977    .MACRO A=BSPHI[02]_D=[A]        ;;=  ASP(R02),WR(AB,H,A)
1978    .MACRO A=BSPHI[03]_D=[A]        ;;=  ASP(R03),WR(AB,H,A)
1979    .MACRO A=BSPHI[04]_D=[A]        ;;=  ASP(R04),WR(AB,H,A)
1980    .MACRO A=BSPHI[05]_D=[A]        ;;=  ASP(R05),WR(AB,H,A)
1981    .MACRO A=BSPHI[06]_D=[A]        ;;=  ASP(R06),WR(AB,H,A)
1982    .MACRO A=BSPHI[07]_D=[A]        ;;=  ASP(R07),WR(AB,H,A)
1983    .MACRO A=BSPHI[10]_D=[A]        ;;=  ASP(R10),WR(AB,H,A)
1984    .MACRO A=BSPHI[11]_D=[A]        ;;=  ASP(R11),WR(AB,H,A)
1985    .MACRO A=BSPHI[12]_D=[A]        ;;=  ASP(P12),WR(AB,H,A)
1986    .MACRO A=BSPHI[13]_D=[A]        ;;=  ASP(R13),WR(AB,H,A)
1987    .MACRO A=BSPHI[14]_D=[A]        ;;=  ASP(P14),WR(AB,H,A)
1988    .MACRO A=BSPHI[15]_D=[A]        ;;=  ASP(P15),WR(AB,H,A)
1989    .MACRO A=BSPHI[16]_D=[A]        ;;=  ASP(R16),WR(AB,H,A)
1990    .MACRO A=BSPHI[17]_D=[A]        ;;=  ASP(R17),WR(AB,H,A)
1991
1992    .MACRO A=BSPLO[00]_D=[B]        ;;=  BSP(P00),WR(AB,L,B)
1993    .MACRO A=BSPLO[01]_D=[B]        ;;=  BSP(R01),WR(AB,L,B)
1994    .MACRO A=BSPLO[02]_D=[B]        ;;=  BSP(R02),WR(AB,L,B)
1995    .MACRO A=BSPLO[03]_D=[B]        ;;=  BSP(P03),WR(AB,L,B)
1996    .MACRO A=BSPLO[04]_D=[B]        ;;=  BSP(R04),WR(AB,L,B)
1997    .MACRO A=BSPLO[05]_D=[B]        ;;=  BSP(R05),WR(AB,L,B)
1998    .MACRO A=BSPLO[06]_D=[B]        ;;=  BSP(R06),WR(AB,L,B)
```



```
1999    .MACRO A=BSPLO[07]_D=[B]        ;;=  BSP(P07),WR(AB,L,B)
2000    .MACRO A=BSPLO[10]_D=[B]        ;;=  BSP(R10),WR(AB,L,B)
2001    .MACRO A=BSPLO[11]_D=[B]        ;;=  BSP(R11),WR(AB,L,B)
2002    .MACRO A=BSPLO[12]_D=[B]        ;;=  BSP(R12),WR(AB,L,B)
2003    .MACRO A=BSPLO[13]_D=[B]        ;;=  BSP(R13),WR(AB,L,B)
2004    .MACRO A=BSPLO[14]_D=[B]        ;;=  BSP(R14),WR(AB,L,B)
2005    .MACRO A=BSPLO[15]_D=[B]        ;;=  BSP(R15),WR(AB,L,B)
2006    .MACRO A=BSPLO[16]_D=[B]        ;;=  BSP(R16),WR(AB,L,B)
2007    .MACRO A=BSPLO[17]_D=[B]        ;;=  BSP(R17),WR(AB,L,B)
2008
2009    .MACRO A=BSPHI[00]_D=[B]        ;;=  BSP(R00),WR(AB,H,B)
2010    .MACRO A=BSPHI[01]_D=[B]        ;;=  BSP(R01),WR(AB,H,B)
2011    .MACRO A=BSPHI[02]_D=[B]        ;;=  BSP(R02),WR(AB,H,B)
2012    .MACRO A=BSPHI[03]_D=[B]        ;;=  BSP(P03),WR(AB,H,B)
2013    .MACRO A=BSPHI[04]_D=[B]        ;;=  BSP(P04),WR(AB,H,B)
2014    .MACRO A=BSPHI[05]_D=[B]        ;;=  BSP(R05),WR(AB,H,B)
2015    .MACRO A=BSPHI[06]_D=[B]        ;;=  BSP(R06),WR(AB,H,B)
2016    .MACRO A=BSPHI[07]_D=[B]        ;;=  BSP(R07),WR(AB,H,B)
2017    .MACRO A=BSPHI[10]_D=[B]        ;;=  BSP(R10),WR(AB,H,B)
2018    .MACRO A=BSPHI[11]_D=[B]        ;;=  BSP(R11),WR(AB,H,B)
2019    .MACRO A=BSPHI[12]_D=[B]        ;;=  BSP(P12),WR(AB,H,B)
2020    .MACRO A=BSPHI[13]_D=[B]        ;;=  BSP(P13),WR(AB,H,B)
2021    .MACRO A=BSPHI[14]_D=[B]        ;;=  BSP(R14),WR(AB,H,B)
2022    .MACRO A=BSPHI[15]_D=[B]        ;;=  BSP(R15),WR(AB,H,B)
2023    .MACRO A=BSPHI[16]_D=[B]        ;;=  BSP(R16),WR(AB,H,B)
2024    .MACRO A=BSPHI[17]_D=[B]        ;;=  BSP(P17),WR(AB,H,B)
2025
2026    .MACRO ASPLO(DF)_D              ;;=  ASP=ADDRS=R[DF],WR(A,L,A)
2027    .MACRO ASPHI(DF)_D              ;;=  ASP=ADDRS=R[DF],WR(A,H,A)
2028    .MACRO BSPLO(DF)_D              ;;=  BSP=ADDRS=R[DF],WR(B,L,B)
2029    .MACRO BSPHI(DF)_D              ;;=  BSP=ADDRS=R[DF],WR(B,H,B)
2030
2031    .MACRO ASPLO(SF)_D              ;;=  ASP=ADDRS=R(SF),WR(A,L,A)
2032    .MACRO ASPHI(SF)_D              ;;=  ASP=ADDRS=R(SF),WR(A,H,A)
2033    .MACRO BSPLO(SF)_D              ;;=  BSP=ADDRS=R(SF),WR(B,L,B)
2034    .MACRO BSPHI(SF)_D              ;;=  BSP=ADDRS=R(SF),WR(B,H,B)
2035
2036    .MACRO A=BSPLO(DF)_D=[A]        ;;=  ASP=ADDRS=R[DF],WR(AB,L,A)
2037    .MACRO A=BSPHI(DF)_D=[A]        ;;=  ASP=ADDRS=R[DF],WR(AB,H,A)
2038    .MACRO A=BSPLO(DF)_D=[B]        ;;=  BSP=ADDRS=R[DF],WR(AB,L,B)
2039    .MACRO A=BSPHI(DF)_D=[B]        ;;=  BSP=ADDRS=R[DF],WR(AB,H,B)
2040
2041    .MACRO A=BSPLO(SF)_D=[A]        ;;=  ASP=ADDRS=R(SF),WR(AB,L,A)
2042    .MACRO A=BSPHI(SF)_D=[A]        ;;=  ASP=ADDRS=R(SF),WR(AB,H,A)
2043    .MACRO A=BSPLO(SF)_D=[B]        ;;=  BSP=ADDRS=R(SF),WR(AB,L,B)
2044    .MACRO A=BSPHI(SF)_D=[B]        ;;=  BSP=ADDRS=R(SF),WR(AB,H,B)
2045
2046    .MACRO A=BSPLO(SF)_D            ;;=  ASP=ADDRS=R(SF),BSP=ADDRS=R(SF),WR(AB,L,A)
2047    .MACRO A=BSPLO(DF)_D            ;;=  ASP=ADDRS=R[DF],BSP=ADDRS=R[DF],WR(AB,L,A)

2048    .MACRO A=BSPHI(SF)_D            ;;=  ASP=ADDRS=R(SF),BSP=ADDRS=R(SF),WR(AB,H,A)
2049    .MACRO A=BSPHI(DF)_D            ;;=  ASP=ADDRS=R[DF],BSP=ADDRS=R[DF],WR(AB,H,A)
2050
2051
2052
```

```
;.PAGE==================================================================================================

.TOC    *       D AND SR <- (BUS-A FCN BUS-B), AT P2-T OR P3-T

            ;LOGIC FUNCTIONS:
.MACRO SR_ZERO                  ;;= ZERO,CLK-SR
.MACRO SR_A-XOR-B               ;;= A-XOR-B,CLK-SR
.MACRO SR_B                     ;;= B,CLK-SR
.MACRO SR_A-AND-B               ;;= A-AND-B,CLK-SR
.MACRO SR_A-IOR-B               ;;= A-IOR-B,CLK-SR
.MACRO SR_A                     ;;= A,CLK-SR
.MACRO SR_NOT-A                 ;;= NOT-A,CLK-SR
.MACRO SR_NOT-A-AND-B           ;;= NOT-A-AND-B,CLK-SR
.MACRO SR_A-AND-NOT-B           ;;= A-AND-NOT-B,CLK-SR
.MACRO D_ZERO                   ;;= ZERO,CLK-D
.MACRO D_A-XOR-B                ;;= A-XOR-B,CLK-D
.MACRO D_B                      ;;= B,CLK-D
.MACRO D_A-AND-B                ;;= A-AND-B,CLK-D
.MACRO D_A-IOR-B                ;;= A-IOR-B,CLK-D
.MACRO D_A                      ;;= A,CLK-D
.MACRO D_NOT-A                  ;;= NOT-A,CLK-D
.MACRO D_NOT-A-AND-B            ;;= NOT-A-AND-B,CLK-D
.MACRO D_A-AND-NOT-B            ;;= A-AND-NOT-B,CLK-D

            ;ARITH FUNCTIONS:
.MACRO D_DIVIDE-STEP                    ;;= DIVIDE,CLK-D
.MACRO D_A-PLUS-B                       ;;= A-PLUS-B,CLK-D
.MACRO D_A-PLUS-B-PLUS-0                ;;= A-PLUS-B,CLK-D
.MACRO D_A-MINUS-B                      ;;= A-MINUS-B,CLK-D
.MACRO D_A-PLUS-B-PLUS-PS(C)            ;;= A-PLUS-B-PLUS-PS(C),CLK-D
.MACRO D_A-PLUS-B-PLUS-D(C)             ;;= A-PLUS-B-PLUS-D(C),CLK-D
.MACRO D_A-PLUS-NOT-B-PLUS-D(C)         ;;= A-PLUS-NOT-B-PLUS-D(C),CLK-D
.MACRO D_A-PLUS-B-PLUS-1                ;;= A-PLUS-B-PLUS-1,CLK-D
.MACRO SR_DIVIDE-STEP                   ;;= DIVIDE,CLK-SR
.MACRO SR_A-PLUS-B                      ;;= A-PLUS-B,CLK-SR
.MACRO SR_A-PLUS-B-PLUS-0               ;;= A-PLUS-B,CLK-SR
.MACRO SR_A-MINUS-B                     ;;= A-MINUS-B,CLK-SR
.MACRO SR_A-PLUS-B-PLUS-PS(C)           ;;= A-PLUS-B-PLUS-PS(C),CLK-SR
.MACRO SR_A-PLUS-B-PLUS-D(C)            ;;= A-PLUS-B-PLUS-D(C),CLK-SR
.MACRO SR_A-PLUS-NOT-B-PLUS-D(C)        ;;= A-PLUS-NOT-B-PLUS-D(C),CLK-SR
.MACRO SR_A-PLUS-B-PLUS-1               ;;= A-PLUS-B-PLUS-1,CLK-SR




.TOC    *       D(C) GETS SET

.MACRO D(C)_CINMUX              ;;=  CLK-D,COUT_CIN
.MACRO D(C)_1                   ;;=  CLK-D,COUT_CIN              ;NEEDS SPECIFIC ALU/---
.MACRO D(C)_0                   ;;=  CLK-D,COUT_CIN              ;NEEDS SPECIFIC ALU/---
.MACRO D(C)_PS(C)               ;;=  CLK-D,COUT_PS(C)
.MACRO D(C)_ALU00               ;;=  CLK-D,COUT_ALU00
.MACRO D(C)_ALU07               ;;=  CLK-D,COUT_ALU07
.MACRO D(C)_ALU15               ;;=  CLK-D,COUT_ALU15
```



```
.MACRO D(C)_COUT07              ;;=  CLK-D,COUT_COUT07
.MACRO D(C)_COUT15              ;;=  CLK-D,COUT_COUT15
.MACRO D(C)_D(C)                ;;=  CLK-D,COUT_D(C)
.MACRO SAVE-D(C)                ;;=  CLK-D,COUT_D(C)




.TOC    *       D-REGISTER <- (BBUS * ABUS), BITWISE, AT P2-T OR P3-T

            ;N.B.: SHIFT TREE ENABLED SEPARATELY
.MACRO D_D-SHIFTED-XOR-CSPB(XX)                 ;;=  A-XOR-B,CSPB(@XX),CLK-D
.MACRO D_D-SHIFTED-XOR-BSPHI(XX)                ;;=  A-XOR-B,BSPHI(@XX),CLK-D

.MACRO D_FLTPT-XOR-CSPB(XX)                     ;;=  A-XOR-B,FLTPT,CSPB(@XX),CLK-D
.MACRO D_FLTPT-XOR-CSPD(XX)                     ;;=  A-XOR-B,FLTPT,CSPD(@XX),CLK-D
.MACRO D_FLTPT-XOR-BSPHI(XX)                    ;;=  A-XOR-B,FLTPT,BSPHI(@XX),CLK-D

.MACRO D_SR-XOR-CSPB(XX)                        ;;=  A-XOR-B,SR,CSPB(@XX),CLK-D
.MACRO D_SR-XOR-CSPD(XX)                        ;;=  A-XOR-B,SR,CSPD(@XX),CLK-D
.MACRO D_SR-XOR-BSPHI(XX)                       ;;=  A-XOR-B,SR,BSPHI(@XX),CLK-D

.MACRO D_ASPLO[17]-XOR-CSPD(XX)                 ;;=  A-XOR-B,ASPLO(R17),CSPD(@XX),CLK-D
.MACRO D_ASPLO[07]-XOR-BSPHI(XX)                ;;=  A-XOR-B,ASPLO(R07),BSPHI(@XX),CLK-D
.MACRO D_ASPLO[05]-XOR-BSPHI(XX)                ;;=  A-XOR-B,ASPLO(R05),BSPHI(@XX),CLK-D

.MACRO D_SR-XOR-BSPLO[SF]                       ;;=  A-XOR-B,SR,R[SF]-LO-B,CLK-D
.MACRO D_SR-XOR-BSPHI[DF]                       ;;=  A-XOR-B,SR,R[DF]-HI-B,CLK-D

.MACRO D_ASPLO[DF]-XOR-BSPHI[SF]                ;;=  A-XOR-B,R[DF]-LO-A,R[SF]-HI-B,CLK-D
.MACRO D_ASPHI[SF]-XOR-BSPLO[DF]                ;;=  A-XOR-B,R[SF]-HI-A,R[DF]-LO-B,CLK-D

.MACRO D_CSPD[05]-XOR-ASPLO(XX)                 ;;=  A-XOR-B,CSPD(D05),ASPLO(@XX),CLK-D
.MACRO D_CSPD[05]-XOR-ASPHI(XX)                 ;;=  A-XOR-B,CSPD(D05),ASPHI(@XX),CLK-D
.MACRO D_CSPD[06]-XOR-ASPLO(XX)                 ;;=  A-XOR-B,CSPD(D06),ASPLO(@XX),CLK-D
.MACRO D_CSPD[06]-XOR-ASPHI(XX)                 ;;=  A-XOR-B,CSPD(D06),ASPHI(@XX),CLK-D
.MACRO D_CSPD[17]-XOR-ASPHI(XX)                 ;;=  A-XOR-B,CSPD(D17),ASPHI(@XX),CLK-D

.MACRO D_ASPLO[02]-XOR-BSPLO(XX)                ;;=  A-XOR-B,ASPLO(R02),BSPLO(@XX),CLK-D
.MACRO D_ASPLO[03]-XOR-BSPLO(XX)                ;;=  A-XOR-B,ASPLO(R03),BSPLO(@XX),CLK-D
.MACRO D_ASPLO[04]-XOR-BSPLO(XX)                ;;=  A-XOR-B,ASPLO(R04),BSPLO(@XX),CLK-D
.MACRO D_ASPLO[05]-XOR-BSPLO(XX)                ;;=  A-XOR-B,ASPLO(R05),BSPLO(@XX),CLK-D



.TOC    *       D-REGISTER <- D-REGISTER THRU SHIFT-TREE

.MACRO D_D-RIGHT-14             ;;=  A,D-RIGHT-14,CLK-D
.MACRO D_D-RIGHT-13             ;;=  A,D-RIGHT-13,CLK-D
.MACRO D_D-RIGHT-12             ;;=  A,D-RIGHT-12,CLK-D
.MACRO D_D-RIGHT-11             ;;=  A,D-RIGHT-11,CLK-D
.MACRO D_D-RIGHT-10             ;;=  A,D-RIGHT-10,CLK-D
.MACRO D_D-RIGHT-9              ;;=  A,D-RIGHT-9,CLK-D
```

```
2161    .MACRO D_D-RIGHT-8              ;;=  A,D-RIGHT-8,CLK-D
2162    .MACRO D_D-RIGHT-7              ;;=  A,D-RIGHT-7,CLK-D
2163    .MACRO D_D-RIGHT-6              ;;=  A,D-RIGHT-6,CLK-D
2164    .MACRO D_D-RIGHT-5              ;;=  A,D-RIGHT-5,CLK-D
2165    .MACRO D_D-RIGHT-4              ;;=  A,D-RIGHT-4,CLK-D
2166    .MACRO D_D-RIGHT-3              ;;=  A,D-RIGHT-3,CLK-D
2167    .MACRO D_D-RIGHT-2              ;;=  A,D-RIGHT-2,CLK-D
2168    .MACRO D_D-RIGHT-1              ;;=  A,D-RIGHT-1,CLK-D
2169    .MACRO D_D-NO-SHIFT             ;;=  A,D-NO-SHIFT,CLK-D
2170    .MACRO D_D-DIRECT               ;;=  A,D-DIRECT,CLK-D
2171    .MACRO D_D                      ;;=  A,D-DIRECT,CLK-D
2172    .MACRO SAVE-D                   ;;=  A,D-DIRECT,CLK-D
2173    .MACRO D_D-LEFT-1               ;;=  A,D-LEFT-1,CLK-D
2174    .MACRO D_D-SWAB                 ;;=  A,D-SWAB,CLK-D
2175    .MACRO D_D-SWAB-RIGHT-3         ;;=  A,D-SWAB-RIGHT-3,CLK-D
2176    .MACRO D_D-SWAB-LEFT-1          ;;=  A,D-SWAB-LEFT-1,CLK-D
2177    .MACRO D_D-SIGNEXT              ;;=  A,D-SIGNEXT,CLK-D
2178    .MACRO D_D-SIGNEXT-RIGHT-1      ;;=  A,D-SIGNEXT-RIGHT-1,CLK-D
2179    .MACRO D_D-SIGNEXT-LEFT-1       ;;=  A,D-SIGNEXT-LEFT-1,CLK-D
2180    .MACRO D_NO-SHIFT               ;;=  A,NO-SHIFT,CLK-D
2181    .MACRO D_DIRECT                 ;;=  A,DIRECT,CLK-D
2182    .MACRO D_COUNT#0[HI]            ;;=  A,COUNT#0[HI],CLK-D
2183    .MACRO D_COUNT#0[LO]            ;;=  A,COUNT#0[LO],CLK-D
2184
2185
2186
2187    .TOC    *       0 <= WHATEVER'S LEFT, AT P2-T OR P3-T
2188
2189    .MACRO D_NOT-ASPHI(XX)                  ;;=  NOT-A,ASPHI(@XX),CLK-D
2190    .MACRO D_NOT-ASPLO(XX)                  ;;=  NOT-A,ASPLO(@XX),CLK-D
2191    .MACRO D_NOT-CSPB(XX)                   ;;=  A-AND-NOT-B,C177777-A,CSPB(@XX),CLK-D
2192    .MACRO D_NOT-CSPD(XX)                   ;;=  A-AND-NOT-B,C177777-A,CSPD(@XX),CLK-D
2193
2194    .MACRO D_CSPD(XX)                       ;;=  B,CSPD(@XX),CLK-D
2195    .MACRO D_CSPB(XX)                       ;;=  B,CSPB(@XX),CLK-D
2196    .MACRO D_CSPB(16)#D[C]_1                ;;=  A-IOR-B,C000000-A,CSPB(B16),CLK-D,D[C]_CINMUX
2197
2198    .MACRO D_BSPHI(XX)                      ;;=  B,BSPHI(@XX),CLK-D
2199    .MACRO D_BSPLO(XX)                      ;;=  B,BSPLO(@XX),CLK-D
2200    .MACRO D_ASPHI(XX)                      ;;=  A,ASPHI(@XX),CLK-D
2201    .MACRO D_ASPLO(XX)                      ;;=  A,ASPLO(@XX),CLK-D
2202
2203    .MACRO D_ASPLO(DF)                      ;;=  A,R(DF)-LO=A,CLK-D
2204    .MACRO D_ASPHI(DF)                      ;;=  A,R(DF)-HI=A,CLK-D
2205    .MACRO D_BSPLO(DF)                      ;;=  B,R(DF)-LO=B,CLK-D
2206    .MACRO D_BSPHI(DF)                      ;;=  B,R(DF)-HI=B,CLK-D
2207    .MACRO D_ASPLO(SF)                      ;;=  A,R(SF)-LO=A,CLK-D
2208    .MACRO D_ASPHI(SF)                      ;;=  A,R(SF)-HI=A,CLK-D
2209    .MACRO D_BSPLO(SF)                      ;;=  B,R(SF)-LO=B,CLK-D
2210    .MACRO D_BSPHI(SF)                      ;;=  B,R(SF)-HI=B,CLK-D
2211
2212    .MACRO D_CSPD(14)=AND=ASPHI(XX)                 ;;=  A=AND=B,CSPD(B14),ASPHI(@XX),CLK=D
2213    .MACRO D_CSPD(15)=AND=ASPHI(XX)                 ;;=  A=AND=B,CSPD(B15),ASPHI(@XX),CLK=D
2214
```



```
2215    .MACRO SP_ASPHI[17]=AND=007700          ;;=  A=AND=B,ASPHI[R17],CSPB[B17],CLK=SR
2216    .MACRO D_SP=IOR=170000                  ;;=  A=IOR=B,SR,CSPB[B16],CLK=D
2217    .MACRO SR_ASPHI[17]=AND=000077          ;;=  A=AND=B,ASPHI[R17],CSPB[B15],CLK=SR
2218    .MACRO D_SP=IOR=000100                  ;;=  A=IOR=B,SR,CSPB[B14],CLK=D
2219
2220    .MACRO D_ASPLO[17]=AND=CSPD(XX)         ;;=  A=AND=B,ASPLO[R17],CSPD(@XX),CLK=D
2221    .MACRO D_ASPHI[00]=IOR=CSPB(XX)         ;;=  A=IOR=B,ASPHI[R00],CSPD(@XX),CLK=D
2222    .MACRO D_ASPHI[00]=IOR=CSPB(XX)         ;;=  A=IOR=B,ASPHI[R00],CSPB(@XX),CLK=D
2223
2224    .MACRO D_SP                             ;;=  A,SR,CLK=D
2225    .MACRO D_ALL-ONES                       ;;=  A,C177777-A,CLK=D
2226    .MACRO D_D-PLUS-1                       ;;=  A=PLUS=B,D=DIRECT,C000001=B,CLK=D
2227    .MACRO D_JUNK                           ;;=  ZERO,CLK=D
2228    .MACRO D_T40                            ;;=  A=PLUS=B,C000001=A,C000001=B,CLK=D
2229
2230
2231
2232    ;.PAGE=========================================================================================
2233
2234    .TOC    *       SR <= DATA, AT P2 T OR P3 T
2235
2236            ;N.B.:  THE PARTICULAR FUNCTION SELECTED REQUIRES THE RESIDUAL
2237            ;       CONTROL REGISTER ("RES-REG") TO HAVE THE APPROPRIATE
2238            ;       FUNCTION SETUP FOR THE SR OPERATION.
2239            ;
2240            ;       POSSIBLE FUNCTIONS:  LOAD, LEFT, RIGHT, NOP
2241
2242    .MACRO SR_ASPHI(XX)                     ;;=  A,ASPHI(@XX),CLK=SR
2243    .MACRO SR_NOT-ASPHI(XX)                 ;;=  NOT-A,ASPHI(@XX),CLK=SR
2244    .MACRO SR_CSPB(XX)                      ;;=  B,CSPB(@XX),CLK=SR
2245    .MACRO SR_CSPD(XX)                      ;;=  B,CSPD(@XX),CLK=SR
2246    .MACRO SR_NOT-BSPHI(XX)                 ;;=  A=AND=NOT=B,BSPHI[@XX],C177777=A,CLK=SR
2247    .MACRO SR_BSPHI(XX)                     ;;=  B,BSPHI(@XX),CLK=SP
2248    .MACRO SR_SR-PLUS-1                     ;;=  A=PLUS=B,C000001=B,SR,CLK=SR
2249    .MACRO SR_ALL-ONES                      ;;=  A,C177777-A,CLK=SR
2250    .MACRO SR_NOT-CSPB(XX)                  ;;=  A=AND=NOT=B,C177777=A,CSPB(@XX),CLK=SR
2251    .MACRO SR_NOT-CSPD(XX)                  ;;=  A=AND=NOT=B,C177777=A,CSPD(@XX),CLK=SR
2252    .MACRO SR_SR-RIGHT-1                    ;;=  D=DIRECT[SHUX],CLK=SR
2253    .MACRO SR_SR-LEFT-1                     ;;=  CLK=SR
2254    .MACRO SR_JUNK                          ;;=  ZERO,CLK=SP
2255    .MACRO SR_D                             ;;=  A,D=DIRECT,CLK=SR
2256    .MACRO SR_ASPLO(DF)                     ;;=  A,R(DF)=LO=A,CLK=SR
2257    .MACRO SR_ASPHI(DF)                     ;;=  A,R(DF)=HI=A,CLK=SR
2258    .MACRO SR_BSPLO(DF)                     ;;=  B,R(DF)=LO=B,CLK=SR
2259    .MACRO SR_BSPHI(DF)                     ;;=  B,R(DF)=HI=B,CLK=SR
2260    .MACRO SR_ASPLO(SF)                     ;;=  A,R(SF)=LO=A,CLK=SR

2261    .MACRO SP_ASPHI(SF)                     ;;=  A,R(SF)=HI=A,CLK=SR
2262    .MACRO SP_BSPLO(SF)                     ;;=  B,R(SF)=LO=B,CLK=SR
2263    .MACRO SR_BSPHI(SF)                     ;;=  B,R(SF)=HI=B,CLK=SR
2264
2265
2266
2267
2268
```

```
2269    .TOC    *       REG-REG OPERATION MACROS
2270
2271    .MACRO REG_CSPD(XX)             ;;=  CSPD(@XX),LOAD-REA
2272    .MACRO REG_CSPB(XX)             ;;=  CSPB(@XX),LOAD-REA
2273
2274
2275
2276
2277
2278    .TOC    *       BASE MACHINE COUNTER
2279
2280    .MACRO COUNTER_CSPD(XX)         ;;=  LOAD-COUNTER,CSPD(@XX)
2281    .MACRO COUNTER_BSPHI(XX)        ;;=  LOAD-COUNTER,BSPHI(@XX)
2282
2283
2284
2285
2286    .TOC    *       ENABLE ON BUS-A/B ONLY
2287
2288    .MACRO BUS-A_ASPLO[SF]          ;;=  R(SF)-LO-A
2289    .MACRO BUS-A_ASPLO[DF]          ;;=  R(DF)-LO-A
2290    .MACRO BUS-A_ASPHI[SF]          ;;=  R(SF)-HI-A
2291    .MACRO BUS-A_ASPHI[DF]          ;;=  R(DF)-HI-A
2292    .MACRO BUS-A                    ;;=  NULL
2293    .MACRO BUS-A_ASPLO(XX)          ;;=  ASPLO(@XX)
2294    .MACRO BUS-A_ASPHI(XX)          ;;=  ASPHI(@XX)
2295    .MACRO BUS-A_SR                 ;;=  SR
2296    .MACRO BUS-A_FLTPT              ;;=  FLTPT
2297
2298    .MACRO BUS-B_BSPLO[SF]          ;;=  R[SF]-LO-B
2299    .MACRO BUS-B_BSPLO[DF]          ;;=  R[DF]-LO-B
2300    .MACRO BUS-B_BSPHI[SF]          ;;=  R[SP]-HI-B
2301    .MACRO BUS-B_BSPHI[DF]          ;;=  R[DP]-HI-B
2302    .MACRO BUS-B                    ;;=  NULL
2303    .MACRO BUS-B_BSPLO(XX)          ;;=  BSPLO(@XX)
2304    .MACRO BUS-B_BSPHI(XX)          ;;=  BSPHI(@XX)
2305    .MACRO BUS-B_CSPD(XX)           ;;=  CSPD(@XX)
2306    .MACRO BUS-B_CSPB(XX)           ;;=  CSPB(@XX)
2307
2308
2309
2310
2311
2312    .TOC    *       LOADING BA REGISTER
2313    ;LOADED AT P1-7 ONLY, FROM BUS-B<01:00>+BUS-A<15:00> -> BA<17:00>
2314
2315    .MACRO BA_BSPLO(XX)             ;;=  CLK-BA,BSPLO(@XX)
2316    .MACRO BA_BSPHI(XX)             ;;=  CLK-BA,BSPHI(@XX)
2317    .MACRO BA_SR                    ;;=  CLK-BA,SR
2318    .MACRO BA_ASPLO(XX)             ;;=  CLK-BA,ASPLO(@XX)
2319    .MACRO BA_ASPHI(XX)             ;;=  CLK-BA,ASPHI(@XX)
2320
2321
2322
```



```
2323
2324
2325    .TOC    *       D AND SP TOGETHER
2326
2327    .MACRO $R3D_SR-PLUS-CSPD(XX)    ;;=  A-PLUS-B,SR,CSPD(@XX),CLK-D,CLK-SR
2328
2329
2330
2331    ;.PAGE*********************************************************************************************************
2332
2333    .TOC    *       UCON FUNCTIONS
2334
2335
2336
2337    .TOC    *       PROCESSOR UCON FUNCTIONS
2338
2339            ;PREVIOUSLY SET UP (UCON-PROC, SET-UCON-CONTROL, EN "FUNCTION"]
2340    .MACRO IR_EMIT                  ;;=  UCON-OPERATION
2341    .MACRO PS[15-12]_D[15:13]       ;;=  UCON-OPERATION
2342    .MACRO FLAG[8-0]_D[15-8]        ;;=  UCON-OPERATION
2343    .MACRO FPS[7-4]_D[7-4]          ;;=  UCON-OPERATION
2344    .MACRO PS[7-4]_D[7-4]           ;;=  UCON-OPERATION
2345    .MACRO PS[3-0]_D[3-0]           ;;=  UCON-OPERATION
2346    .MACRO PS_D                     ;;=  UCON-OPERATION
2347    .MACRO UBREAK_BUSDIN[11-00]     ;;=  UCON-OPERATION
2348
2349            ;SETUP UCON AND EXECUTE IN 1 MICROWORD!
2350    .MACRO PS[15-12]_D[15:13]-[I]   ;;=  UCON-PROC,SET-UCON-CONTROL,UCON-OPERATION,EN-CLK-PS[15-12]
2351    .MACRO FLAG[8-0]_D[15-8]-[I]    ;;=  UCON-PROC,SET-UCON-CONTROL,UCON-OPERATION,EN-CLK-FLAG[8-0]
2352    .MACRO FPS[7-4]_D[7-4]-[I]      ;;=  UCON-PROC,SET-UCON-CONTROL,UCON-OPERATION,EN-CLK-FPS[7-4]
2353    .MACRO PS[7-4]_D[7-4]-[I]       ;;=  UCON-PROC,SET-UCON-CONTROL,UCON-OPERATION,EN-CLK-PS[7-4]
2354    .MACRO PS[3-0]_D[3-0]-[I]       ;;=  UCON-PROC,SET-UCON-CONTROL,UCON-OPERATION,EN-CLK-PS[3-0]
2355    .MACRO PS_D-[I]                 ;;=  UCON-PROC,SET-UCON-CONTROL,UCON-OPERATION,
2356                                         EN-CLK-PS[15-12],EN-CLK-PS[7-4],EN-CLK-PS[3-0]
2357    .MACRO BUSDIN_CUA-[I]           ;;=  UCON-PROC,SET-UCON-CONTROL,BUSDIN_CUA[14-03]
2358    .MACRO BUSDIN_FLAGS&FPS-[I]     ;;=  UCON-PROC,SET-UCON-CONTROL,BUSDIN_FLAG[8-0]$FPS[7-0]
2359    .MACRO BUSDIN_PS-[I]            ;;=  UCON-PROC,SET-UCON-CONTROL,BUSDIN_PS[15-00]
2360    .MACRO BUSDIN_EMIT-[I]          ;;=  UCON-PROC,SET-UCON-CONTROL,BUSDIN_EMIT[15-00]
2361
2362
2363
2364    .TOC    *       CACHE/KT UCON FUNCTIONS
2365
2366            ;SETUP, EXECUTE IN 1 MICROWORD
2367    .MACRO KT-NO-RELOCATE-[I]               ;;=  UCON-CACHE-KT,SET-UCON-CONTROL,EN-KT-NO-RELOCATE
2368    .MACRO BUSDIN_BUS-INTERNAL-ADDR-[I]     ;;=  UCON-CACHE-KT,SET-UCON-CONTROL,BUSDIN_BUS-INTERNAL-ADDR[15-00]

2369    .MACRO BUSDIN_CPU-INTERNAL-ADDR-[I]     ;;=  UCON-CACHE-KT,SET-UCON-CONTROL,BUSDIN_CPU-INTERNAL-ADDR[15-00]
2370    .MACRO BUSDIN_MMR2-[I]                  ;;=  UCON-CACHE-KT,SET-UCON-CONTROL,BUSDIN_MMR2[15-00]
2371    .MACRO BUSDIN_CACHE-STATUS-[I]          ;;=  UCON-CACHE-KT,SET-UCON-CONTROL,BUSDIN_CACHE-STATUS[15-00]
2372    .MACRO BUSDIN_SLR#CCR-(I)               ;;=  UCON-CACHE-KT,SET-UCON-CONTROL,BUSDIN_KT-SEL,KT-SEL-SLR#CCR
2373    .MACRO BUSDIN_MMR0-[I]                  ;;=  UCON-CACHE-KT,SET-UCON-CONTROL,BUSDIN_KT-SEL,KT-SEL-MMR0
2374    .MACRO BUSDIN_PDR-[I]                   ;;=  UCON-CACHE-KT,SET-UCON-CONTROL,BUSDIN_KT-SEL,KT-SEL-PDR
2375    .MACRO BUSDIN_PAR-[I]                   ;;=  UCON-CACHE-KT,SET-UCON-CONTROL,BUSDIN_KT-SEL,KT-SEL-PAR
2376    .MACRO SLR[15-08]_D[15-08]-[I]          ;;=  UCON-CACHE-KT,SET-UCON-CONTROL,UCON-OPERATION,KT-SEL-SLR#CCR,KT-WRITE-HIGH
```

```
2377    .MACRO CCP[07=02]_D[07=02]=[I]          ;;=  UCON=CACHE=XT,SET=UCON=CONTROL,UCON=OPERATION,XT=SEL=BLKCCR,XT=WRITE=LOW
2378    .MACRO MMRO_D=[I]                       ;;=  UCON=CACHE=XT,SET=UCON=CONTROL,UCON=OPERATION,XT=SEL=MMRO,XT=WRITE
2379    .MACRO MMRO[00]_D[00]=[I]               ;;=  UCON=CACHE=XT,SET=UCON=CONTROL,UCON=OPERATION,XT=SEL=MMRO,XT=WRITE=LOW
2380    .MACRO MMRO[15=01]_D[15=01]=[I]         ;;=  UCON=CACHE=XT,SET=UCON=CONTROL,UCON=OPERATION,XT=SEL=MMRO,XT=WRITE=HIGH
2381    .MACRO PDR_D=[I]                        ;;=  UCON=CACHE=XT,SET=UCON=CONTROL,UCON=OPERATION,XT=SEL=PDR,XT=WRITE
2382    .MACRO PDR[03=01]_D[03=01]=[I]          ;;=  UCON=CACHE=XT,SET=UCON=CONTROL,UCON=OPERATION,XT=SEL=PDR,XT=WRITE=LOW
2383    .MACRO PDR[14=08]_D[14=08]=[I]          ;;=  UCON=CACHE=XT,SET=UCON=CONTROL,UCON=OPERATION,XT=SEL=PDR,XT=WRITE=HIGH
2384    .MACRO PAR_D=[I]                        ;;=  UCON=CACHE=XT,SET=UCON=CONTROL,UCON=OPERATION,XT=SEL=PAR,XT=WRITE
2385    .MACRO PAR[07=00]_D[07=00]=[I]          ;;=  UCON=CACHE=XT,SET=UCON=CONTROL,UCON=OPERATION,XT=SEL=PAR,XT=WRITE=LOW
2386    .MACRO PAR[11=08]_D[11=08]=[I]          ;;=  UCON=CACHE=XT,SET=UCON=CONTROL,UCON=OPERATION,XT=SEL=PAR,XT=WRITE=HIGH
2387
2388
2389
2390    .TOC    *       I=O UCON FUNCTIONS
2391
2392            !N.A.!  SETUP IN 1 MICROWORD
2393    .MACRO BUSDIN_JAM=[I]           ;;=  UCON=I=O,EN=STATUS=MUX,SET=UCON=CONTROL,BUSDIN_JAM[15=00]
2394    .MACRO BUSDIN_SERVICE=[I]       ;;=  UCON=I=O,EN=STATUS=MUX,SET=UCON=CONTROL,BUSDIN_SERVICE[15=00]
2395    .MACRO BUSDIN_PBA=[I]           ;;=  UCON=I=O,EN=STATUS=MUX,SET=UCON=CONTROL,BUSDIN_PBA[15=00]
2396    .MACRO BC=FCN=0=[I]             ;;=  UCON=I=O,SET=UCON=CONTROL,UCON=OPERATION,EN=BC=FCN=0
2397    .MACRO START=DELAY=[I]          ;;=  UCON=I=O,SET=UCON=CONTROL,UCON=OPERATION,EN=START=DELAY
2398    .MACRO CLR=JAM=ERRORS=[I]       ;;=  UCON=I=O,SET=UCON=CONTROL,UCON=OPERATION,EN=CLR=JAM=ERRORS
2399    .MACRO CLR=NPR=TIMEOUT=[I]      ;;=  UCON=I=O,SET=UCON=CONTROL,UCON=OPERATION,EN=CLR=NPR=TIMEOUT
2400    .MACRO CLR=PWR=FAIL=[I]         ;;=  UCON=I=O,SET=UCON=CONTROL,UCON=OPERATION,EN=CLR=PWR=FAIL
2401    .MACRO CLR=YELLOW=ZONE=[I]      ;;=  UCON=I=O,SET=UCON=CONTROL,UCON=OPERATION,EN=CLR=YELLOW=ZONE
2402    .MACRO ALLOW=BG[1]H=[I]         ;;=  UCON=I=O,SET=UCON=CONTROL,UCON=OPERATION,EN=ALLOW=BG[1]H
2403    .MACRO BUS=INIT=UCON=[I]        ;;=  UCON=I=O,SET=UCON=CONTROL,UCON=OPERATION,EN=BUS=INIT=UCON
2404
2405
2406    .TOC    *       DCS UCON FUNCTIONS
2407
2408            !SETUP IN 1 MICROWORD!
2409    .MACRO BUSDIN_INUA=[I]          ;;=  UCON=DCS,SET=UCON=CONTROL,BUSDIN_INUA[11=00]
2410    .MACRO BUSDIN_ERROR=CODE=[I]    ;;=  UCON=DCS,SET=UCON=CONTROL,BUSDIN_ERR!EOP!ERRCOD[11=00)
2411
2412
2413    .TOC    *       CONSOLE UCON FUNCTIONS
2414
2415            !SETS UP AND PERFORMS INDICATED OPERATION IN 1 MICROWORD!
2416    .MACRO CONSOLE=NOP                      ;;=  UCON=I=O,EN=CONSOLE=COMMAND,SET=UCON=CONTROL,UCON=OPERATION,EN=CNSL=NOP
2417    .MACRO CLEAR=CONSOLE=COUNTER            ;;=  UCON=I=O,EN=CONSOLE=COMMAND,SET=UCON=CONTROL,UCON=OPERATION,EN=CLR=COUNTR
2418    .MACRO INCREMENT=CONSOLE=COUNTER        ;;=  UCON=I=O,EN=CONSOLE=COMMAND,SET=UCON=CONTROL,UCON=OPERATION,EN=INCR=COUNTR
2419    .MACRO CLEAR=CONSOLE=SERVICE            ;;=  UCON=I=O,EN=CONSOLE=COMMAND,SET=UCON=CONTROL,UCON=OPERATION,EN=CLR=CNSL=SRVC
2420    .MACRO STROBE=CONSOLE=DISPLAY           ;;=  UCON=I=O,EN=CONSOLE=COMMAND,SET=UCON=CONTROL,UCON=OPERATION,EN=STRB=DISP
2421    .MACRO CLEAR=CONSOLE=LED                ;;=  UCON=I=O,EN=CONSOLE=COMMAND,SET=UCON=CONTROL,UCON=OPERATION,EN=CLR=CNSL
2422    .MACRO SET=CONSOLE=LED                  ;;=  UCON=I=O,EN=CONSOLE=COMMAND,SET=UCON=CONTROL,UCON=OPERATION,EN=SET=CNSL
2423    .MACRO SET=CONSOLE=DP=LEDS              ;;=  UCON=I=O,EN=CONSOLE=COMMAND,SET=UCON=CONTROL,UCON=OPERATION,EN=SET=DP
2424    .MACRO BUSDIN_CONSOLE=[I]               ;;=  UCON=I=O,EN=STATUS=MUX,SET=UCON=CONTROL,BUSDIN_CONSOLE[06=00]
2425
2426
2427
2428    .TOC    *       DBUF UCON FUNCTIONS
2429
2430            !PREVIOUSLY SETUP UCON=I=O, EN LOAD DBUF
```



```
2431    .MACRO DBUF_D                   ;;=  UCON=OPERATION
2432
2433            !SETUP AND EXECUTE IN 1 MICROWORD!
2434    .MACRO DBUF_D=[I]               ;;=  UCON=I=O,SET=UCON=CONTROL,UCON=OPERATION,EN=LOAD=DBUF[15=00]
2435
2436
2437    .TOC    *       MULTIPLE UCON FUNCTIONS
2438
2439            !THESE ARE FUNCTIONS OF MORE THAN 1 UCON ENABLED SIMULTANEOUSLY
2440
2441            !PREVIOUSLY SETUP!
2442    .MACRO IR_DBUF                  ;;=  UCON=OPERATION
2443
2444            !SETUP AND EXECUTE IN 1 MICROWORD!
2445    .MACRO IR_DBUF=[I]              ;;=  UCON=PROC,UCON=I=O,SET=UCON=CONTROL,
2446                                            UCON=OPERATION,EN=CLK=IR[15=00],BUSDIN_DBUF[15=00]
2447
2448
2449
2450    !.PAGE=========================================================================================

2451
2452    .TOC    *       SPECIFIC MACROS FOR PREFETCH/OVERLAP/SP-INHIBIT TESTS
2453
2454    .MACRO CSPD[17]_020010          ;;=  EMIT/020010,CSPD[17]_EMIT
2455    .MACRO A=BSPLO[OVERLAP]_D       ;;=  ASPLO(R10),WR(AB,L,A)
2456    .MACRO A=BSPHI[PATTERN]_D       ;;=  ASPHI(R17),WR(AB,H,A)
2457    .MACRO A=BSPHI[PREFETCH]_D      ;;=  ASPHI(R16),WR(AB,H,A)
2458    .MACRO ASPLO[OVERLAP]_D         ;;=  ASPLO(R10),WR(A,L,A)
2459    .MACRO BSPLO[OVERLAP]_D         ;;=  BSPLO(R10),WR(B,L,B)
2460    .MACRO ASPHI[PREFETCH]_D        ;;=  ASPHI(R10),WR(A,H,A)
2461
2462    .MACRO D_ASPLO[OVERLAP]=PLUS=1          ;;=  A=PLUS=B,ASPLO(R10),C000001=B,CLK=D
2463    .MACRO D_BSPLO[OVERLAP]=PLUS=1          ;;=  A=PLUS=B,BSPLO(R10),C000001=A,CLK=D
2464    .MACRO D_ASPHI[PREFETCH]=PLUS=1         ;;=  A=PLUS=B,ASPHI(R10),C000001=B,CLK=D
2465    .MACRO D_ASPHI[PATTERN]=PLUS=020010=PLUS=1      ;;=  A=PLUS=B=PLUS=1,ASPHI(R17),CSPB(B17),CLK=D
2466    .MACRO D_ASPHI[PATTERN]=AND=NOT=020010          ;;=  A=AND=NOT=B,ASPHI(R17),CSPB(B17),CLK=D
2467
2468    .MACRO D_BSPHI[PREFETCH]                ;;=  B,BSPHI(R10),CLK=D
2469    .MACRO D_ASPLO[OVERLAP]=MINUS=CSPB[EXPEC]       ;;=  A=MINUS=B,ASPLO(R10),CSPB(B17),CLK=D
2470    .MACRO D_ASPHI[PPEFETCH]=MINUS=CSPB[EXPEC]      ;;=  A=MINUS=B,ASPHI(R10),CSPB(B17),CLK=D
2471    .MACRO D_ASPLO[OVERLAP]=MINUS=BSPLO[OVERLAP]    ;;=  A=MINUS=B,ASPLO(R10),BSPLO(R10),CLK=D
2472
2473    .MACRO CSPD[EXPEC]_EMIT                 ;;=  CSPD[17]_EMIT
2474
2475
2476

2477    !.PAGE=========================================================================================

2478
2479    .TOC    *       SPECIFIC MACROS FOR BYTE/BYTE CONSTANT/O=ZERO TESTS
2480
2481    .MACRO D_ASPLO[NONZERO]=PLUS=1          ;;=  A=PLUS=B,ASPLO(R11),C000001=B,CLK=D
2482    .MACRO D_BSPLO[ZERO]=PLUS=1             ;;=  A=PLUS=B,C000001=A,BSPLO(R11),CLK=D
2483    .MACRO D_BSPLO[]=A=DATA)=PLUS=2         ;;=  A=PLUS=A=PLUS=1,C000001=A,BSPLO(R10),CLK=D
2484    .MACRO D_ASPHI[WORD]=PLUS=1             ;;=  A=PLUS=B,C000001=A,BSPHI(R10),CLK=D
```

```
2485    .MACRO D_BSPLO[IR=DATA]                 ;;=  B,BSPLO(R10),CLK=D
2486    .MACRO D_BSPLO[ZERO]                    ;;=  B,BSPLO(R11),CLK=D
2487    .MACRO D_BR=MINUS=BSPHI[WORD]           ;;=  A=MINUS=B,BR,BSPHI(R10),CLK=D,P3=T
2488    .MACRO D_ASPLO[DNOWZERO]=MINUS=CSPD[17]       ;;=  A=MINUS=B,ASPLO(R11),CSPD(D17),CLK=D,P3=T
2489    .MACRO D_ASPHI[BYTE-FIRST]=MINUS=CSPD[17]     ;;=  A=MINUS=B,ASPHI(R10),CSPD(D17),CLK=D,P3=T
2490    .MACRO D_ASPLO[BYTE-SECOND]=MINUS=CSPD[17]    ;;=  A=MINUS=B,ASPLO(R10),CSPD(D17),CLK=D,P3=T
2491    .MACRO D_ASPHI[BYTE-FIRST]=PLUS=CSP[1=0]      ;;=  A=PLUS=B,ASPHI(R10),CSPD(D13),CLK=D
2492    .MACRO D_ASPLO[BYTE=SECOND]=PLUS=CSP[1=0]     ;;=  A=PLUS=B,ASPLO(R10),CSPD(D13),CLK=D
2493
2494
2495
2496    !.PAGE*************************************************************************************************
2497
2498    .TOC    *       SUBROUTINE CALL MACROS
2499
2500    .MACRO CALL[DISPLAY]                    ;;=  GOTO=PAGE(7),J/DISPLAY
2501
2502    .MACRO CALL[DINTOIR]                    ;;=  GOTO=PAGE(7),J/DINTOIR
2503    .MACRO CALL[DINTOIR=5]                  ;;=  GOTO=PAGE(7),J/DINTOIR5
2504    .MACRO CALL[SRINTOIR]                   ;;=  GOTO=PAGE(7),J/SRINTOIR
2505    .MACRO CALL[SRINTOIR=5]                 ;;=  GOTO=PAGE(7),J/SRINTOIR5
2506
2507    .MACRO CALL[FLAGFP8TOD]                 ;;=  GOTO=PAGE(7),J/FLAGFP8TOD
2508    .MACRO CALL[P8TOD]                      ;;=  GOTO=PAGE(7),J/P8TOD
2509    .MACRO CALL[CUATOD]                     ;;=  GOTO=PAGE(7),J/CUATOD
2510    .MACRO CALL[CLRJAMTOD]                  ;;=  GOTO=PAGE(7),J/CLRJAMTOD
2511    .MACRO CALL[ODDJAMTOD]                  ;;=  GOTO=PAGE(7),J/ODDJAMTOD
2512    .MACRO CALL[JAMTOD]                     ;;=  GOTO=PAGE(7),J/JAMTOD
2513    .MACRO CALL[CLRSERVICETOD]              ;;=  GOTO=PAGE(7),J/CLRSERVICETOD
2514    .MACRO CALL[DATISERVICETOD]             ;;=  GOTO=PAGE(7),J/DATISERVICETOD
2515    .MACRO CALL[DATOSERVICETOD]             ;;=  GOTO=PAGE(7),J/DATOSERVICETOD
2516    .MACRO CALL[CJESERVICETOD]              ;;=  GOTO=PAGE(7),J/CJESERVICETOD
2517    .MACRO CALL[SERVICETOD]                 ;;=  GOTO=PAGE(7),J/SERVICETOD
2518    .MACRO CALL[P8ATOD]                     ;;=  GOTO=PAGE(7),J/P8ATOD
2519    .MACRO CALL[P8SEQLOD]                   ;;=  GOTO=PAGE(7),J/P8SEQLOD
2520    .MACRO CALL[FLAGFP8SEQLOD]              ;;=  GOTO=PAGE(7),J/FLAGFP8SEQLOD
2521    .MACRO CALL[FLAGLOD]                    ;;=  GOTO=PAGE(7),J/FLAGFP801
2522    .MACRO CALL[GETPROCDAT]                 ;;=  GOTO=PAGE(7),J/GETPROCDAT
2523
2524    .MACRO CALL[CLEAR=I=0=A]                ;;=  GOTO=PAGE(7),J/CLEAR=I=0=A
2525    .MACRO CALL[CLEAR=I=0=B]                ;;=  GOTO=PAGE(7),J/CLEAR=I=0=B
2526
2527    .MACRO CALL[BUSDINXOR[25252]            ;;=  GOTO=PAGE(7),J/BDX12
2528    .MACRO CALL[BUSDINXOR0525252]           ;;=  GOTO=PAGE(7),J/BDX05
2529    .MACRO CALL[C8P17XOR[25252]             ;;=  GOTO=PAGE(7),J/C17X12
2530    .MACRO CALL[C8P17XOR0525251             ;;=  GOTO=PAGE(7),J/C17X05
2531
2532    .MACRO CALL[D15=12]                     ;;=  GOTO=PAGE(7),J/D[15-12]
2533    .MACRO CALL[D11=06]                     ;;=  GOTO=PAGE(7),J/D[11-06]
2534    .MACRO CALL[D05=00]                     ;;=  GOTO=PAGE(7),J/D[05-00]
2535    .MACRO CALL[DZERO]                      ;;=  GOTO=PAGE(7),J/DZERO
2536
2537    .MACRO CALL[ZERO8F04DF02]               ;;=  GOTO=PAGE(7),J/ZERO8F04DF02
2538    .MACRO CALL[ZERO8F02DF04]               ;;=  GOTO=PAGE(7),J/ZERO8F02DF04
```



```
2539    .MACRO CALL[ZERO8FDF]                   ;;=  GOTO=PAGE(7),J/ZERO8FDF
2540    .MACRO CALL[ZERODF]                     ;;=  GOTO=PAGE(7),J/ZERODF
2541
2542    .MACRO CALL[8FDFTO8R]                   ;;=  GOTO=PAGE(5),J/8FDFTO8R
2543
2544    .MACRO CALL[ALUCARRY1]                  ;;=  GOTO=PAGE(7),J/ALUCARRY1
2545    .MACRO CALL[ALUCARRY2]                  ;;=  GOTO=PAGE(7),J/ALUCARRY2
2546
2547    .MACRO CALL[LOADFP8CC]                  ;;=  GOTO=PAGE(7),J/LOADFP8CC
2548    .MACRO XFR=TO=RN[LOADNZW4]              ;;=  GOTO=PAGE(4),J/LOADNZW4
2549
2550    .MACRO CALL[SETUPP8CC=DC]               ;;=  GOTO=PAGE(7),J/SETUPP8CC=DC
2551    .MACRO CALL[P8CCTO8R]=0]                ;;=  GOTO=PAGE(7),J/P8CCTO8R]=0
2552
2553    .MACRO CALL[C8P16XORSRTOIR=5]           ;;=  GOTO=PAGE(7),J/C8P16XOR8SRTOIR5
2554    .MACRO CALL[C8P16XORFLTTOIR=5]          ;;=  GOTO=PAGE(7),J/C8P16XORFLTTOIR5
2555
2556    .MACRO CALL[MF88=TEST]                  ;;=  GOTO=PAGE(6),J/MF8801
2557
2558    .MACRO CALL[KTSRCOST]                   ;;=  GOTO=PAGE(7),J/KTSRCD8T01
2559    .MACRO CALL[KTD8T8P]                    ;;=  GOTO=PAGE(7),J/KTSRCD8T08
2560    .MACRO CALL[KTSRC8P]                    ;;=  GOTO=PAGE(7),J/KTSRCD8T07
2561
2562    .MACRO CALL[COUNT=TEST]                 ;;=  GOTO=PAGE(4),J/COUNTER01
2563
2564
2565
2566    !.PAGE*************************************************************************************************
2567
2568    .TOC    *       JAM UPP LOG MACROS
2569
2570          !MACROS CONCERNED WITH CSP LOG AFTER UNEXPECTED JAMUPP
2571          !MACROS REQUIRE APPROPRIATE REGISTER ENABLED ON BUSDIN
2572
2573    .MACRO CSPD[00]_LOG=CHA                 ;;=  CSPD(D00),WR=CSP
2574    .MACRO CSPD[01]_LOG=SERVICE             ;;=  CSPD(D01),WR=CSP
2575    .MACRO CSPD[02]_LOG=JAM                 ;;=  CSPD(D02),WR=CSP
2576
2577
2578
2579    !*****  END OF MACRO DEFINITIONS  *****
2580
2581
2582
2583    !.PAGE*************************************************************************************************
2584

2585    .TOC    * - - -  MICRODIAGNOSTIC CODE - - - - - - - - - - - - - - - - - -
2586
2587    .CODE

2588
2589
2590
2591
2592
```

```
;.PAGE=========================================================================================

;*** VERSION /V10140/ ***

;***** MICRODIAGNOSTIC INITIAL STARTUP LOCATION *****

.TOC    *   TEST001-007: MUA SEQUENCING

;*****************************************************************************************
;*
;*      TESTS:  001 - 007                                    UWORDS: 010 + 000
;*
;*      FUNCTIONS:       TESTS 001 - 007 TEST THE MUA SEQUENCING LOGIC,
;*                               PATTERNS ARE RUN THRU THE MUA LOGIC ESTABLISHING
;*                               THAT ALL BITS CAN BE SET AND CLEARED, AND THAT THE
;*                               STATE (SET, CLEARED) OF NO ONE BIT AFFECTS THE
;*                               ABILITY TO SET/CLEAR ANY OTHER BIT.  THE PAGE
;*                               CHANGING FUNCTIONS OF BUTA(SUBR=A) AND BUTA(SUBR=B)
;*                               ARE ALSO TESTED FOR PAGES 4, 5, 6, & 7.  NOTE
;*                               THAT THE RETURN ADDRESS SIMULTANEOUSLY LOADED
;*                               IS NOT CHECKED FOR VALIDITY AT THIS POINT.
;*
;*      NOTES:  TEST(N) DOES THE SETUP FOR TEST(N+1).  THE ACTUAL TEST CONSISTS
;*              OF BEING IN THE RIGHT PLACE (MICROWORD) AT THE RIGHT TIME.
;*
;*****************************************************************************************

; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

;*** TEST 001 ***
;TEST   MUA LOGIC   WITH PATTERN "100 000 000 000"
4000:
TEST001:
        P0,     LOAD-ENUA(6252),                        ;LOAD ENUA WITH ADDR(NEXT UWORD)
                LOAD-ERROR(TEST001),                    ;ERROR DIRECTORY KEY
                DCS-CTR(C1,),                           ;COMPARE AT NEXT UWORD
        NEXT,   PAGE(2), BUTA(SUBR=B),                  ;CHANGING TO PAGE 2 (ACTUALLY 6), VIA SUBR=B
                J/TEST002                               ;NOTE OVERLAP:  NEXTPAGE<2:0>=ENUA<2:0>
(4000)  DCS[1,00,1,0,0,0]  BM[1110,,00,11,,00,10,,101,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,101,010]
                                                        ;NOTE ALSO DCS MODULE FORCES MUA<11>=1, SO
                                                        ;PAGES 2/6 ARE EQUIVALENT REFERENCES.
```



```
; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

;*** TEST 002 ***
;TEST   MUA LOGIC   WITH PATTERN "010 010 101 010"
6252:
TEST002:
        P0,     LOAD-ENUA(6631),                        ;LOAD ENUA WITH ADDR(NEXT UWORD)
                LOAD-ERROR(TEST002),                    ;ERROR DIRECTORY KEY
                DCS-CTR(C1,),                           ;COMPARE AT NEXT UWORD
                BUMP-VERIFY,                            ;COUNT
        NEXT,   J/TEST003                               ;
(6252)  DCS[1,00,1,0,0,1]  BM[1110,,00,11,,01,10,,011,,001,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,110,011,001]

; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

;*** TEST 003 ***
;TEST   MUA LOGIC   WITH PATTERN "010 110 011 001"
6631:
TEST003:
        P0,     LOAD-ENUA(5525),                        ;LOAD ENUA WITH ADDR(NEXT UWORD)
                LOAD-ERROR(TEST003),                    ;ERROR DIRECTORY KEY
                DCS-CTR(C1,),                           ;COMPARE AT NEXT UWORD
        NEXT,   PAGE(5), BUTA(SUBR=A),                  ;CHANGING TO PAGE 5, VIA SUBR=A
                J/TEST004                               ;NOTE OVERLAP:  NEXTPAGE<2:0>=ENUA<2:0>
(6631)  DCS[1,00,1,0,0,0]  BM[1110,,00,10,,11,01,,010,,101,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,101,,,101,010,101]

; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

;*** TEST 004 ***
;TEST   MUA LOGIC   WITH PATTERN "101 101 010 101"
5525:
TEST004:
        P0,     LOAD-ENUA(5146),                        ;LOAD ENUA WITH ADDR(NEXT UWORD)
                LOAD-ERROR(TEST004),                    ;ERROR DIRECTORY KEY
```

```
                DCS-CTR(C1,),                                   ;COMPARE AT NEXT UWORD
                BUMP-VERIFY,                                    ;COUNT
        NEXT,   J/TEST005                                       ;
  (5535)  DCS[1,00,1,0,0,1]  BM[1110,,00,10,,10,01,,100,,110,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,100,110]



   ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

   ;*** TEST 005 ***
   ;TEST   NUA LOGIC   WITH PATTERN "101 001 100 110"
   5146:
   TEST005:
        PO,     LOAD-ENUA(4474),                                ;LOAD ENUA WITH ADDR(NEXT UWORD)
                LOAD-ERROR(TEST005),                            ;ERROR DIRECTORY KEY
                DCS-CTR(C1,),                                   ;COMPARE AT NEXT UWORD
        NEXT,   PAGE(4), BUTA(SUBR-B),                          ;CHANGING TO PAGE 4, VIA SUBR-B
                J/TEST006                                       ;NOTE OVERLAP:  NEXTPAGE<2:0>=ENUA<2:0>
  (5146)  DCS[1,00,1,0,0,0]  BM[1110,,00,10,,01,00,,111,,100,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,100,111,100]



   ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

   ;*** TEST 006 ***
   ;TEST   NUA LOGIC   WITH PATTERN "100 100 111 100"
   4474:
   TEST006:
        PO,     LOAD-ENUA(4377),                                ;LOAD ENUA WITH ADDR(NEXT UWORD)
                LOAD-ERROR(TEST006),                            ;ERROR DIRECTORY KEY
                DCS-CTR(C1,),                                   ;COMPARE AT NEXT UWORD
                BUMP-VERIFY,                                    ;COUNT
        NEXT,   J/TEST007                                       ;
  (4474)  DCS[1,00,1,0,0,1]  BM[1110,,00,10,,00,11,,111,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,011,111,111]



   ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
```



```
   ;*** TEST 007 ***
   ;TEST   NUA LOGIC   WITH PATTERN "100 011 111 111"
   4377:
   TEST007:
        PO,     LOAD-ENUA(7303),                                ;LOAD ENUA WITH ADDR(NEXT UWORD)
                LOAD-ERROR(TEST007),                            ;ERROR DIRECTORY KEY
                DCS-CTR(C1,),                                   ;COMPARE AT NEXT UWORD
        NEXT,   PAGE(3), BUTA(SUBR-A),                          ;CHANGING TO PAGE 3 (ACTUALLY 7), VIA SUBR-A
                J/NEXT007                                       ;NOTE OVERLAP:  NEXTPAGE<2:0>=ENUA<2:0>
  (4377)  DCS[1,00,1,0,0,0]  BM[1110,,00,11,,10,11,,000,,011,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,101,,,011,000,011]
                                                                ;NOTE ALSO DCS MODULE FORCES NUA<11>=1, SO
                                                                ;PAGES 3/7 ARE EQUIVALENT REFERENCES,



   ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

   ;*** TEST 007-1/2 ***
   ;TEST   NUA LOGIC   WITH PATTERN "011 011 000 011"
   7303:
   NEXT007:
        NEXT,   PAGE(4), BUTA(SUBR-B),                          ;CHANGING TO PAGE 4, VIA SUBR B
                J/TEST010                                       ;OFF TO NEXT TEST
  (7303)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,100,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,111,111,101]



   ;.PAGE=========================================================================================

   .TOC    *   TEST010-011:  MICROSUBROUTINE OPERATION

   ;*************************************************************************************
   ;*                                                                                   *
   ;*      TESTS:  010 - 011                               UWORDS: 007 + 002            *
   ;*                                                                                   *
   ;*      FUNCTIONS:      THESE TWO TESTS DETERMINE THAT THE RETURN REGISTER AND       *
   ;*                      ASSOCIATED DECODE, MUX, AND ENABLING LOGIC IS ABLE TO        *
   ;*                      LOAD THE 12-BIT NUA RETURN REGISTER FROM THE EMIT            *
   ;*                      FIELD, AND THEN ENTER THE REGISTER CONTENTS ONTO THE         *
   ;*                      UWORD ADDRESS BUS (WHEN ENABLED BY A BUTA[RETURN]) IN        *
   ;*                      TIME TO FETCH THE NEXT MICROWORD.  TWO ALTERNATING           *
   ;*                      BIT TESTS ARE USED TO CHECK THAT EACH BIT CAN BE SET         *
   ;*                      AND CLEARED, INDEPENDENT OF ADJACENT BITS.                   *

   ;*                                                                                   *
   ;*************************************************************************************
```

```
        ;- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        ;*** TEST 010 ***
        ;TEST RETURN LOGIC WITH PATTERN "1010 1010 0101" (5245)
        4775:
        TEST010:
                P0,     LOAD-ENUA(NEXT010),                ;LOAD ENUA WITH EXPECTED RETURN ADDRESS
                        LOAD-ERROR(TEST010),               ;ERROR DIRECTORY KEY
                        DCS-CTR(C4.),                      ;COMPARE ENUA:THUA IN 4. MICROWORDS
                NEXT,   J/LOAD010                          ;GO LOAD
  (4775)  DCS[1,00,1,0,0,0]  BM[1011,,00,10,,10,10,,100,,101,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,000,000]

        4100:  ;(FREE)
        LOAD010:
                SETUP,  RETURN/NEXT010,                    ;SET RETURN ADDRESS FROM EMIT
                NEXT,   PAGE(7),                           ;"SUBR" IS ON PAGE 7
                        BUTA(SUBR-B),                      ;WITH B VERSION
                        J/SUBR010                          ;"SUBR" DISP. ON PAGE
  (4100)  DCS[0,00,0,0,0,0]  BM[0101,,00,01,,01,00,,101,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,000,000,000]

        7000:  ;(FREE)
        SUBR010:
                SETUP,  RETURN/ERROR010,                   ;NOISE BITS IN EMIT-RETURN FIELD
                P3,     BUTA(CLR-FLAG-RES-UCON),           ;NOISE IN UPF FIELD
                NEXT,   J/ZTARGET777                       ;BEFORE DO BUTA[RETURN]
  (7000)  DCS[0,00,0,0,0,0]  BM[0110,,00,10,,11,00,,011,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,010,,,111,111,111]

        ;   REMAINDER OF SUBROUTINE IS AT ZTARGET777 [LOCATION 7777[8]], WHICH IMMEDIATELY DOES A BUTA[RETURN]
        ;   TO THE ADDRESS IN THE RETURN REGISTER, WHICH SHOULD BE THE VALUE LOADED IN WORD
        ;   LOAD010.  THE RETURN REGISTER WAS SET TO POINT TO THE START OF THE NEXT TEST
        ;   (NEXT010), AT WHICH POINT THE ENUA:THUA COMPARE IS SET TO TAKE PLACE.

        6543:
        ERROR010:
                P0,     BUMP-VERIFY,                       ;COUNT
                NEXT,   PAGE(4),                           ;IF WE END UP HERE, THE NOISE BITS
                        J/TEST010                          ;WERE LOADED INTO THE RETURN REGISTER INSTEAD
  (6543)  DCS[0,00,0,0,0,1]  BM[0000,,00,00,,00,00,,000,,100,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,111,111,101]

        5245:
        NEXT010:
                P0,     BUMP-VERIFY,                       ;COUNT
                NEXT,   J/TEST011                          ;RETURNED OK, COMPARE ENUA:THUA DONE HERE
  (5245)  DCS[0,00,0,0,0,1]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,111,111,111]
```



```
        ;- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        ;*** TEST 011 ***
        ;TEST RETURN LOGIC WITH PATTERN "1101 0101 1010" (6532)
        5777:
        TEST011:
                P0,     LOAD-ENUA(INITIALIZE01),           ;LOAD ENUA WITH EXPECTED RETURN ADDRESS
                        LOAD-ERROR(TEST011),               ;ERROR DIRECTORY KEY
                        DCS-CTR(C3.),                      ;COMPARE ENUA:THUA IN 3. MICROWORDS
                NEXT,   J/LOAD011                          ;GO LOAD
  (5777)  DCS[1,00,1,0,0,0]  BM[1100,,00,11,,01,01,,011,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,000,001]

        5001:  ;(FREE)
        LOAD011:
                SETUP,  RETURN/INITIALIZE01,               ;SET RETURN ADDRESS IN EMIT
                NEXT,   PAGE(7),                           ;"SUBR" IS ON PAGE 7
                        BUTA(SUBR-B),                      ;VIA B VERSION
                        J/RESETUCONP                       ;"SUBR" DISP. ON PAGE
  (5001)  DCS[0,00,0,0,0,0]  BM[0110,,00,10,,10,11,,010,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,111,001]

        ;   "ONE-WORD SUBROUTINE" IS AT LABEL "RESETUCONP", WHICH IMMEDIATELY DOES A BUTA[RETURN]
        ;   TO THE ADDRESS IN THE RETURN REGISTER, WHICH SHOULD BE THE VALUE LOADED AT SYMBOLIC LABEL
        ;   "LOAD011".  THE RETURN REGISTER WAS SET TO POINT TO THE INITIALIZATION CODE ON THE NEXT PAGE:
        ;   "INITIALIZE01", AT WHICH POINT THE ENUA:THUA COMPARE IS SET TO TAKE PLACE.
        ;   THIS SUBROUTINE ALSO "ATTEMPTS" TO SET THE UCON CONTROL TO:  "PROC", "BUSDIN_EMIT", AND
        ;   "EN-CLK-IR", FOR USE IN THE FOLLOWING INITIALIZATION ROUTINE.



        ; + + + + + + + + + + + + + + + + + + + + + + + + + + + + + + + + + + + + + + + + +

        ;       ***  BUTA(RETURN) ERROR LOOP WORDS:
        ;
        ;       IF AT ANY TIME, A "BUTA(RETURN)" FAILS, ONE OF THE FOLLOWING MICROWORDS WILL BE
        ;       CONTINUOUSLY LOOPED UPON.  THIS IS DUE TO THE FACT THAT FOR EACH MICROWORD THAT
        ;       CONTAINS THIS "BUTA(RETURN)", THE UPF FIELD (WHICH WOULD NORMALLY NOT BE USED IN THIS
        ;       MICROWORD EXECUTION), CONTAINS THE VALUE "J/BUTERROR#", WHERE NUMBER (#) IS THE
        ;       CURRENT MICROWORD'S PAGE NUMBER (4-7).

        4376:
        BUTERROR4:
                P0,     DCS-CTR(C0.),                      ;FORCE AN ERROR, ERROR-CODE=LAST LOADED
                NEXT,   J/BUTERROR4                        ;HANG UP
  (4376)  DCS[0,00,1,0,0,0]  BM[1111,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,011,111,110]


        5376:
        BUTERROR5:
                P0,     DCS-CTR(C0.),                      ;FORCE AN ERROR, ERROR-CODE=LAST LOADED
                NEXT,   J/BUTERROR5                        ;HANG UP
  (5376)  DCS[0,00,1,0,0,0]  BM[1111,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,011,111,110]
```

```
2891:    6376:
2892:    BUTERROR6:
2893:            P0,     DCS-CTR[C0,],                   ;FORCE AN ERROR, ERROR-CODE=LAST LOADED
2894:            NEXT,   J/BUTERROR6                     ;HANG UP
     (6376)  DCS[0,00,1,0,0,0]  BM[1111,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,011,111,110]
2895:
2896:
2897:    7376:
2898:    BUTERROR7:
2899:            P0,     ;DCS-CTR[C0,],                  ;FORCE AN ERROR, ERROR-CODE=LAST LOADED
2900:            NEXT,   J/BUTERROR7                     ;HANG UP
     (7376)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,011,111,110]
2901:
2902:
2903:
2904:    |.PAGE**********************************************************************************************************
2905:
2906:    .TOC    *    INIT REGISTERS, CONSOLE DEFAULT ERROR DISPLAY, REVISION CODE
2907:
2908:
2909:    |**********************************************************************************************
2910:    |*
2911:    |*      TESTS:  INITIALIZATION                          OWORDS:  011 + 011
2912:    |*
2913:    |*      FUNCTIONS:
2914:    |*
2915:    |*      TRY TO PUT AN OCTAL (000000) IN THE CONSOLE DISPLAY AS AN ERROR INDICATOR
2916:    |*
2917:    |*      PUT REVISION CODE, BIT15 CLEAR, IN GPR R5
2918:    |*
2919:    |*      SET FLAGS, FPS, PS, UBREAK REGISTERS TO ALL ZEROS TO DISABLE AS MUCH
2920:    |*      AS POSSIBLE ANY SPURIOUS NOT FLOATING POINT STARTUPS, UBREAKS, ETC.
2921:    |*
2922:    |*      TURN OFF CACHE (SET BOTH FORCE MISS BITS), AND TURN OFF KT (MEMORY
2923:    |*      MANAGEMENT (BY CLEARING ENABLE BIT).
2924:    |*
2925:    |**********************************************************************************************
2926:
2927:
2928:                    ;RETURNED ON, COMPARE OF ENUA:TMUA
2929:                    ;FOR PREV TEST DONE HERE
2930:    6532:
2931:    INITIALIZE01:
2932:            SETUP,  RETURN/INITIALIZE03,            ;GO TO SUBR THAT PUTS REVISION NUMBER,
2933:            NEXT,   GOTO-PAGE(7),                   ;  WITH R=<15>=(0), INTO R,M, GPR "R5"
2934:                    J/INSERTREVNO                   ;
     (6532)  DCS[0,00,0,0,0,0]  BM[0111,,00,00,,00,00,,001,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,001,110]
2935:
2936:    7001:  ;(FREE)
2937:    INITIALIZE03:
2938:            P1,     CSPD[17]_INIT, EMIT/000014,     ;EMITCON TO DISABLE CACHE, KT
2939:            NEXT,   J/INITIALIZE04                  ;
     (7001)  DCS[0,00,0,0,0,0]  BM[0000,,10,00,,00,00,,001,,100,,,0,0,0,,0,,0,,,0,0000,,,1,,0000,0,,,11,000,,,000,000,011]
```



```
2940:
2941:    7003:  ;(FREE)
2942:    INITIALIZE04:
2943:            P2-T,   D_CSPD(D17), 0[C]_0,            ;SET ABOVE CONSTANT INTO D
2944:            P2,     RES_CSPD(D17),                  ;BIT@<14:11>=0/00/0 FOR SR=LOAD, GUARD=DIS
2945:            NEXT,   J/INITIALIZE05                  ;
     (7003)  DCS[0,00,0,0,0,0]  BM[1010,,10,00,,00,00,,000,,000,,,0,1,0,,0,,0,,,0,0000,,,0,,1000,1,,,11,000,,,000,000,100]
2946:
2947:    7004:  ;(FREE)
2948:    INITIALIZE05:
2949:            P2-T,   CCR[07-02]_D[07-02]=[I],        ;LOAD CCR, DISABLING CACHE
2950:            NEXT,   J/INITIALIZE06                  ;  BIT@<3:2> SET FORCE MISS
     (7004)  DCS[0,00,0,0,0,0]  BM[0001,,00,01,,00,00,,000,,000,,,0,0,0,,0,,0,,,1,1011,,,0,,0000,0,,,11,000,,,000,000,101]
2951:
2952:    7005:  ;(FREE)
2953:    INITIALIZE06:
2954:            P2-T,   MMR0_0=[I],                     ;LOAD MMR0, DISABLING KT;1
2955:            NEXT,   J/INITIALIZE07                  ;  BIT@<00> CLEAR DISABLES KT;1
     (7005)  DCS[0,00,0,0,0,0]  BM[0001,,00,11,,01,00,,000,,000,,,0,0,0,,0,,0,,,1,1011,,,0,,0000,0,,,11,000,,,000,000,110]
2956:
2957:    7006:  ;(FREE)
2958:    INITIALIZE07:
2959:            P2-T,   0_ZERO,                         ;DEFAULT TO ALL ZEROS FOR ERROR
2960:                    SP_ZERO,                        ;ZERO SP FOR JAMUPP IS ERROR
2961:            NEXT,   GOTO-PAGE(4),                   ;IFER
2962:                    J/INITIALIZE10                  ;
     (7006)  DCS[0,00,0,0,0,0]  BM[0011,,00,00,,00,00,,000,,100,,,0,1,1,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,001,000,001]
2963:
2964:    4101:  ;(FREE)
2965:    INITIALIZE10:
2966:            SELECT, UCON-PROC,                      ;ENABLE CLOCKING THE FOLLOWING
2967:            ENABLE, EN-CLK-PS[15-12], EN-CLK-PS[7-4],       ;  PROCESSOR REGISTERS:
2968:                    EN-CLK-PS[3-0], EN-CLK-FLAG[8-0],       ;
2969:                    EN-CLK-FPS[7-4], EN-CLK-UBREAK[11-00],  ;
2970:                    BUSDIN_EMIT[15-00],             ;FOR UBREAK CONSTANT
2971:            P1,     SET-UCON-CONTROL,               ;WRITE CONTROLS
2972:                    BUMP=VERIFY,                    ;COUNT
2973:            P1,     RUTA(CUA-TRACK),                ;RESET CUA TRACKING
2974:            NEXT,   J/INITIALIZE11                  ;
     (4101)  DCS[0,00,0,0,0,1]  BM[1000,,01,00,,00,01,,110,,011,,,0,0,0,,0,,0,,,1,1001,,,0,,0000,0,,,11,001,,,001,000,010]
2975:
2976:    4102:  ;(FREE)
2977:    INITIALIZE11:
2978:            P0,     DCS-CTR[C15,],                  ;DISABLE DCS-CTR FOR NOW
2979:            EMITC,  EMIT/000000,                    ;FOR UBREAK REGISTER LOAD
2980:            P2,     PS[3-0]_D[3-0],                 ;

2981:                    UBREAK_BUSDIN[11-00],           ;
2982:            P1,     PS[15-12]_D[15:13], PS[7-4]_D[7-4],     ;
2983:                    FPS[7-4]_D[7-4], FLAG[8-0]_D[15-8],     ;
2984:            NEXT,   RUTA[SCOPE],                    ;NO ERROR:  "TMITIALIZE12" (+1, WORDS)
2985:                    J/INITIALIZE12                  ;   ERROR:  "TESTnn"  (BACK AT START)
     (4102)  DCS[0,00,1,1,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,1,1010,,,0,,0000,0,,,11,000,,,000,000,001]
2986:
```

```
2987    4001;
2988    INITIALIZE12:
2989            SETUP,  RETURN/TEST012A,                        ;RETURN TO NEXT TEST START
2990            NEXT,   CALL.(DISPLAY)                          ;GO DISPLAY CONTENTS OF D-REGISTER IN LIGHTS
        (4001)  DCS[0,00,0,0,0,0]  BM[0101,,00,11,,11,11,,101,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,010,001]
2991
2992
2993
2994    ].PAGE*********************************************************************************************************************
2995
2996    .TOC    *   TEST012-050:  IR DECODE (INSTR1, INSTR8, FLTPT, RELATED "BUTS")
2997
2998    ;*****************************************************************************
2999    ;*
3000    ;*      TESTS:  012 - 050                               UNIBUS: 274 + 206
3001    ;*
3002    ;*      FUNCTIONS:
3003    ;*
3004    ;*      THE FOLLOWING TESTS EXERCISE THE IR-DECODE RELATED LOGIC:
3005    ;*
3006    ;*              INSTR-1, INSTR-8, FLOATING-POINT DECODE "BUTS"
3007    ;*              IR<15:12>, IR<11> (TWO), IR<9:6>, IR<5:1> BIT "BUTS"
3008    ;*              MOV/DR7, BYTE/DM6/SM0, DR6-7 DECODE-RELATED "BUTS"
3009    ;*
3010    ;*      NOTE ALSO THAT THE FIRST TIME THE PROCESSOR "UCON"S
3011    ;*              BUSDIN_EMIT, AND EN-CLK-IR, IS EMPLOYED IS IN TEST-012-A.
3012    ;*
3013    ;*****************************************************************************
3014
3015
3016
3017
3018
3019
3020
3021    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - *
3022
3023    ;*** TEST 012 ***
3024    ;TEST-012 USES A DATA PATTERN OF:  "1 111 111 111 111 111" (177777)
3025
3026    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - *
3027
3028    ;* PART A *
3029    ;TEST-012-A CHECKS THAT BUT(IR<15:12>) READS THE "1111" IN IR<15:12>N CORRECTLY
3030    5775:
3031    TEST012A:
3032            P0,     LOAD-ENUA(ZTARGET417),                  ;LOAD EXPECTED ADDRESS AFTER "BUT"
3033                    LOAD-ERROR(TEST012A),                   ;ERROR DIRECTORY KEY
3034                    DCS-CTR(C5,),                           ;COMPARE ENUA:TNUA IN 5, MICROWORDS
3035            NEXT,   J/SETIRA                                ;SETUP FOR IR TESTS
        (5775)  DCS[1,00,1,0,0,0]  BM[1010,,00,11,,11,00,,001,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,111,110,000]
3036
```



```
3037    ;*** SETUP PROCESSOR UCON FOR BUSDIN <- EMIT, CLOCKING INSTRUCTION REGISTER ***
3038    5760:
3039    SETIRA:
3040            SELECT, UCON=PROC,                              ;SELECT PROCESSOR UCON CONTROL:
3041            ENABLE, EN-CLK-IR(15-00),                       ;  ENABLE CLOCK IR OPERATION
3042                    BUSDIN_EMIT(15-00),                     ;  PUT EMIT(15:00] ONTO BUSDIN
3043            P0,     SET-UCON-CONTROL,                       ;LOAD UCON REGISTER AT P0
3044                    BUMP-VERIFY,                            ;COUNT
3045            NEXT,   J/LOAD012A                              ;GO TO FIRST TEST, PART A
        (5760)  DCS[0,00,0,0,0,1]  BM[0000,,00,00,,00,01,,000,,100,,,0,0,0,,0,,1,,1001,,,0,,0000,0,,,11,000,,,000,000,010]
3046
3047
3048    5002:  ;(FREE)
3049    LOAD012A:
3050            P2=U,   IR_EMIT,                                ;LOAD IR WITH TEST PATTERN
3051                    EMIT/177777,                            ;(177777)
3052            NEXT,   J/GOBUT012A                             ;GO SETUP FOR "BUT"
        (5002)  DCS[0,00,0,0,0,0]  BM[1111,,00,11,,11,11,,111,,111,,,0,0,0,,0,,1,1010,,,0,,0000,0,,,11,000,,,000,000,011]
3053
3054
3055    5003:  ;(FREE)
3056    GOBUT012A:
3057            SETUP,  RETURN/TEST012B,                        ;RETURN TO START OF NEXT SUBTEST
3058            NEXT,   GOTO-PAGE(7),                           ;BUT TABLE IS ON PAGE 7
3059                    J/BUTIR15-12                            ;GO DO "BUT" ON IR<15:12>N
        (5003)  DCS[0,00,0,0,0,0]  BM[0101,,00,11,,11,11,,011,,111,,,0,0,0,,0,,0,0000,,,0,,0000,0,,,11,100,,,011,000,000]
3060
3061
3062    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - *
3063
3064    ;* PART B *
3065    ;TEST-012-B CHECKS THAT BUT(IR<11>&FLTPT<3:0>) READS THE "1" IN IR<11>N CORRECTLY,
3066    ;AND THE FLTPT DECODE ROM GETS ADDRESS (776), WHICH IS A CMPF/D INSTR;
3067    ;DATA OUTPUT SHOULD BE (17)
3068    5773:
3069    TEST012B:
3070            P0,     LOAD-ENUA(ZTARGET437),                  ;LOAD EXPECTED ADDRESS AFTER "BUT"
3071                    LOAD-ERROR(TEST012B),                   ;ERROR DIRECTORY KEY
3072                    DCS-CTR(C3,),                           ;COMPARE ENUA:TNUA IN 3, MICROWORDS
3073            NEXT,   J/GOBUT012B                             ;GO SETUP FOR "BUT"
        (5773)  DCS[1,00,1,0,0,0]  BM[1100,,00,11,,11,00,,011,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,000,100]
3074
3075
3076    5004:  ;(FREE)
3077    GOBUT012B:
3078            SETUP,  RETURN/TEST012C,                        ;RETURN TO START OF NEXT SUBTEST

3079            NEXT,   GOTO-PAGE(7),                           ;BUT TABLE IS ON PAGE 7
3080                    J/BUTIR11&FLTPT3=0                      ;GO DO "BUT" ON IR<11>N&FLTPT<3:0>N
        (5004)  DCS[0,00,0,0,0,0]  BM[0101,,00,11,,11,11,,001,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,000,010]
3081
3082
3083    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
```

```
3084
3085    ;* PART C *
3086    ;TEST-012-C CHECKS THAT BUTR(IR<11>B) READS THE '1' IN IR<11>H CORRECTLY
3087    5771;
3088    TEST012C:
3089            PO,     LOAD-ENUA(ZTARGET403),                  ;LOAD EXPECTED ADDRESS AFTER "BUT"
3090                    LOAD-ERROR(TEST012C),                   ;ERROR DIRECTORY KEY
3091                    DCS-CTR(C),,                            ;COMPARE ENUA;TWUA IN 3, MICROWORDS
3092                    BUMP-VERIFY,                            ;COUNT
3093            NEXT,   J/GOBUT012C                             ;GO SETUP FOR "BUT"
    (5771)  DCS[1,00,1,0,0,1]  BM[1100,,00,11,,11,00,,000,,011,,0,0,0,,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,000,101]
3094
3095
3096    5005:  ;(FREE)
3097    GOBUT012C:
3098            SETUP,  RETURN/TEST012D,                        ;RETURN TO START OF NEXT SUBTEST
3099            NEXT,   GOTO-PAGE(7),                           ;BUT TABLE IS ON PAGE 7
3100                    J/BUTIR11B                              ;GO DO "BUT" ON IR<11>H
    (5005)  DCS[0,00,0,0,0,0]  BM[0101,,00,11,,11,10,,111,,111,,,0,0,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,100,010]
3101
3102
3103    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
3104
3105    ;* PART D *
3106    ;TEST-012-D CHECKS THAT BUT(IR<9:6>) READS THE "1111" IN IR<9:6>H CORRECTLY
3107    5767;
3108    TEST012D:
3109            PO,     LOAD-ENUA(ZTARGET417),                  ;LOAD EXPECTED ADDRESS AFTER "BUT"
3110                    LOAD-ERROR(TEST012D),                   ;ERROR DIRECTORY KEY
3111                    DCS-CTR(C),,                            ;COMPARE ENUA;TWUA IN 3, MICROWORDS
3112            NEXT,   J/GOBUT012D                             ;GO SETUP FOR "BUT"
    (5767)  DCS[1,00,1,0,0,0]  BM[1100,,00,11,,11,00,,001,,111,,,0,0,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,000,110]
3113
3114
3115    5006:  ;(FREE)
3116    GOBUT012D:
3117            SETUP,  RETURN/TEST012E,                        ;RETURN TO START OF NEXT SUBTEST
3118            NEXT,   GOTO-PAGE(7),                           ;BUT TABLE IS ON PAGE 7
3119                    J/BUTIR9-6                              ;GO DO "BUT" ON IR<9:6>H
    (5006)  DCS[0,00,0,0,0,0]  BM[0101,,00,11,,11,10,,101,,111,,,0,0,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,000,100]
3120
3121
3122    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
3123
3124    ;* PART E *
3125    ;TEST-012-E CHECKS THAT BUT(MOV-DR7#IR<5:3>) READS THE (-FLTPT*MOV+FLTPT*DR7) AND "111" IN IR<5:3>H CORRECTLY
3126    5765;
3127    TEST012E:
3128            PO,     LOAD-ENUA(ZTARGET417),                  ;LOAD EXPECTED ADDRESS AFTER "BUT"
3129                    LOAD-ERROR(TEST012E),                   ;ERROR DIRECTORY KEY
3130                    DCS-CTR(C),,                            ;COMPARE ENUA;TWUA IN 3, MICROWORDS
3131            NEXT,   J/GOBUT012E                             ;GO SETUP FOR "BUT"
```



```
    (5765)  DCS[1,00,1,0,0,0]  BM[1100,,00,11,,11,00,,001,,111,,,0,0,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,000,111]
3132
3133
3134    5007:  ;(FREE)
3135    GOBUT012E:
3136            SETUP,  RETURN/TEST012F,                        ;RETURN TO START OF NEXT SUBTEST
3137            NEXT,   GOTO-PAGE(7),                           ;BUT TABLE IS ON PAGE 7
3138                    J/BUTMOVDR7IR5=3                        ;GO DO "BUT" ON (-FLTPT*MOV+FLTPT*DR7) # IR<5:3>H
    (5007)  DCS[0,00,0,0,0,0]  BM[0101,,00,11,,11,10,,100,,111,,,0,0,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,000,101]
3139
3140
3141    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
3142
3143    ;* PART F *
3144    ;TEST-012-F CHECKS THAT BUT(DR6-7 L) READS THE "11" IN IR<2:1> H CORRECTLY
3145    ;AND DOES ASSERT THE SIGNAL
3146    5764;
3147    TEST012F:
3148            PO,     LOAD-ENUA(ZTARGET402),                  ;LOAD EXPECTED ADDRESS AFTER "BUT"
3149                    LOAD-ERROR(TEST012F),                   ;ERROR DIRECTORY KEY
3150                    DCS-CTR(C),,                            ;COMPARE ENUA;TWUA IN 3, MICROWORDS
3151            NEXT,   J/GOBUT012F                             ;GO SETUP FOR "BUT"
    (5764)  DCS[1,00,1,0,0,0]  BM[1100,,00,11,,11,00,,000,,010,,,0,0,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,001,000]
3152
3153
3154    5010:  ;(FREE)
3155    GOBUT012F:
3156            SETUP,  RETURN/TEST012G,                        ;RETURN TO START OF NEXT SUBTEST
3157            PO,     BUMP-VERIFY,                            ;COUNT
3158            NEXT,   GOTO-PAGE(7),                           ;BUT TABLE IS ON PAGE 7
3159                    J/BUTDR6-7L                             ;GO DO "BUT" ON DR 6/7 L
    (5010)  DCS[0,00,0,0,0,1]  BM[0101,,00,11,,11,10,,011,,111,,,0,0,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,100,111]
3160
3161
3162    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
3163
3164    ;* PART G *
3165    ;TEST-012G CHECKS THAT BUT(INSTR) READS THE ONES IN IR<15:00>H CORRECTLY
3166    ;AS CLASS-A-FLTPT, AND CORRECTLY TARGETS TO INSTR; FLTPT (474)
3167    ;  BIT<1:0> = FLAG<4:5>H = "00" FROM INITIALIZATION CODE
3168    5763;
3169    TEST012G:
3170            PO,     LOAD-ENUA(ZTARGET474),                  ;LOAD EXPECTED ADDRESS AFTER "BUT"
3171                    LOAD-ERROR(TEST012G),                   ;ERROR DIRECTORY KEY
3172                    DCS-CTR(C),,                            ;COMPARE ENUA;TWUA IN 3, MICROWORDS
3173            NEXT,   J/GOBUT012G                             ;GO SETUP FOR "BUT"

    (5763)  DCS[1,00,1,0,0,0]  BM[1100,,00,11,,11,00,,111,,100,,,0,0,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,001,001]
3174
3175
3176    5011:  ;(FREE)
3177    GOBUT012G:
3178            SETUP,  RETURN/SCOPE012,                        ;RETURN TO SCOPE LOOP TEST WORD
```

```
3179            NEXT,   GOTO-PAGE(7),                   ;BUT TABLE IS ON PAGE 7
3180                    J/BUTINSTR1                     ;GO DO INSTR; "BUT"
     (5011)  DCS[0,00,0,0,0,0]  BM[0101,,00,00,,00,01,,010,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,000,110]
3181
3182
3183
3184    5012:  ;(FREE)
3185    SCOPE012:
3186            PO,     BUMP-VERIFY,                    ;COUNT
3187            NEXT,   BUTD[SCOPE],                    ;NO ERROR:  "TEST013A" [+1, WORDS]
3188                    J/TEST013A                      ;   ERROR:  "SETIRA"   [-16, WORDS]
     (5012)  DCS[0,00,0,1,0,1]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,111,110,001]
3189
3190
3191
3192
3193
3194
3195    ;* - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - *
3196
3197    ;*** TEST 013 ***
3198    ;TEST-013 USES AN IR DATA PATTERN OF:  "0 000 000 000 000 000" (000000)
3199
3200    ;  - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - *
3201
3202    ;* PART A *
3203    ;TEST-013-A CHECKS THAT BUT [IR<15:12>] READS THE "0000" IN IR<15:12>H CORRECTLY
3204    5761:
3205    TEST013A:
3206            PO,     LOAD-ENUA(ZTARGET400),          ;LOAD EXPECTED ADDRESS AFTER "BUT"
3207                    LOAD-ERROR(TEST013A),           ;ERROR DIRECTORY KEY
3208                    DCS-CTR(C4,),                   ;COMPARE ENUA:TNUA IN 4, MICROWORDS
3209            NEXT,   J/LOAD013A                      ;GO LOAD PATTERN
     (5761)  DCS[1,00,1,0,0,0]  BM[1011,,00,11,,11,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,111,100,100]
3210
3211
3212    5744:
3213    LOAD013A:
3214            P2=0,   IR_EMIT,                        ;LOAD IR WITH TEST PATTERN
3215                    EMIT/000000,                    ;(000000)
3216            NEXT,   J/GOBUT013A                     ;GO SETUP FOR "BUT"
     (5744)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,1,1010,,,0,,0000,0,,,11,000,,,000,001,011]
3217
3218
3219    5013:  ;(FREE)
3220    GOBUT013A:
3221            SETUP,  RETURN/TEST013B,                ;RETURN TO START OF NEXT SUBTEST
3222            NEXT,   GOTO-PAGE(7),                   ;BUT TABLE IS ON PAGE 7
3223                    J/BUTIR15-12                    ;GO DO "BUT" ON IR<15:12>H
     (5013)  DCS[0,00,0,0,0,0]  BM[0101,,00,11,,11,01,,111,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,000,000]
3224
3225
```



```
3226    ;  - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - *
3227
3228    ;* PART B *
3229    ;TEST-013-B CHECKS THAT BUT[IR<11>&FLTPT<1:0>] READS THE "0" IN IR<11>H CORRECTLY,
3230    ;AND THE FLTPT DECODE ROM GETS ADDRESS (000), WHICH IS A &ETP/SETI/CFCC INSTR;
3231    ;DATA OUTPUT SHOULD BE (00)
3232    5767:
3233    TEST013B:
3234            PO,     LOAD-ENUA(ZTARGET400),          ;LOAD EXPECTED ADDRESS AFTER "BUT"
3235                    LOAD-ERROR(TEST013B),           ;ERROR DIRECTORY KEY
3236                    DCS-CTR(C3,),                   ;COMPARE ENUA-TNUA IN 3, MICROWORDS
3237                    BUMP-VERIFY,                    ;COUNT
3238            NEXT,   J/GOBUT013B                     ;GO SETUP FOR "BUT"
     (5767)  DCS[1,00,1,0,0,1]  BM[1100,,00,11,,11,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,001,100]
3239
3240
3241    5014:  ;(FREE)
3242    GOBUT013B:
3243            SETUP,  RETURN/TEST013C,                ;RETURN TO START OF NEXT SUBTEST
3244            NEXT,   GOTO-PAGE(7),                   ;BUT TABLE IS ON PAGE 7
3245                    J/BUTIR11&FLTPT3-0              ;GO DO "BUT" ON IR<11>H&FLTPT<3:0>H
     (5014)  DCS[0,00,0,0,0,0]  BM[0101,,00,11,,11,01,,101,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,000,010]
3246
3247
3248    ;  - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - *
3249
3250    ;* PART C *
3251    ;TEST-013-C CHECKS THAT BUTR[IR<11>H] READS THE "0" IN IR<11>H CORRECTLY
3252    5755:
3253    TEST013C:
3254            PO,     LOAD-ENUA(ZTARGET401),          ;LOAD EXPECTED ADDRESS AFTER "BUT"
3255                    LOAD-ERROR(TEST013C),           ;ERROR DIRECTORY KEY
3256                    DCS-CTR(C3,),                   ;COMPARE ENUA:TNUA IN 3, MICROWORDS
3257            NEXT,   J/GOBUT013C                     ;GO SETUP FOR "BUT"
     (5755)  DCS[1,00,1,0,0,0]  BM[1100,,00,11,,11,00,,000,,001,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,001,101]
3258
3259
3260    5015:  ;(FREE)
3261    GOBUT013C:
3262            SETUP,  RETURN/TEST013D,                ;RETURN TO START OF NEXT SUBTEST
3263            NEXT,   GOTO-PAGE(7),                   ;BUT TABLE IS ON PAGE 7
3264                    J/BUTIR1&R                      ;GO DO BUT ON IR<11>H
     (5015)  DCS[0,00,0,0,0,0]  BM[0101,,00,11,,11,01,,011,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,100,010]
3265
3266
3267    ;  - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - *

3268
3269    ;* PART D *
3270    ;TEST-013-D CHECKS THAT BUT[IR<9:6>] READS THE "0000" IN IR<9:6>H CORRECTLY

3271    5753:
3272    TEST013D:
3273            PO,     LOAD-ENUA(ZTARGET400),          ;LOAD EXPECTED ADDRESS AFTER "BUT"
```

```
                LOAD-ERROR(TEST013D),                         !ERROR DIRECTORY KEY
                DCS-CTR(C3,),                                 !COMPARE ENUA:TNUA IN 3, MICROWORDS
        NEXT,   J/GOBUT013D                                   !GO SETUP FOR "BUT"
(5753)  DCS[1,00,1,0,0,0]  BM[1100,,00,11,,11,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,001,110]
   5016:  !(FREE)
   GOBUT013D:
        SETUP,  RETURN/TEST013E,                              !RETURN TO START OF NEXT SUBTEST
        NEXT,   GOTO-PAGE(7),                                 !BUT TABLE IS ON PAGE 7
                J/BUTIR9-6                                    !GO DO "BUT" ON IR<9:6>H
(5016)  DCS[0,00,0,0,0,0]  BM[0101,,00,11,,11,01,,001,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,000,100]
   ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
   !* PART E *
   !TEST-013-E CHECKS THAT BUT[MOV-DR7&IR<5:3>] READS THE -(-FLTPT+MOV+FLTPT+DR7) AND "000" IN IR<5:3>H CORRECTLY
   5751:
   TEST013E:
        PO,     LOAD-ENUA(ZTARGET400),                        !LOAD EXPECTED ADDRESS AFTER "BUT"
                LOAD-ERROR(TEST013E),                         !ERROR DIRECTORY KEY
                DCS-CTR(C3,),                                 !COMPARE ENUA:TNUA IN 3, MICROWORDS
        NEXT,   J/GOBUT013E                                   !GO SETUP FOR "BUT"
(5751)  DCS[1,00,1,0,0,0]  BM[1100,,00,11,,11,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,001,111]
   5017:  !(FREE)
   GOBUT013E:
        SETUP,  RETURN/TEST013F,                              !RETURN TO START OF NEXT SUBTEST
        PO,     BUMP-VERIFY,                                  !COUNT
        NEXT,   GOTO-PAGE(7),                                 !BUT TABLE IS ON PAGE 7
                J/BUTMOVDR7IR5-3                              !GO DO "BUT" ON (-FLTPT+MOV+FLTPT+DR7) & IR<5:3>H
(5017)  DCS[0,00,0,0,0,1]  BM[0101,,00,11,,11,01,,000,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,000,101]
   ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
   !* PART F *
   !TEST-013-F CHECKS THAT BUT[INSTR8] READS THE ZEROS IN IR<15:00>H CORRECTLY
   !AS (000000)=HALT, AND CORRECTLY TARGETS TO (434)
   5750:
   TEST013F:
        PO,     LOAD-ENUA(ZTARGET434),                        !LOAD EXPECTED ADDRESS AFTER "BUT"
                LOAD-ERROR(TEST013F),                         !ERROR DIRECTORY KEY
                DCS-CTR(C3,),                                 !COMPARE ENUA:TNUA IN 3, MICROWORDS
        NEXT,   J/GOBUT013F                                   !GO SETUP FOR "BUT"
(5750)  DCS[1,00,1,0,0,0]  BM[1100,,00,11,,11,00,,011,,100,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,010,000]
   5020:  !(FREE)
   GOBUT013F:
```



```
        SETUP,  RETURN/TEST013G,                              !RETURN TO START OF NEXT SUBTEST
        NEXT,   GOTO-PAGE(7),                                 !BUT TABLE IS ON PAGE 7
                J/BUTINSTR8                                   !GO DO INSTR8 "BUT"
(5020)  DCS[0,00,0,0,0,0]  BM[0101,,00,11,,11,00,,111,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,000,001]
   ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
   !* PART G *
   !TEST-013-G CHECKS THAT BUT[INSTR1] READS THE ZEROS IN IR<15:00>H CORRECTLY
   !AS NOT(CLASS-A THRU G), AND CORRECTLY TARGETS TO (417) [OTHER]
   5747:
   TEST013G:
        PO,     LOAD-ENUA(ZTARGET417),                        !LOAD EXPECTED ADDRESS AFTER "BUT"
                LOAD-ERROR(TEST013G),                         !ERROR DIRECTORY KEY
                DCS-CTR(C3,),                                 !COMPARE ENUA:TNUA IN 3, MICROWORDS
        NEXT,   J/GOBUT013G                                   !GO SETUP FOR "BUT"
(5747)  DCS[1,00,1,0,0,0]  BM[1100,,00,11,,11,00,,001,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,010,001]
   5021:  !(FREE)
   GOBUT013G:
        SETUP,  RETURN/SCOPE013,                              !RETURN TO SCOPE LOOP TEST WORD
        NEXT,   GOTO-PAGE(7),                                 !BUT TABLE IS ON PAGE 7
                J/BUTINSTR1                                   !GO DO INSTR1 "BUT"
(5021)  DCS[0,00,0,0,0,0]  BM[0101,,00,00,,00,10,,010,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,000,110]
   5022:  !(FREE)
   SCOPE013:
        NEXT,   BUTO[SCOPE],                                  !NO ERROR:  "TEST014A" [+1, WORD]
                J/TEST014A                                    !   ERROR:  "LOAD013A" [-14, WORDS]
(5022)  DCS[0,00,0,1,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,111,100,101]
   ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
   !*** TEST 014 ***
   !TEST-014 USES AN IR DATA PATTERN OF:  "0 000 000 001 010 101" (000125)
   ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
   !* PART A *
   !TEST-014-A CHECKS THAT BUT[INSTR5] READS THE IR CORRECTLY
   !AS ROM ADDRESS=(125) ON THE INSTRS E&S ROM, AND RECEIVES THE DIAGNOSTIC VALUE
   !OF (12), TARGETING TO (432) AFTER THE DECODE
   5745:
   TEST014A:
```

```
3369            P0,     LOAD=ENUA(ZTARGET432),                  ;LOAD EXPECTED ADDRESS AFTER "BUT"
3370                    LOAD=ERROR(TEST014A),                   ;ERROR DIRECTORY KEY
3371                    DCS=CTR(C4,),                           ;COMPARE ENUA;TNUA IN 4, MICROWORDS
3372            NEXT,   J/LOAD014A                              ;GO LOAD PATTERN
    (5745)  DCS(1,00,1,0,0,0)  BM(1011,,00,11,,11,00,,011,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,111,100,010)
3373
3374
3375    5742:
3376    LOAD014A:
3377            P2=U,   IR_EMIT,                                ;LOAD IR WITH TEST PATTERN
3378                    EMIT/000125,                            ;(000125)
3379            NEXT,   J/GOBUT014A                             ;GO SETUP FOR "BUT"
    (5742)  DCS(0,00,0,0,0,0)  BM(0000,,00,00,,00,01,,010,,101,,,0,0,0,,0,,0,,,1,1010,,,0,,0000,0,,,11,000,,,000,010,011)
3380
3381    5023:  ;(FREE)
3382    GOBUT014A:
3383            SETUP,  RETURN/TEST014B,                        ;RETURN TO START OF NEXT SUBTEST
3384            NEXT,   GOTO=PAGE(7),                           ;BUT TABLE IS ON PAGE 7
3385                    J/BUTINSTR6                             ;GO DO INSTR6 "BUT"
    (5023)  DCS(0,00,0,0,0,0)  BM(0101,,00,11,,11,01,,010,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,000,001)
3386
3387
3388    ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
3389
3390    !* PART B *
3391    !TEST=014=B CHECKS THAT BUT(DR6=7 L) READS THE "10" IN IR<5:1> H CORRECTLY
3392    !AND DOES NOT ASSERT THE SIGNAL
3393    5752:
3394    TEST014B:
3395            P0,     LOAD=ENUA(ZTARGET403),                  ;LOAD EXPECTED ADDRESS AFTER "BUT"
3396                    LOAD=ERROR(TEST014B),                   ;ERROR DIRECTORY KEY
3397                    DCS=CTR(C3,),                           ;COMPARE ENUA;TNUA IN 3, MICROWORDS
3398                    BUMP=VERIFY,                            ;COUNT
3399            NEXT,   J/GOBUT014B                             ;GO SETUP FOR "BUT"
    (5752)  DCS(1,00,1,0,0,1)  BM(1100,,00,11,,11,00,,000,,011,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,010,100)
3400
3401
3402    5024:  ;(FREE)
3403    GOBUT014B:
3404            SETUP,  RETURN/TEST014C,                        ;RETURN TO START OF NEXT SUBTEST
3405            NEXT,   GOTO=PAGE(7),                           ;BUT TABLE IS ON PAGE 7
3406                    J/BUTDR6=7L                             ;GO DO DR 6=7 L "BUT"
    (5024)  DCS(0,00,0,0,0,0)  BM(0101,,00,11,,11,10,,010,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,100,111)
3407
3408    ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
3409
3410    !* PART C *
3411    !TEST=014=C CHECKS THAT BUT(INSTR1) READS THE IR CORRECTLY
3412    !AS CLASS=6=JMP, IR<5:3>H="010", AND TARGETS TO (652)
3413    5762:
3414    TEST014C:
3415            P0,     LOAD=ENUA(ZTARGET652),                  ;LOAD EXPECTED ADDRESS AFTER "BUT"
```



```
3416                    LOAD=ERROR(TEST014C),                   ;ERROR DIRECTORY KEY
3417                    DCS=CTR(C1,),                           ;COMPARE ENUA;TNUA IN 3, MICROWORDS
3418            NEXT,   J/GOBUT014C                             ;GO SETUP FOR "BUT"
    (5762)  DCS(1,00,1,0,0,0)  BM(1100,,00,11,,11,10,,101,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,010,101)
3419
3420
3421    5025:  ;(FREE)
3422    GOBUT014C:
3423            SETUP,  RETURN/TEST014D,                        ;RETURN TO START OF NEXT SUBTEST
3424            NEXT,   GOTO=PAGE(7),                           ;BUT TABLE IS ON PAGE 7
3425                    J/BUTINSTR1                             ;GO DO INSTR1 "BUT"
    (5025)  DCS(0,00,0,0,0,0)  BM(0101,,00,11,,10,01,,000,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,000,110)
3426
3427
3428
3429    ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
3430
3431    !* PART D *
3432    !TEST=014=D CHECKS THAT BUT(IR<11>=FLTPT<3:0>) READS THE "0" IN IR<11>H CORRECTLY,
3433    !AND THE FLTPT DECODE ROM GETS ADDRESS (020), WHICH IS A BTST INSTR;
3434    !DATA OUTPUT SHOULD BE (01)
3435    5710:
3436    TEST014D:
3437            P0,     LOAD=ENUA(ZTARGET401),                  ;LOAD EXPECTED ADDRESS AFTER "BUT"
3438                    LOAD=ERROR(TEST014D),                   ;ERROR DIRECTORY KEY
3439                    DCS=CTR(C3,),                           ;COMPARE ENUA;TNUA IN 3, MICROWORDS
3440            NEXT,   J/GOBUT014D                             ;GO SETUP FOR "BUT"
    (5710)  DCS(1,00,1,0,0,0)  BM(1100,,00,11,,11,00,,000,,001,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,010,110)
3441
3442
3443    5026:  ;(FREE)
3444    GOBUT014D:
3445            SETUP,  RETURN/SCOPE014,                        ;RETURN TO SCOPE LOOP TEST WORD
3446            NEXT,   GOTO=PAGE(7),                           ;BUT TABLE IS ON PAGE 7
3447                    J/BUTIR11=FLTPT3=0                      ;GO DO "BUT" ON IR<11>H=FLTPT<3:0>H
    (5026)  DCS(0,00,0,0,0,0)  BM(0101,,00,00,,00,10,,111,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,000,010)
3448
3449
3450
3451    5027:  ;(FREE)
3452    SCOPE014:
3453            NEXT,   BUTD[SCOPE],                            ;NO ERROR:  "TEST015A" (+1, WORD)
3454                    J/TEST015A                              ;   ERROR:  "LOAD014A" (+0, WORD)
    (5027)  DCS(0,00,0,1,0,0)  BM(0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,111,100,011)
3455
3456
3457
3458
3459
3460
3461    ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
3462
```

```
3463    ;*** TEST 015 ***
3464    ;TEST-015 USES AN IR DATA PATTERN OF:  "0 000 000 001 101 010" (0000152)
3465
3466    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
3467
3468    ;* PART A *
3469    ;TEST-015-A CHECKS THAT BUT[INSTR] READS THE IR CORRECTLY
3470    ;AS ROM ADDRESS=(152) ON THE INSTRS ESS ROM, AND RECEIVES THE DIAGNOSTIC VALUE
3471    ;OF (05), TARGETING TO (425) AFTER THE DECODE
3472    5743:
3473    TEST015A:
3474            P0,     LOAD=ENUA(TARGET425),                   ;LOAD EXPECTED ADDRESS AFTER "BUT"
3475                    LOAD=ERROR(TEST015A),                   ;ERROR DIRECTORY KEY
3476                    DCS=CTR(C4,),                           ;COMPARE ENUA:TNUA IN 4, MICROWORDS
3477            NEXT,   J/LOAD015A                              ;GO LOAD PATTERN
    (5743)  DCS[1,00,1,0,0,0]  BM[1011,,00,11,,11,00,,010,,101,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,111,100,000]
3478
3479
3480    5740:
3481    LOAD015A:
3482            P0,     BUMP=VERIFY,                            ;COUNT
3483            P2=0,   IR_EMIT,                                ;LOAD IR WITH TEST PATTERN
3484                    EMIT/0000152,                           ;(0001[illegible]
3485            NEXT,   J/GOBUT015A                             ;GO SETUP FOR "BUT"
    (5740)  DCS[0,00,0,0,0,1]  BM[0000,,00,00,,00,01,,101,,010,,,0,0,0,,0,,0,,,1,1010,,,0,,0000,0,,,11,000,,,000,011,000]
3486
3487    5030:  ;(FREE)
3488    GOBUT015A:
3489            SETUP,  RETURN/TEST015B,                        ;RETURN TO START OF NEXT SUBTEST
3490            NEXT,   GOTO=PAGE(7),                           ;BUT TABLE IS ON PAGE 7
3491                    J/BUTINSTR                              ;GO DO INSTR "BUT"
    (5030)  DCA[0,00,0,0,0,0]  BM[0101,,00,11,,11,11,,010,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,000,001]
3492
3493
3494    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
3495
3496    ;* PART B *
3497    ;TEST-015-B CHECKS THAT BUT[DR6=7 L] READS THE "01" IN IR<11:10> N CORRECTLY
3498    ;AND DOES NOT ASSERT THE SIGNAL
3499    5772:
3500    TEST015B:
3501            P0,     LOAD=ENUA(TARGET403),                   ;LOAD EXPECTED ADDRESS AFTER "BUT"
3502                    LOAD=ERROR(TEST015B),                   ;ERROR DIRECTORY KEY
3503                    DCS=CTR(C3,),                           ;COMPARE ENUA:TNUA IN 3, MICROWORDS
3504            NEXT,   J/GOBUT015B                             ;GO SETUP FOR "BUT"
    (5772)  DCS[1,00,1,0,0,0]  BM[1100,,00,11,,11,00,,000,,011,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,011,001]
3505
3506
3507    5031:  ;(FREE)
3508    GOBUT015B:
3509            SETUP,  RETURN/TEST015C,                        ;RETURN TO START OF NEXT SUBTEST
3510            NEXT,   GOTO=PAGE(7),                           ;BUT TABLE IS ON PAGE 7
3511                    J/BUTDR6=7L                             ;GO DO DR 6=7 L "BUT"
```



```
    (5031)  DCS[0,00,0,0,0,0]  BM[0101,,00,11,,10,10,,010,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,100,111]
3512
3513    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
3514
3515    ;* PART C *
3516    ;TEST-015-C CHECKS THAT BUT[INSTR1] READS THE IR CORRECTLY
3517    ;AS CLASS=E=JMP, IR<5:3>H="101", AND TARGETS TO (656)
3518    5722:
3519    TEST015C:
3520            P0,     LOAD=ENUA(TARGET656),                   ;LOAD EXPECTED ADDRESS AFTER "BUT"
3521                    LOAD=ERROR(TEST015C),                   ;ERROR DIRECTORY KEY
3522                    DCS=CTR(C3,),                           ;COMPARE ENUA:TNUA IN 3, MICROWORDS
3523            NEXT,   J/GOBUT015C                             ;GO SETUP FOR "BUT"
    (5722)  DCS[1,00,1,0,0,0]  BM[1100,,00,11,,11,10,,101,,101,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,011,010]
3524
3525
3526    5032:  ;(FREE)
3527    GOBUT015C:
3528            SETUP,  RETURN/SCOPE015,                        ;RETURN TO SCOPE LOOP TEST WORD
3529            NEXT,   GOTO=PAGE(7),                           ;BUT TABLE IS ON PAGE 7
3530                    J/BUTINSTR1                             ;GO DO INSTR1 "BUT"
    (5032)  DCS[0,00,0,0,0,0]  BM[0101,,00,00,,00,11,,011,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,000,110]
3531
3532    5033:  ;(FREE)
3533    SCOPE015:
3534            P0,     BUMP=VERIFY,                            ;COUNT
3535            NEXT,   BUT0[SCOPE],                            ;NO ERROR:  "TEST016A" [+1, WORD]
3536                    J/TEST016A                              ;   ERROR:  "LOAD015A" [-5, WORDS]
    (5033)  DCS[0,00,0,1,0,1]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,111,100,001]
3537
3538
3539
3540
3541
3542
3543    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
3544
3545    ;*** TEST 016 ***
3546    ;TEST-016 USES AN IR DATA PATTERN OF:  "1 000 000 000 000 000" (1000000)
3547
3548    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
3549
3550    ;* PART A *
3551    ;TEST-016-A CHECKS THAT BUT[INSTR1] READS THE IR CORRECTLY
3552    ;AS CLASS=F=BRANCH, IR<14:11>H="0000", IR<15#10:08>H="0000",
3553    ;AND TARGETS TO (757)

3554    5741:
3555    TEST016A:
3556            P0,     LOAD=ENUA(TARGET757),                   ;LOAD EXPECTED ADDRESS AFTER "BUT"
3557                    LOAD=ERROR(TEST016A),                   ;ERROR DIRECTORY KEY
3558                    DCS=CTR(C4,),                           ;COMPARE ENUA:TNUA IN 4, MICROWORDS
3559            NEXT,   J/LOAD016A                              ;GO LOAD PATTERN
```

```
    (5741)  DCS[1,00,1,0,0,0]  BM[1011,,00,11,,11,11,,101,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,111,011,110]


        5736:
        LOAD016A:
                P2-U,   IR_EMIT,                                ;LOAD IR WITH TEST PATTERN
                        EMIT/100000,                            ;(100000)
                NEXT,   J/GOBUT016A                             ;GO SETUP FOR "BUT"
    (5736)  DCS[0,00,0,0,0,0]  BM[1000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,1,1010,,,0,,0000,0,,,11,000,,,000,011,100]


        5034:  ;(FREE)
        GOBUT016A:
                SETUP,  RETURN/SCOPE016,                        ;RETURN TO SCOPE LOOP TEST WORD
                P0,     BUMP-VERIFY,                            ;COUNT
                NEXT,   GOTO-PAGE(7),                           ;BUT TABLE IS ON PAGE 7
                        J/BUTINSTR1                             ;GO DO INSTR1 "BUT"
    (5034)  DCS[0,00,0,0,0,1]  BM[0101,,00,00,,00,11,,101,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,000,110]


        5035:  ;(FREE)
        SCOPE016:
                NEXT,   BUTD[SCOPE],                            ;NO ERROR:  "TEST017A" (+1, WORD)
                        J/TEST017A                              ;   ERROR:  "LOAD016A" (-3, WORDS)
    (5035)  DCS[0,00,0,1,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,111,011,111]






        ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        ;*** TEST 017 ***
        ;TEST-017 USES A DATA PATTERN OF:  "0 000 000 101 010 010" (000522)

        ;  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -

        ;* PART A *
        ;TEST-017-A CHECKS THAT BUT[IR<9:6>] READS THE ALTERNATING PATTERN "0101"
        ;IN IR<9:6>H CORRECTLY
        5737:
        TEST017A:
                P0,     LOAD-ENUA(ZTARGET405),                  ;LOAD EXPECTED ADDRESS AFTER "BUT"
                        LOAD-ERROR(TEST017A),                   ;ERROR DIRECTORY KEY
                        DCS-CTR(C4,),                           ;COMPARE ENUA:TNUA IN 4, MICROWORDS
                NEXT,   J/LOAD017A                              ;GO LOAD PATTERN
    (5737)  DCS[1,00,1,0,0,0]  BM[1011,,00,11,,11,00,,000,,101,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,111,011,010]


        5732:
        LOAD017A:
                P2-U,   IR_EMIT,                                ;LOAD IR WITH TEST PATTERN
```



```
                        EMIT/000522,                            ;(000522)
                NEXT,   J/GOBUT017A                             ;GO SETUP FOR "BUT"
    (5732)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,01,01,,010,,010,,,0,0,0,,0,,0,,,1,1010,,,0,,0000,0,,,11,000,,,000,011,110]


        5036:  ;(FREE)
        GOBUT017A:
                SETUP,  RETURN/TEST017B,                        ;RETURN TO START OF NEXT SUBTEST
                NEXT,   GOTO-PAGE(7),                           ;BUT TABLE IS ON PAGE 7
                        J/BUTIR9-6                              ;GO DO "BUT" ON IR<9:6>H
    (5036)  DCS[0,00,0,0,0,0]  BM[0101,,00,11,,10,11,,101,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,000,100]


        ;  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -

        ;* PART B *
        ;TEST-017-B CHECKS THAT BUT[INSTR1] READS THE IR CORRECTLY
        ;AS CLASS-F=BRANCH, IR<14:11>H="0000", IR<15:10:08>H="0000",
        ;AND TARGETS TO (757)
        5735:
        TEST017B:
                P0,     LOAD-ENUA(ZTARGET757),                  ;LOAD EXPECTED ADDRESS AFTER "BUT"
                        LOAD-ERROR(TEST017B),                   ;ERROR DIRECTORY KEY
                        DCS-CTR(C3,),                           ;COMPARE ENUA:TNUA IN 3, MICROWORDS
                        BUMP-VERIFY,                            ;COUNT
                NEXT,   J/GOBUT017B                             ;GO SETUP FOR "BUT"
    (5735)  DCS[1,00,1,0,0,1]  BM[1100,,00,11,,11,11,,101,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,011,111]


        5037:  ;(FREE)
        GOBUT017B:
                SETUP,  RETURN/SCOPE017,                        ;RETURN TO SCOPE LOOP TEST WORD
                NEXT,   GOTO-PAGE(7),                           ;BUT TABLE IS ON PAGE 7
                        J/BUTINSTR1                             ;GO DO INSTR1 "BUT"
    (5037)  DCS[0,00,0,0,0,0]  BM[0101,,00,00,,01,00,,000,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,000,110]


        5040:  ;(FREE)
        SCOPE017:
                NEXT,   BUTD[SCOPE],                            ;NO ERROR:  "TEST020A" (+1, WORD)
                        J/TEST020A                              ;   ERROR:  "LOAD017A" (-4, WORDS)
    (5040)  DCS[0,00,0,1,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,111,011,011]






        ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        ;*** TEST 020 ***
        ;TEST-020 USES A DATA PATTERN OF:  "0 000 001 010 101 111" (001257)
```

```
3655
3656    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
3657
3658    ;* PART A *
3659    ;TEST-020-A CHECKS THAT BUT[IR<9:6>] READS THE ALTERNATING PATTERN "1010"
3660    ;IN IR<9:6>H CORRECTLY
3661    5733:
3662    TEST020A:
3663            P0,     LOAD=ENUA(ZTARGET412),                  ;LOAD EXPECTED ADDRESS AFTER "BUT"
3664                    LOAD=ERROR(TEST020A),                   ;ERROR DIRECTORY KEY
3665                    DCS=CTR(C4,),                           ;COMPARE ENUA:TWUA IN 4, MICROWORDS
3666            NEXT,   J/LOAD020A                              ;GO LOAD PATTERN
    (5733)  DCS[1,00,1,0,0,0]  BM[1011,,00,11,,11,00,,001,,010,,,0,010,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,111,010,100]
3667
3668
3669    5734:
3670    LOAD020A:
3671            P2=0,   IR_EMIT,                                ;LOAD IR WITH TEST PATTERN
3672                    EMIT/001257,                            ;[001257]
3673            NEXT,   J/GOBUT020A                             ;GO SETUP FOR "BUT"
    (5734)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,10,10,,101,,111,,,0,0,0,,0,,0,,,1,1010,,,0,,0000,0,,,11,000,,,000,100,001]
3674
3675
3676    5041:  ;(FREE)
3677    GOBUT020A:
3678            SETUP,  RETURN/TEST020B,                        ;RETURN TO START OF NEXT SUBTEST
3679            NEXT,   GOTO=PAGE(7),                           ;BUT TABLE IS ON PAGE 7
3680                    J/BUTIR9=A                              ;GO DO "BUT" ON IR<9:6>H
    (5041)  DCS[0,00,0,0,0,0]  BM[0101,,00,11,,10,11,,001,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,000,100]
3681
3682
3683    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
3684
3685    ;* PART B *
3686    ;TEST-020-B CHECKS THAT BUT(NOV-DR7&IR<8:3>) READS THE ~(~FLTPT&NOV&FLTPT&DR7) AND
3687    ;ALTERNATING PATTERN "101" IN IR<8:3>H CORRECTLY
3688    5731:
3689    TEST020B:
3690            P0,     LOAD=ENUA(ZTARGET405),                  ;LOAD EXPECTED ADDRESS AFTER "BUT"
3691                    LOAD=ERROR(TEST020B),                   ;ERROR DIRECTORY KEY
3692                    DCS=CTR(C3,),                           ;COMPARE ENUA:TWUA IN 3, MICROWORDS
3693            NEXT,   J/GOBUT020B                             ;GO SETUP FOR "BUT"
    (5731)  DCS[1,00,1,0,0,0]  BM[1100,,00,11,,11,00,,000,,101,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,100,010]
3694
3695
3696    5042:  ;(FREE)
3697    GOBUT020B:
3698            SETUP,  RETURN/TEST020C,                        ;RETURN TO START OF NEXT SUBTEST
3699            P0,     BUMP=VERIFY,                            ;COUNT
3700            NEXT,   GOTO=PAGE(7),                           ;BUT TABLE IS ON PAGE 7
3701                    J/BUTNOVDR7IR8=3                        ;GO DO "BUT"
    (5042)  DCS[0,00,0,0,0,1]  BM[0101,,00,11,,10,10,,111,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,000,101]
```



```
3703
3704    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
3705
3706    ;* PART C *
3707    ;TEST-020-C CHECKS THAT BUT[IR&STR1] READS THE IR CORRECTLY
3708    ;AS CLASS=F=BRANCH, IR<14:11>H="0000", IR<15:10:08>H#"0000",
3709    ;AND TARGETS TO [757]
3710    5727:
3711    TEST020C:
3712            P0,     LOAD=ENUA(ZTARGET757),                  ;LOAD EXPECTED ADDRESS AFTER "BUT"
3713                    LOAD=ERROR(TEST020C),                   ;ERROR DIRECTORY KEY
3714                    DCS=CTR(C3,),                           ;COMPARE ENUA:TWUA IN 3, MICROWORDS
3715            NEXT,   J/GOBUT020C                             ;GO SETUP FOR "BUT"
    (5727)  DCS[1,00,1,0,0,0]  BM[1100,,00,11,,11,11,,101,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,100,011]
3716
3717
3718    5043:  ;(FREF)
3719    GOBUT020C:
3720            SETUP,  RETURN/SCOPE020,                        ;RETURN TO SCOPE LOOP TEST WORD
3721            NEXT,   GOTO=PAGE(7),                           ;BUT TABLE IS ON PAGE 7
3722                    J/BUTINSTR1                             ;GO DO INSTR1 "BUT"
    (5043)  DCS[0,00,0,0,0,0]  BM[0101,,00,00,,01,00,,100,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,000,110]
3723
3724
3725    5044:  ;(FREE)
3726    SCOPE020:
3727            NEXT,   AUTO[SCOPE],                            ;NO ERROR:  "TEST021A" [+1, WRD]
3728                    J/TEST021A                              ;   ERROR:  "LOAD020A" [-5, WORDS]
    (5044)  DCS[0,00,0,1,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,111,010,101]
3729
3730
3731
3732
3733
3734
3735    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
3736
3737    ;*** TEST 021 ***
3738    ;TEST-021 USES A DATA PATTERN OF:  "0 000 010 011 001 101" (002315)
3739
3740    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
3741
3742    ;* PART A *
3743    ;TEST-021-A CHECKS THAT BUT[DM0&SM0&BYTE],
3744    ;DM=IR<5:3>H="001", DM0H=0; SM=IR<11:9>H="010", SM0H=0; BYTE H=0
3745    5725:
3746    TEST021A:
3747            P0,     LOAD=ENUA(ZTARGET400),                  ;LOAD EXPECTED ADDRESS AFTER "BUT"
3748                    LOAD=ERROR(TEST021A),                   ;ERROR DIRECTORY KEY
3749                    DCS=CTR(C4,),                           ;COMPARE ENUA:TWUA IN 4, MICROWORDS
3750            NEXT,   J/LOAD021A                              ;GO LOAD PATTERN
    (5725)  DCS[1,00,1,0,0,0]  BM[1011,,00,11,,11,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,111,010,000]
```

```
    5720:
    LOAD021A:
            P0,     BUMP-VERIFY,                           ;COUNT
            P2-U,   IR_EMIT,                               ;LOAD IR WITH TEST PATTERN
                    EMIT/002315,                           ;(002315)
            NEXT,   J/GOBUT021A                            ;GO SETUP FOR "BUT"
 (5720)  DCS[0,00,0,0,0,1]  BM[0000,,00,01,,00,11,,001,,101,,,0,0,0,,0,,0,,,1,1010,,,0,,0000,0,,,11,000,,,000,100,101]


    5045:  ;(FREE)
    GOBUT021A:
            SETUP,  RETURN/TEST021B,                       ;RETURN TO START OF NEXT SUBTEST
            NEXT,   GOTO-PAGE(7),                          ;BUT TABLE IS ON PAGE 7
                    J/BUTONSMBYTE                          ;GO DO "BUT" ON DMS&MODEBYTE
 (5045)  DCS[0,00,0,0,0,0]  BM[0101,,00,11,,10,10,,011,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,001,011]


    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

    ;* PART B *
    ;TEST-021-B CHECKS THAT BUT(INSTR1) READS THE IR CORRECTLY
    ;AS CLASS-F=BRANCH, IR<14:11>H="0000", IR<15&10:08>H="0000",
    ;AND TARGETS TO (757)
    5723:
    TEST021B:
            P0,     LOAD-ENUA(ZTARGET757),                 ;LOAD EXPECTED ADDRESS AFTER "BUT"
                    LOAD-ERROR(TEST021B),                  ;ERROR DIRECTORY KEY
                    DCS-CTR(C3,),                          ;COMPARE ENUA:TNUA IN 3, MICROWORDS
            NEXT,   J/GOBUT021B                            ;GO SETUP FOR "BUT"
 (5723)  DCS[1,00,1,0,0,0]  BM[1100,,00,11,,11,11,,101,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,100,110]


    5046:  ;(FREE)
    GOBUT021B:
            SETUP,  RETURN/TEST021C,                       ;RETURN TO START OF NEXT SUBTEST
            NEXT,   GOTO-PAGE(7),                          ;BUT TABLE IS ON PAGE 7
                    J/BUTINSTR1                            ;GO DO INSTR1 "BUT"
 (5046)  DCS[0,00,0,0,0,0]  BM[0101,,00,11,,11,11,,100,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,000,110]



    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

    ;* PART C *
    ;TEST-021-C CHECKS THAT BUT(IR<11>!FLTPT<3:0>) READS THE "0" IN IR<11>H CORRECTLY,
    ;AND THE FLTPT DECODE ROM GETS ADDRESS (462), WHICH IS A ADDF/D MODE1-7 INSTR;
    ;DATA OUTPUT SHOULD BE (11)
    5774:
    TEST021C:
            P0,     LOAD-ENUA(ZTARGET411),                 ;LOAD EXPECTED ADDRESS AFTER "BUT"
```



```
                    LOAD-ERROR(TEST021C),                  ;ERROR DIRECTORY KEY
                    DCS-CTR(C3,),                          ;COMPARE ENUA:TNUA IN 3, MICROWORDS
            NEXT,   J/GOBUT021C                            ;GO SETUP FOR "BUT"
 (5774)  DCS[1,00,1,0,0,0]  BM[1100,,00,11,,11,00,,001,,001,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,100,111]


    5047:  ;(FREE)
    GOBUT021C:
            SETUP,  RETURN/SCOPE021,                       ;RETURN TO SCOPE LOOP TEST WORD
            NEXT,   GOTO-PAGE(7),                          ;BUT TABLE IS ON PAGE 7
                    J/BUTIR11FLTPT3-0                      ;GO DO "BUT" ON IR<11>H&FLTPT<3:0>H
 (5047)  DCS[0,00,0,0,0,0]  BM[0101,,00,00,,01,01,,000,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,000,010]


    5050:  ;(FREE)
    SCOPE021:
            NEXT,   BUTD[SCOPE],                           ;NO ERROR:  "TEST022A" [+1, WORDS]
                    J/TEST022A                             ;   ERROR:  "LOAD021A" [-6, WORDS]
 (5050)  DCS[0,00,0,1,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,111,010,001]






    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

    ;*** TEST 022 ***
    ;TEST-022 USES A DATA PATTERN OF:  "0 000 011 000 110 011" (003063)

    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

    ;* PART A *
    ;TEST-022-A CHECKS THAT BUT(INSTR1) READS THE IR CORRECTLY
    ;AS CLASS-F=BRANCH, IR<14:11>H="0000", IR<15&10:08>H="0000",
    ;AND TARGETS TO (757)
    5721:
    TEST022A:
            P0,     LOAD-ENUA(ZTARGET757),                 ;LOAD EXPECTED ADDRESS AFTER "BUT"
                    LOAD-ERROR(TEST022A),                  ;ERROR DIRECTORY KEY
                    DCS-CTR(C4,),                          ;COMPARE ENUA:TNUA IN 4, MICROWORDS
            NEXT,   J/LOAD022A                             ;GO LOAD PATTERN
 (5721)  DCS[1,00,1,0,0,0]  BM[1011,,00,11,,11,11,,101,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,111,001,110]


    5716:

    LOAD022A:
            P2-U,   IR_EMIT,                               ;LOAD IR WITH TEST PATTERN
                    EMIT/003063,                           ;(003063)
            NEXT,   J/GOBUT022A                            ;GO SETUP FOR "BUT"
 (5716)  DCS[0,00,0,0,0,0]  BM[0000,,00,01,,10,00,,110,,011,,,0,0,0,,0,,0,,,1,1010,,,0,,0000,0,,,11,000,,,000,101,001]
```

```
	5051:	!(FREE)
	GOBUT022A:
		SETUP,	RETURN/SCOPE022,			!RETURN TO SCOPE LOOP TEST WORD
		NEXT,	GOTO-PAGE(7),				!BUT TABLE IS ON PAGE 7
			J/BUTINSTR1				!GO DO INSTR1 "BUT"
(5051)	DCS[0,00,0,0,0,0]	BM[0101,,00,00,,01,01,,010,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,000,110]



	5052:	!(FREE)
	SCOPE022:
		NEXT,	BUTD[SCOPE],				!NO ERROR:  "TEST023A" (+1, WORD]
			J/TEST023A				!    ERROR:  "LOAD022A" (-2, WORDS]
(5052)	DCS[0,00,0,1,0,0]	BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,111,001,111]






	! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

	!*** TEST 023 ***
	!TEST-023 USES A DATA PATTERN OF:  "1 000 110 000 000 100" (106004)

	! -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -

	!* PART A *
	!TEST-023-A CHECKS THAT BUT[INSTR1] READS THE IR CORRECTLY
	!AS CLASS-D=SOP*DMO; IR<14:06>H="000 110 000"; IR<15>H="1";
	!DM=IR<5:3>H="000", DMOH=1; AND TARGETS TO (560)
	5717:
	TEST023A:
		P0,	LOAD-ENUA(ZTARGET560),			!LOAD EXPECTED ADDRESS AFTER "BUT"
			LOAD-ERROR(TEST023A),			!ERROR DIRECTORY KEY
			DCS-CTR(C4,),				!COMPARE ENUA:TNUA IN 4, MICROWORDS
		NEXT,	J/LOAD023A				!GO LOAD PATTERN
(5717)	DCS[1,00,1,0,0,0]	BM[1011,,00,11,,11,01,,110,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,111,001,100]


	5714:
	LOAD023A:
		P2-U,	IR_EMIT,				!LOAD IR WITH TEST PATTERN
			EMIT/106004,				!(106004)
		NEXT,	J/GOBUT023A				!GO SETUP FOR "BUT"
(5714)	DCS[0,00,0,0,0,0]	BM[1000,,00,11,,00,00,,000,,100,,,0,0,0,,0,,0,,,1,1010,,,0,,0000,0,,,11,000,,,000,101,011]


	5053:	!(FREE)
	GOBUT023A:
		SETUP,	RETURN/SCOPE023,			!RETURN TO SCOPE LOOP TEST WORD
		NEXT,	GOTO-PAGE(7),				!BUT TABLE IS ON PAGE 7
			J/BUTINSTR1				!GO DO INSTR1 "BUT"
```



```
(5053)	DCS[0,00,0,0,0,0]	BM[0101,,00,00,,01,01,,100,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,000,110]


	5054:	!(FREE)
	SCOPE023:
		P0,	BUMP-VERIFY,				!COUNT
		NEXT,	BUTD[SCOPE],				!NO ERROR:  "TEST024A" (+1, WORD]
			J/TEST024A				!    ERROR:  "LOAD023A" (-2, WORDS]
(5054)	DCS[0,00,0,1,0,1]	BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,111,001,101]






	! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

	!*** TEST 024 ***
	!TEST-024 USES A DATA PATTERN OF:  "0 000 101 111 000 110" (005706)

	! -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -

	!* PART A *
	!TEST-024-A CHECKS THAT BUT[DR6-7 L] READS THE "11" IN IR<2:1> H CORRECTLY
	!AND DOES ASSERT THE SIGNAL
	5715:
	TEST024A:
		P0,	LOAD-ENUA(ZTARGET402),			!LOAD EXPECTED ADDRESS AFTER "BUT"
			LOAD-ERROR(TEST024A),			!ERROR DIRECTORY KEY
			DCS-CTR(C4,),				!COMPARE ENUA:TNUA IN 4, MICROWORDS
		NEXT,	J/LOAD024A				!
(5715)	DCS[1,00,1,0,0,0]	BM[1011,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,111,000,110]


	5706:
	LOAD024A:
		P0,	BUMP-VERIFY,				!COUNT
		P2-U,	IR_EMIT,				!LOAD IR WITH TEST PATTERN
			EMIT/005706,				!(005706)
		NEXT,	J/GOBUT024A				!GO SETUP FOR "BUT"
(5706)	DCS[0,00,0,0,0,1]	BM[0000,,00,10,,11,11,,000,,110,,,0,0,0,,0,,0,,,1,1010,,,0,,0000,0,,,11,000,,,000,101,101]


	5055:	!(FREE)
	GOBUT024A:
		SETUP,	RETURN/TEST024B,			!RETURN TO START OF NEXT SUBTEST

		NEXT,	GOTO-PAGE(7),				!BUT TABLE IS ON PAGE 7
			J/BUTDR6-7L				!GO DO "BUT" ON DR 6/7 L
(5055)	DCS[0,00,0,0,0,0]	BM[0101,,00,11,,10,01,,001,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,100,111]


	! -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -
```

```
	;* PART B *
	;TEST-024-B CHECKS THAT BUT[INSTR1] READS THE IR CORRECTLY
	;AS CLASS=D=SOP=DMO; IR<14:06>H="000 101 111"; IR<15>H="0";
	;DM=IR<5:3>H="000", DMOH=1; AND TARGETS TO (557)
	5711:
	TEST024B:
		P0,	LOAD-ENUA(ZTARGET557),		;LOAD EXPECTED ADDRESS AFTER "BUT"
			LOAD-ERROR(TEST024B),		;ERROR DIRECTORY KEY
			DCS-CTR(C3.),			;COMPARE ENUA:THUA IN 3. MICROWORDS
		NEXT,	J/GOBUT024B			;GO SETUP FOR "BUT"
(5711)	DCS[1,00,1,0,0,0]  BM[1100,,00,11,,11,01,,101,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,101,110]


	5056:	;(FREE)
	GOBUT024B:
		SETUP,	RETURN/SCOPE024,		;RETURN TO SCOPE LOOP TEST WORD
		P0,	BUMP-VERIFY,			;COUNT
		NEXT,	GOTO-PAGE(7),			;BUT TABLE IS ON PAGE 7
			J/BUTINSTR1			;GO DO INSTR1 "BUT"
(5056)	DCS[0,00,0,0,0,1]  BM[0101,,00,00,,01,01,,111,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,000,110]


	5057:	;(FREE)
	SCOPE024:
		NEXT,	BUTD[SCOPE],			;NO ERROR:  "TEST025A" [+1, WORD]
			J/TEST025A			;    ERROR:  "LOAD024A" [-4, WORDS]
(5057)	DCS[0,00,0,1,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,111,000,111]






	;- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

	;*** TEST 025 ***
	;TEST-025 USES A DATA PATTERN OF:  "0 000 110 011 000 011" (006303)

	;-   -   -   -   -   -   -   -   -   -   -   -   -   -   -   -   -   -   -   -   -   -   -   -   -   -   -

	;* PART A *
	;TEST-025-A CHECKS THAT BUT[INSTR1] READS THE IR CORRECTLY
	;AS CLASS=D=SOP=DMO; IR<14:06>H="000 110 011"; IR<15>H="0";
	;DM=IR<5:3>H="000", DMOH=1; AND TARGETS TO (543)
	5707:
	TEST025A:
		P0,	LOAD-ENUA(ZTARGET543),		;LOAD EXPECTED ADDRESS AFTER "BUT"
			LOAD-ERROR(TEST025A).		;ERROR DIRECTORY KEY
			DCS-CTR(C4.),			;COMPARE ENUA:THUA IN 4. MICROWORDS
		NEXT,	J/LOAD025A			;GO LOAD PATTERN
(5707)	DCS[1,00,1,0,0,0]  BM[1011,,00,11,,11,01,,100,,011,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,111,000,100]
```



```
	5704:
	LOAD025A:
		P2-U,	IR_EMIT,			;LOAD IR WITH TEST PATTERN
			EMIT/006303,			;(006303)
		NEXT,	J/GOBUT025A			;GO SETUP FOR "BUT"
(5704)	DCS[0,00,0,0,0,0]  BM[0000,,00,11,,00,11,,000,,011,,,0,0,0,,0,,0,,,1,1010,,,0,,0000,0,,,11,000,,,000,110,000]


	5060:	;(FREE)
	GOBUT025A:
		SETUP,	RETURN/SCOPE025,		;RETURN TO SCOPE LOOP TEST WORD
		NEXT,	GOTO-PAGE(7),			;BUT TABLE IS ON PAGE 7
			J/BUTINSTR1			;GO DO INSTR1 "BUT"
(5060)	DCS[0,00,0,0,0,0]  BM[0101,,00,00,,01,10,,001,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,000,110]


	5061:	;(FREE)
	SCOPE025:
		NEXT,	BUTD[SCOPE],			;NO ERROR:  "TEST026A" [+1, WORD]
			J/TEST026A			;    ERROR:  "LOAD025A" [-3, WORDS]
(5061)	DCS[0,00,0,1,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,111,000,101]






	;- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

	;*** TEST 026 ***
	;TEST-026 USES A DATA PATTERN OF:  "1 000 101 100 000 011" (105403)

	;-   -   -   -   -   -   -   -   -   -   -   -   -   -   -   -   -   -   -   -   -   -   -   -   -   -   -

	;* PART A *
	;TEST-026-A CHECKS THAT BUT[INSTR1] READS THE IR CORRECTLY
	;AS CLASS=D=SOP=DMO; IR<14:06>H="000 101 100"; IR<15>H="1";
	;DM=IR<5:3>H="000", DMOH=1; AND TARGETS TO (574)
	5705:
	TEST026A:
		P0,	LOAD-ENUA(ZTARGET574),		;LOAD EXPECTED ADDRESS AFTER "BUT"
			LOAD-ERROR(TEST026A).		;ERROR DIRECTORY KEY
			DCS-CTR(C4.),			;COMPARE ENUA:THUA IN 4. MICROWORDS
			BUMP-VERIFY,			;COUNT
		NEXT,	J/LOAD026A			;GO LOAD PATTERN

(5705)	DCS[1,00,1,0,0,1]  BM[1011,,00,11,,11,01,,111,,100,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,111,000,010]


	5702:
	LOAD026A:
		P2-U,	IR_EMIT,			;LOAD IR WITH TEST PATTERN
			EMIT/105403,			;(105403)
```

```
                NEXT,   J/GOBUT026A                          ;GO SETUP FOR "BUT"
        (5702)  DCS[0,00,0,0,0,0]  BM[1000,,00,10,,11,00,,000,,011,,,0,0,0,,0,,0,,,1,1010,,,0,,0000,0,,,11,000,,,000,110,010]


        5062:   ;(FREE)
        GOBUT026A:
                SETUP,  RETURN/SCOPE026,                     ;RETURN TO SCOPE LOOP TEST WORD
                P0,     BUMP-VERIFY,                         ;COUNT
                NEXT,   GOTO-PAGE(7),                        ;BUT TABLE IS ON PAGE 7
                        J/BUTINSTR1                          ;GO DO INSTR1 "BUT"
        (5062)  DCS[0,00,0,0,0,1]  BM[0101,,00,00,,01,10,,011,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,000,110]


        5063:   ;(FREE)
        SCOPE026:
                NEXT,   BUTD[SCOPE],                         ;NO ERROR:  "TEST027A" [+1, WORD]
                        J/TEST027A                           ;   ERROR:  "LOAD026A" [-2, WORDS]
        (5063)  DCS[0,00,0,1,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,111,000,011]






        ;- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        ;*** TEST 027 ***
        ;TEST-027 USES A DATA PATTERN OF:  "1 000 110 001 000 010" (106102)

        ;  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -

        ;* PART A *
        ;TEST-027-A CHECKS THAT BUT[INSTR1] READS THE IR CORRECTLY
        ;AS CLASS-D=SOP+DM0; IR<14:06>H="000 110 001"; IR<15>H="1";
        ;DN=IR<5:3>H="000", DNOH=1; AND TARGETS TO (561)
        5703:
        TEST027A:
                P0,     LOAD-ENUA(ETARGET561),               ;LOAD EXPECTED ADDRESS AFTER "BUT"
                        LOAD-ERROR(TEST027A),                ;ERROR DIRECTORY KEY
                        DCS-CTR(C4,);                        ;COMPARE ENUA:TNUA IN 4, MICROWORDS
                NEXT,   J/LOAD027A                           ;GO LOAD PATTERN
        (5703)  DCS[1,00,1,0,0,0]  BM[1011,,00,11,,11,01,,110,,001,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,111,000,000]


        5700:
        LOAD027A:
                P2-U,   IR_EMIT,                             ;LOAD IR WITH TEST PATTERN
                        EMIT/106102,                         ;(106102)
                NEXT,   J/GOBUT027A                          ;GO SETUP FOR "BUT"
        (5700)  DCS[0,00,0,0,0,0]  BM[1000,,00,11,,00,01,,000,,010,,,0,0,0,,0,,0,,,1,1010,,,0,,0000,0,,,11,000,,,000,110,100]


        5064:   ;(FREE)
```



```
        GOBUT027A:
                SETUP,  RETURN/SCOPE027,                     ;RETURN TO SCOPE LOOP TEST WORD
                NEXT,   GOTO-PAGE(7),                        ;BUT TABLE IS ON PAGE 7
                        J/BUTINSTR1                          ;GO DO INSTR1 "BUT"
        (5064)  DCS[0,00,0,0,0,0]  BM[0101,,00,00,,01,10,,101,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,000,110]

        5065:   ;(FREE)
        SCOPE027:
                P0,     BUMP-VERIFY,                         ;COUNT
                NEXT,   BUTD[SCOPE],                         ;NO ERROR:  "TEST030A" [+1, WORD]
                        J/TEST030A                           ;   ERROR:  "LOAD027A" [-2, WORDS]
        (5065)  DCS[0,00,0,1,0,1]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,111,000,001]






        ;- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        ;*** TEST 030 ***
        ;TEST-030 USES A DATA PATTERN OF:  "0 000 101 010 000 100" (005204)

        ;  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -

        ;* PART A *
        ;TEST-030-A CHECKS THAT BUT[INSTR1] READS THE IR CORRECTLY
        ;AS CLASS-D=SOP+DM0; IR<14:06>H="000 101 010"; IR<15>H="0";
        ;DN=IR<5:3>H="000", DNOH=1; AND TARGETS TO (552)
        5701:
        TEST030A:
                P0,     LOAD-ENUA(ETARGET552),               ;LOAD EXPECTED ADDRESS AFTER "BUT"
                        LOAD-ERROR(TEST030A),                ;ERROR DIRECTORY KEY
                        DCS-CTR(C4,);                        ;COMPARE ENUA:TNUA IN 4, MICROWORDS
                NEXT,   J/LOAD030A                           ;GO LOAD PATTERN
        (5701)  DCS[1,00,1,0,0,0]  BM[1011,,00,11,,11,01,,101,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,110,111,110]


        5676:
        LOAD030A:
                P2-U,   IR_EMIT,                             ;LOAD IR WITH TEST PATTERN
                        EMIT/005204,                         ;(005204)
                NEXT,   J/GOBUT030A                          ;GO SETUP FOR "BUT"
        (5676)  DCS[0,00,0,0,0,0]  BM[0000,,00,10,,10,10,,000,,100,,,0,0,0,,0,,0,,,1,1010,,,0,,0000,0,,,11,000,,,000,110,110]


        5066:   ;(FREE)
        GOBUT030A:
                SETUP,  RETURN/TEST030B,                     ;RETURN TO START OF NEXT SUBTEST
                NEXT,   GOTO-PAGE(7),                        ;BUT TABLE IS ON PAGE 7
                        J/BUTINSTR1                          ;GO DO INSTR1 "BUT"
        (5066)  DCS[0,00,0,0,0,0]  BM[0101,,00,11,,10,01,,010,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,000,110]
```

```
	! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

	;* PART B *
	;TEST-030-B CHECKS THAT BUT[IR<11>;FLTPT<3:0>] READS THE "1" IN IR<11>H CORRECTLY,
	;AND THE FLTPT DECODE ROM GETS ADDRESS (240), WHICH IS A MULF/MODE-0 INSTR,
	;DATA SHOULD BE (04)
	5712:
	TEST030B:
		P0,	LOAD-ENUA(ZTARGET424),		;LOAD EXPECTED ADDRESS AFTER "BUT"
			LOAD-ERROR(TEST030B),		;ERROR DIRECTORY KEY
			DCS-CTR(C3.),			;COMPARE ENUA:TNUA IN 3, MICROWORDS
		NEXT,	J/GOBUT030B			;
(5712)	DCS[1,00,1,0,0,0]	BM[1100,,00,11,,11,00,,010,,100,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,110,111]


	5067:	;(FREE)
	GOBUT030B:
		SETUP,	RETURN/TEST030C,		;RETURN TO START OF NEXT SUBTEST
		NEXT,	GOTO-PAGE(7),			;BUT TABLE IS ON PAGE 7
			J/BUTIR11FLTPT3-0		;GO DO "BUT" ON IR<11>H&FLTPT<3:0>H
(5067)	DCS[0,00,0,0,0,0]	BM[0101,,00,11,,10,01,,011,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,000,010]


	! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

	;* PART C *
	;TEST-030-C CHECKS THAT BUTR[IR<11>B] READS THE "1" IN IR<11>H CORRECTLY
	5713:
	TEST030C:
		P0,	LOAD-ENUA(ZTARGET403),		;LOAD EXPECTED ADDRESS AFTER "BUT"
			LOAD-ERROR(TEST030C),		;ERROR DIRECTORY KEY
			DCS-CTR(C3.),			;COMPARE ENUA:TNUA IN 3, MICROWORDS
		NEXT,	J/GOBUT030C			;GO SETUP FOR "BUT"
(5713)	DCS[1,00,1,0,0,0]	BM[1100,,00,11,,11,00,,000,,011,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,111,000]


	5070:	;(FREE)
	GOBUT030C:
		SETUP,	RETURN/TEST030D,		;RETURN TO START OF NEXT SUBTEST
		NEXT,	GOTO-PAGE(7),			;BUT TABLE IS ON PAGE 7
			J/BUTIR11B			;GO DO "BUT" ON IR<11>H
(5070)	DCS[0,00,0,0,0,0]	BM[0101,,00,11,,10,11,,100,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,100,010]




	! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

```



```
	;* PART D *
	;TEST-030-D CHECKS THAT BUT[INSTRS] READS THE IR CORRECTLY
	;AS ROM ADDRESS=(452) ON THE INSTRS E78 ROM, AND RECEIVES THE DIAGNOSTIC VALUE
	;OF (05), TARGETING TO (405) AFTER THE DECODE
	5734:
	TEST030D:
		P0,	LOAD-ENUA(ZTARGET405),		;LOAD EXPECTED ADDRESS AFTER "BUT"
			LOAD-ERROR(TEST030D),		;ERROR DIRECTORY KEY
			DCS-CTR(C3.),			;COMPARE ENUA:TNUA IN 3, MICROWORDS
		NEXT,	J/GOBUT030D			;GO SETUP FOR "BUT"
(5734)	DCS[1,00,1,0,0,0]	BM[1100,,00,11,,11,00,,000,,101,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,111,001]


	5071:	;(FREE)
	GOBUT030D:
		SETUP,	RETURN/SCOPE030,		;RETURN TO SCOPE LOOP TEST WORD
		NEXT,	GOTO-PAGE(7),			;BUT TABLE IS ON PAGE 7
			J/BUTINSTRS			;GO DO "BUT" ON INSTRS<4:0>H
(5071)	DCS[0,00,0,0,0,0]	BM[0101,,00,00,,01,11,,010,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,000,001]


	5072:	;(FREE)
	SCOPE030:
		NEXT,	PUTD[SCOPE],			;NO ERROR:  "TEST031A" [+1, WORD]
			J/TEST031A			;   ERROR:  "LOAD030A" [-8, WORDS]
(5072)	DCS[0,00,0,1,0,0]	BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,110,111,111]






	! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

	;*** TEST 031 ***
	;TEST-031 USES A DATA PATTERN OF:   "0 001 000 010 100 010" (040212)

	! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

	;* PART A *
	;TEST-031-A CHECKS THAT BUT(DMO&SMO&BYTE)
	;DM=IR<5:3>H="100", DMOH=0; SM=IR<11:9>H="000", SMOH=1; BYTE H=0
	5677:
	TEST031A:
		P0,	LOAD-ENUA(ZTARGET402),		;LOAD EXPECTED ADDRESS AFTER "BUT"
			LOAD-ERROR(TEST031A),		;ERROR DIRECTORY KEY

			DCS-CTR(C4.),			;COMPARE ENUA:TNUA IN 4, MICROWORDS
		NEXT,	J/LOAD031A			;GO LOAD PATTERN
(5677)	DCS[1,00,1,0,0,0]	BM[1011,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,110,111,010]


	5672:
	LOAD031A:
```

```
4233            P0,     BUMP=VERIFY,                            ;COUNT
4234            P2=B,   IR_EMIT,                                ;LOAD IR WITH TEST PATTERN
4235                    EMIT/010242,                            ;(010242)
4236            NEXT,   J/GOBUT031A                             ;GO SETUP FOR "BUT"
    (5672)  DCS[0,00,0,0,0,1]  BM[0001,,00,00,,00,10,,100,,010,,,0,0,0,,0,,0,,,1,1010,,,0,,0000,0,,,11,000,,,000,111,011]
4237
4238
4239    5073:   ;(FREE)
4240    GOBUT031A:
4241            SETUP,  RETURN/TEST031B,                        ;RETURN TO START OF NEXT SUBTEST
4242            NEXT,   GOTO=PAGE(7),                           ;BUT TABLE IS ON PAGE 7
4243                    J/BUTDNSMRYTE                           ;GO DO "BUT"
    (5073)  DCS[0,00,0,0,0,0]  BM[0101,,00,11,,01,11,,101,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,001,011]
4244
4245
4246    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
4247
4248    ;* PART B *
4249    ;TEST-031-A CHECKS THAT BUT[INSTR1] READS THE IR CORRECTLY
4250    ;AS CLASS-C=MOV+SNO+-DMO; IR<14:9>#="001 000"; IR<15>#="0";
4251    ;DM=IR<5:3>#="100"; AND TARGETS TO (604)
4252    5675:
4253    TEST031B:
4254            P0,     LOAD=ENUA(ZTARGET604),                  ;LOAD EXPECTED ADDRESS AFTER "BUT"
4255                    LOAD=ERROR(TEST031B),                   ;ERROR DIRECTORY KEY
4256                    DCS-CTR(C3,),                           ;COMPARE ENUA:TNUA IN 3. MICROWORDS
4257            NEXT,   J/GOBUT031B                             ;GO SETUP FOR "BUT"
    (5675)  DCS[1,00,1,0,0,0]  BM[1100,,00,11,,11,10,,000,,100,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,111,100]
4258
4259
4260    5074:   ;(FREE)
4261    GOBUT031B:
4262            SETUP,  RETURN/SCOPE031,                        ;RETURN TO SCOPE LOOP TEST WORD
4263            NEXT,   GOTO=PAGE(7),                           ;BUT TABLE IS ON PAGE 7
4264                    J/BUTINSTR1                             ;GO DO INSTR1 "BUT"
    (5074)  DCS[0,00,0,0,0,0]  BM[0101,,00,00,,01,11,,101,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,000,110]
4265
4266
4267    5075:   ;(FREE)
4268    SCOPE031:
4269            NEXT,   SUTO[$COPE],                            ;NO ERROR:  "TEST032A" (+1, WORDS)
4270                    J/TEST032A                              ;   ERROR:  "LOAD031A" (-4, WORDS)
    (5075)  DCS[0,00,0,1,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,110,111,011]
4271
4272
4273
4274
4275
4276
4277    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
4278
4279    ;*** TEST 032 ***
```



```
4280    ;TEST-032 USES A DATA PATTERN OF: "1 001 000 001 010 101" (110125)
4281
4282    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
4283
4284    ;* PART A *
4285    ;TEST-032-A CHECKS THAT BUT[MOV-DR7#IR<B13>] READS THE (-FLTPT+MOV+FLTPT+DR7) AND
4286    ;ALTERNATING PATTERN "010" IN IR<5:3># CORRECTLY
4287    5671:
4288    TEST032A:
4289            P0,     LOAD=ENUA(ZTARGET612),                  ;LOAD EXPECTED ADDRESS AFTER "BUT"
4290                    LOAD=ERROR(TEST032A),                   ;ERROR DIRECTORY KEY
4291                    DCS-CTR(C4,),                           ;COMPARE ENUA:TNUA IN 4. MICROWORDS
4292                    BUMP=VERIFY,                            ;COUNT
4293            NEXT,   J/LOAD032A                              ;GO LOAD PATTERN
    (5671)  DCS[1,00,1,0,0,1]  BM[1011,,00,11,,11,00,,001,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,110,110,110]
4294
4295
4296    5666:
4297    LOAD032A:
4298            P2=H,   IR_EMIT,                                ;LOAD IR WITH TEST PATTERN
4299                    EMIT/110125,                            ;(110125)
4300            NEXT,   J/GOBUT032A                             ;GO SETUP FOR "BUT"
    (5666)  DCS[0,00,0,0,0,0]  BM[1001,,00,00,,00,01,,010,,101,,,0,0,0,,0,,0,,,1,1010,,,0,,0000,0,,,11,000,,,000,111,110]
4301
4302
4303    5076:   ;(FREE)
4304    GOBUT032A:
4305            SETUP,  RETURN/TEST032B,                        ;RETURN TO START OF NEXT SUBTEST
4306            NEXT,   GOTO=PAGE(7),                           ;BUT TABLE IS ON PAGE 7
4307                    J/BUTMOVDR7IR5=3                        ;GO DO "BUT"
    (5076)  DCS[0,00,0,0,0,0]  BM[0101,,00,11,,01,11,,001,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,000,101]
4308
4309
4310    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
4311
4312    ;* PART B *
4313    ;TEST-032-B CHECKS THAT BUT[INSTR1] READS THE IR CORRECTLY
4314    ;AS CLASS-C=MOV+SNO+-DMO; IR<14:9>#="001 000"; IR<15>#="1";
4315    ;DM=IR<5:3>#="010"; AND TARGETS TO (612)
4316    5671:
4317    TEST032B:
4318            P0,     LOAD=ENUA(ZTARGET612),                  ;LOAD EXPECTED ADDRESS AFTER "BUT"
4319                    LOAD=ERROR(TEST032B),                   ;ERROR DIRECTORY KEY
4320                    DCS-CTR(C3,),                           ;COMPARE ENUA:TNUA IN 3. MICROWORDS
4321            NEXT,   J/GOBUT032B                             ;GO SETUP FOR "BUT"
    (5671)  DCS[1,00,1,0,0,0]  BM[1100,,00,11,,11,10,,001,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,111,111]

4322
4323
4324    5077:   ;(FREE)
4325    GOBUT032B:
4326            SETUP,  RETURN/SCOPE032,                        ;RETURN TO SCOPE LOOP TEST WORD
4327            P0,     BUMP=VERIFY,                            ;COUNT
4328            NEXT,   GOTO=PAGE(7),                           ;BUT TABLE IS ON PAGE 7
```

```
                J/BUTINSTR1                                 ;GO DO INSTR: "BUT"
(5077)  DCS[0,00,0,0,0,1]  BM[0101,,00,00,,10,00,,000,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,000,110]


    5100:  ;(FREE)
    SCOPE032:
            NEXT,   BUTD[SCOPE],                            ;NO ERROR:  "TEST033A" [+1, WORDS]
                    J/TEST033A                              ;   ERROR:  "LOAD033A" [-4, WORDS]
(5100)  DCS[0,00,0,1,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,110,110,111]






    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

    ;*** TEST 033 ***
    ;TEST-033 USES A DATA PATTERN OF:  "1 000 110 010 111 100" (108274)

    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

    ;* PART A *
    ;TEST-033-A CHECKS THAT BUT[INSTR1] READS THE IR CORRECTLY
    ;AS CLASS-B=SOP*-DMO; IR<14:08>H="000 110 0";
    ;DM=IR<5:3>H="111", DMOH=0; AND TARGETS TO (517)
    5667:
    TEST033A:
            P0,     LOAD-ENUA(ZTARGET517),                  ;LOAD EXPECTED ADDRESS AFTER "BUT"
                    LOAD-ERROR(TEST033A),                   ;ERROR DIRECTORY KEY
                    DCS-CTR(C4,),                           ;COMPARE ENUA:TNUA IN 4, MICROWORDS
                    BUMP-VERIFY,                            ;COUNT
            NEXT,   J/LOAD033A                              ;GO LOAD PATTERN
(5667)  DCS[1,00,1,0,0,1]  BM[1011,,00,11,,11,01,,001,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,110,110,100]


    5664:
    LOAD033A:
            P2-U,   IR_EMIT,                                ;LOAD IR WITH TEST PATTERN
                    EMIT/108274,                            ;(108274)
            NEXT,   J/GOBUT033A                             ;GO SETUP FOR "BUT"
(5664)  DCS[0,00,0,0,0,0]  BM[1000,,00,11,,00,10,,111,,100,,,0,0,0,,0,,0,,,1,1010,,,0,,0000,0,,,11,000,,,001,000,001]


    5101:  ;(FREE)
    GOBUT033A:
            SETUP,  RETURN/SCOPE033,                        ;RETURN TO SCOPE LOOP TEST WORD
            P0,     BUMP-VERIFY,                            ;COUNT
            NEXT,   GOTO-PAGE(7),                           ;BUT TABLE IS ON PAGE 7
                    J/BUTINSTR1                             ;GO DO INSTR1 "BUT"
(5101)  DCS[0,00,0,0,0,1]  BM[0101,,00,00,,10,00,,010,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,000,110]
```



```

    5102:  ;(FREE)
    SCOPE033:
            NEXT,   BUTD[SCOPE],                            ;NO ERROR:  "TEST034A" [+1, WORD]
                    J/TEST034A                              ;   ERROR:  "LOAD033A" [-2, WORDS]
(5102)  DCS[0,00,0,1,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,110,110,101]





    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

    ;*** TEST 034 ***
    ;TEST-034 USES A DATA PATTERN OF:  "0 000 101 001 001 010" (005112)

    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

    ;* PART A *
    ;TEST-034-A CHECKS THAT BUT[INSTR1] READS THE IR CORRECTLY
    ;AS CLASS-B=SOP*-DMO; IR<14:09>H="000 101";
    ;DM=IR<5:3>H="001", DMOH=0; AND TARGETS TO (511)
    5665:
    TEST034A:
            P0,     LOAD-ENUA(ZTARGET511),                  ;LOAD EXPECTED ADDRESS AFTER "BUT"
                    LOAD-ERROR(TEST034A),                   ;ERROR DIRECTORY KEY
                    DCS-CTR(C4,),                           ;COMPARE ENUA:TNUA IN 4, MICROWORDS
            NEXT,   J/LOAD034A                              ;GO LOAD PATTERN
(5665)  DCS[1,00,1,0,0,0]  BM[1011,,00,11,,11,01,,001,,001,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,110,110,010]


    5662:
    LOAD034A:
            P0,     BUMP-VERIFY,                            ;COUNT
            P2-U,   IR_EMIT,                                ;LOAD IR WITH TEST PATTERN
                    EMIT/005112,                            ;(005112)
            NEXT,   J/GOBUT034A                             ;GO SETUP FOR "BUT"
(5662)  DCS[0,00,0,0,0,1]  BM[0000,,00,10,,10,01,,001,,010,,,0,0,0,,0,,0,,,1,1010,,,0,,0000,0,,,11,000,,,001,000,011]


    5103:  ;(FREE)
    GOBUT034A:
            SETUP,  RETURN/SCOPE034,                        ;RETURN TO SCOPE LOOP TEST WORD
            NEXT,   GOTO-PAGE(7),                           ;BUT TABLE IS ON PAGE 7
                    J/BUTINSTR1                             ;GO DO INSTR1 "BUT"
(5103)  DCS[0,00,0,0,0,0]  BM[0101,,00,00,,10,00,,100,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,000,110]


    5104:  ;(FREE)
    SCOPE034:
            NEXT,   BUTD[SCOPE],                            ;NO ERROR:  "TEST035A" [+1, WORD]
                    J/TEST035A                              ;   ERROR:  "LOAD034A" [-2, WORDS]
(5104)  DCS[0,00,0,1,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,110,110,011]
```

```
;- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

;*** TEST 035 ***
;TEST-035 USES A DATA PATTERN OF:  "0 011 000 111 000 001" (030701)

;  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -

;* PART A *
;TEST-035-A CHECKS THAT BUT[INSTR1] READS THE IR CORRECTLY
;AS CLASS=A=DOP*SMO*DMO; IR<15:12>H="0011";
;SM=IR<11:9>H="000", SMOH=1; DM=IR<5:3>H="000", DMOH=1; AND TARGETS TO (443)
5663:
TEST035A:
        P0,     LOAD-ENUA(ZTARGET443),              ;LOAD EXPECTED ADDRESS AFTER "BUT"
                LOAD-ERROR(TEST035A),               ;ERROR DIRECTORY KEY
                DCS-CTR(C4,1),                      ;COMPARE ENUA:TWUA IN 4, MICROWORDS
        NEXT,   J/LOAD035A                          ;GO LOAD PATTERN
 (5663)  DCS[1,00,1,0,0,0]  BM[1011,,00,11,,11,00,,100,,011,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,110,110,000]


5660:
LOAD035A:
        P2-U,   IR_EMIT,                            ;LOAD IR WITH TEST PATTERN
                EMIT/030701,                        ;(030701)
        NEXT,   J/GOBUT035A                         ;GO SETUP FOR "BUT"
 (5660)  DCS[0,00,0,0,0,0]  BM[0011,,00,00,,01,11,,000,,001,,,0,0,0,,0,,0,,,1,1010,,,0,,0000,0,,,11,000,,,001,000,101]


5105: ;(FREE)
GOBUT035A:
        SETUP,  RETURN/SCOPE035,                    ;RETURN TO SCOPE LOOP TEST WORD
        P0,     BUMP-VERIFY,                        ;COUNT
        NEXT,   GOTO-PAGE(7),                       ;BUT TABLE IS ON PAGE 7
                J/BUTINSTR1                         ;GO DO INSTR1 "BUT"
 (5105)  DCS[0,00,0,0,0,1]  BM[0101,,00,00,,10,00,,110,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,000,110]


5106: ;(FREE)
SCOPE035:
        NEXT,   BUTD[SCOPE],                        ;NO ERROR:  "TEST036A" [+1, WORD]
                J/TEST036A                          ;   ERROR:  "LOAD035A" [-2, WORDS]
 (5106)  DCS[0,00,0,1,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,110,110,001]
```



```
;- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

;*** TEST 036 ***
;TEST-036 USES A DATA PATTERN OF:  "1 101 000 101 000 110" (150506)

;  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -

;* PART A *
;TEST-036-A CHECKS THAT BUT[INSTR1] READS THE IR CORRECTLY
;AS CLASS=A=DOP*SMO*DMO; IR<15:12>H="1101";
;SM=IR<11:9>H="000", SMOH=1; DM=IR<5:3>H="000", DMOH=1; AND TARGETS TO (455)
5661:
TEST036A:
        P0,     LOAD-ENUA(ZTARGET455),              ;LOAD EXPECTED ADDRESS AFTER "BUT"
                LOAD-ERROR(TEST036A),               ;ERROR DIRECTORY KEY
                DCS-CTR(C4,3),                      ;COMPARE ENUA:TWUA IN 4, MICROWORDS
        NEXT,   J/LOAD036A                          ;GO LOAD PATTERN
 (5661)  DCS[1,00,1,0,0,0]  BM[1011,,00,11,,11,00,,101,,101,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,110,101,110]


5656:
LOAD036A:
        P0,     BUMP-VERIFY,                        ;COUNT
        P2-U,   IR_EMIT,                            ;LOAD IR WITH TEST PATTERN
                EMIT/150506,                        ;(150506)
        NEXT,   J/GOBUT036A                         ;GO SETUP FOR "BUT"
 (5656)  DCS[0,00,0,0,0,1]  BM[1101,,00,00,,01,01,,000,,110,,,0,0,0,,0,,0,,,1,1010,,,0,,0000,0,,,11,000,,,001,000,111]


5107: ;(FREE)
GOBUT036A:
        SETUP,  RETURN/SCOPE036,                    ;RETURN TO SCOPE LOOP TEST WORD
        NEXT,   GOTO-PAGE(7),                       ;BUT TABLE IS ON PAGE 7
                J/BUTINSTR1                         ;GO DO INSTR1 "BUT"
 (5107)  DCS[0,00,0,0,0,0]  BM[0101,,00,00,,10,01,,000,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,000,110]


5110: ;(FREE)
SCOPE036:
        NEXT,   BUTD[SCOPE],                        ;NO ERROR:  "TEST037A" [+1, WORD]
                J/TEST037A                          ;   ERROR:  "LOAD036A" [-2, WORDS]
 (5110)  DCS[0,00,0,1,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,110,101,111]




;- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

;*** TEST 037 ***
```

```
4522    ;TEST-037 USES A DATA PATTERN OF:  "0 110 000 010 000 101" (060205)
4523
4524    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
4525
4526    ;* PART A *
4527    ;TEST-037-A CHECKS THAT BUT(INSTR1) READS THE IR CORRECTLY
4528    ;AS CLASS-A=DOP=3"0=DMO; IR<15:12>M="0110";
4529    ;SM=IR<11:9>M="000", SMOM=1; DM=IR<5:3>M="000", SMO=1; AND TARGETS TO (446)
4530    56579
4531    TEST037A:
4532            PO,     LOAD=ENUA(ZTARGET446),                  ;LOAD EXPECTED ADDRESS AFTER "BUT"
4533                    LOAD=ERROR(TEST037A),                   ;ERROR DIRECTORY KEY
4534                    DCS=CTR(C4,),                           ;COMPARE ENUA:TNUA IN 4, MICROWORDS
4535            NEXT,   J/LOAD037A                              ;GO LOAD PATTERN
    (56547)  DCS[1,00,1,0,0,0]  BM[1011,,00,11,,11,00,,100,,110,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,110,101,100]
4536
4537
4538    56548
4539    LOAD037A:
4540            P2=U,   IR_EMIT,                                ;LOAD IR WITH TEST PATTERN
4541                    EMIT/060205,                            ;(060205)
4542            NEXT,   J/GOBUT037A                             ;GO SETUP FOR "BUT"
    (56548)  DCS[0,00,0,0,0,0]  BM[0110,,00,00,,00,10,,000,,101,,,0,0,0,,0,,0,,,1,1010,,,0,,0000,0,,,11,000,,,001,001,001]
4543
4544
4545    51111:  ;[FREE]
4546    GOBUT037A:
4547            SETUP,  RETURN/SCOPE037,                        ;RETURN TO SCOPE LOOP TEST WORD
4548            PO,     BUMP=VERIFY,                            ;COUNT
4549            NEXT,   GOTO=PAGE(7),                           ;BUT TABLE IS ON PAGE 7
4550                    J/BUTINSTR1                             ;GO DO INSTR1 "BUT"
    (51111)  DCS[0,00,0,0,0,1]  BM[0101,,00,00,,10,01,,010,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,000,110]
4551
4552
4553    51121:  ;[FREE]
4554    SCOPE037:
4555            NEXT,   BUTO[SCOPE],                            ;NO ERROR:  "TEST040A" (+1, WORD]
4556                    J/TEST040A                              ;   ERROR:  "LOAD037A" (+2, WORDS]
    (51121)  DCS[0,00,0,1,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,110,101,101]
4557
4558
4559
4560
4561
4562
4563    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
4564
4565    ;*** TEST 040 ***
4566    ;TEST-040 USES A DATA PATTERN OF:  "0 101 000 111 111 111 " (050777)
4567
4568    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
4569
4570    ;* PART A *
```



```
4571    ;TEST-040-A CHECKS THAT BUT (IR<15:12>) READS THE
4572    ;ALTERNATING PATTERN "0101" IN IR<15:12>M CORRECTLY
4573    56550
4574    TEST040A:
4575            PO,     LOAD=ENUA(ZTARGET405),                  ;LOAD EXPECTED ADDRESS AFTER "BUT"
4576                    LOAD=ERROR(TEST040A),                   ;ERROR DIRECTORY KEY
4577                    DCS=CTR(C4,),                           ;COMPARE ENUA:TNUA IN 4, MICROWORDS
4578            NEXT,   J/LOAD040A                              ;GO LOAD PATTERN
    (56559)  DCS[1,00,1,0,0,0]  BM[1011,,00,11,,11,00,,000,,101,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,110,101,000]
4579
4580
4581    56501
4582    LOAD040A:
4583            P2=U,   IR_EMIT,                                ;LOAD IR WITH TEST PATTERN
4584                    EMIT/050777,                            ;(050777)
4585            NEXT,   J/GOBUT040A                             ;GO SETUP FOR "BUT"
    (56560)  DCS[0,00,0,0,0,0]  BM[0101,,00,00,,01,11,,111,,111,,,0,0,0,,0,,0,,,1,1010,,,0,,0000,0,,,11,000,,,001,001,011]
4586
4587
4588    51131:  ;[FREE]
4589    GOBUT040A:
4590            SETUP,  RETURN/TEST040B,                        ;RETURN TO START OF NEXT SUBTEST
4591            NEXT,   GOTO=PAGE(7),                           ;BUT TABLE IS ON PAGE 7
4592                    J/BUTIR[$=12                            ;GO DO "BUT"
    (51131)  DCS[0,00,0,0,0,0]  BM[0101,,00,11,,01,01,,011,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,000,000]
4593
4594
4595    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
4596
4597    ;* PART B *
4598    ;TEST-040-B CHECKS THAT BUT(INSTR1) READS THE IR CORRECTLY
4599    ;AS CLASS-B=DOP=-MOV=,SMO=-DMO; DM=IR<5:3>M="111";
4600    ;DOP=IR<14:12>M="101"; TARGETS TO (517)
4601    56531
4602    TEST040B:
4603            PO,     LOAD=ENUA(ZTARGET517),                  ;LOAD EXPECTED ADDRESS AFTER "BUT"
4604                    LOAD=ERROR(TEST040B),                   ;ERROR DIRECTORY KEY
4605                    DCS=CTR(C3,),                           ;COMPARE ENUA:TNUA IN 3, MICROWORDS
4606                    BUMP=VERIFY,                            ;COUNT
4607            NEXT,   J/GOBUT040B                             ;GO SETUP FOR "BUT"
    (56531)  DCS[1,00,1,0,0,1]  BM[1100,,00,11,,11,01,,001,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,001,100]
4608
4609
4610    51141:  ;[FREE]
4611    GOBUT040B:
4612            SETUP,  RETURN/SCOPE040,                        ;RETURN TO SCOPE LOOP TEST WORD

4613            NEXT,   GOTO=PAGE(7),                           ;BUT TABLE IS ON PAGE 7
4614                    J/BUTINSTR1                             ;GO DO INSTR1 "BUT"
    (51141)  DCS[0,00,0,0,0,0]  BM[0101,,00,00,,10,01,,101,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,000,110]
4615
4616
4617    51151:  ;[FREE]
```

```
        SCOPE040:
                NEXT,   BUTD[SCOPE],                            !NO ERROR:  "TEST041A" [+1, WORDS]
                        J/TEST041A                              !   ERROR:  "LOAD040A" [-4, WORDS]
    (5115)  DCS[0,00,0,1,0,0]  BN[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,110,101,001]






        ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        !*** TEST 041 ***
        !TEST-041 USES A DATA PATTERN OF:  "1 010 000 111 111 111 " (120777)

        ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        !* PART A *
        !TEST-041-A CHECKS THAT BUT[IR<15:12>] READS THE
        !ALTERNATING PATTERN "1010" IN IR<15:12>H CORRECTLY
        5651:
        TEST041A:
                P0,     LOAD-ENUA(ZTARGET412),                  !LOAD EXPECTED ADDRESS AFTER "BUT"
                        LOAD-ERROR(TEST041A),                   !ERROR DIRECTORY KEY
                        DCS-CTR(C4,),                           !COMPARE ENUA:TNUA IN 4, MICROWORDS
                NEXT,   J/LOAD041A                              !GO LOAD PATTERN
    (5651)  DCS[1,00,1,0,0,0]  BN[1011,,00,11,,11,00,,001,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,110,100,100]


        5644:
        LOAD041A:
                P0,     BUMP-VERIFY,                            !COUNT
                P2-U,   IR_EMIT,                                !LOAD IR WITH TEST PATTERN
                        EMIT/120777,                            !(120777)
                NEXT,   J/GOBUT041A                             !GO SETUP FOR "BUT"
    (5644)  DCS[0,00,0,0,0,1]  BN[1010,,00,00,,01,11,,111,,111,,,0,0,0,,0,,0,,,1,1010,,,0,,0000,0,,,11,000,,,001,001,110]


        5116:  !(FREE)
        GOBUT041A:
                SETUP,  RETURN/TEST041B,                        !RETURN TO START OF NEXT SUBTEST
                NEXT,   GOTO-PAGE(7),                           !BUT TABLE IS ON PAGE 7
                        J/BUTIR15-12                            !GO DO "BUT"
    (5116)  DCS[0,00,0,0,0,0]  BN[0101,,00,11,,01,00,,111,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,000,000]


        ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        !* PART B *
        !TEST041-B CHECKS THAT BUT[INSTR1] READS THE IR CORRECTLY
        !AS CLASS-B=DOP*-MOV*SMO*-DMO; DM=IR<5:3>H="111";
        !DOP=IR<14:12>H="010"; TARGETS TO (517)
        5647:
```



```
        TEST041B:
                P0,     LOAD-ENUA(ZTARGET517),                  !LOAD EXPECTED ADDRESS AFTER "BUT"
                        LOAD-ERROR(TEST041B),                   !ERROR DIRECTORY KEY
                        DCS-CTR(C3,),                           !COMPARE ENUA:TNUA IN 3, MICROWORDS
                        BUMP-VERIFY,                            !COUNT
                NEXT,   J/GOBUT041B                             !GO SETUP FOR "BUT"
    (5647)  DCS[1,00,1,0,0,1]  BN[1100,,00,11,,11,01,,001,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,001,111]


        5117:  !(FREE)
        GOBUT041B:
                SETUP,  RETURN/SCOPE041,                        !RETURN TO SCOPE LOOP TEST WORD
                NEXT,   GOTO-PAGE(7),                           !BUT TABLE IS ON PAGE 7
                        J/BUTINSTR1                             !GO DO INSTR1 "BUT"
    (5117)  DCS[0,00,0,0,0,0]  BN[0101,,00,00,,10,10,,000,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,000,110]


        5120:  !(FREE)
        SCOPE041:
                NEXT,   BUTD[SCOPE],                            !NO ERROR:  "TEST042A" (+1, WORDS]
                        J/TEST042A                              !   ERROR:  "LOAD041A" [-4, WORDS]
    (5120)  DCS[0,00,0,1,0,0]  BN[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,110,100,101]






        ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        !*** TEST 042 ***
        !TEST-042 USES A DATA PATTERN OF:  "1 100 100 000 010 000" (144020)

        ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        !* PART A *
        !TEST-042-A CHECKS THAT BUT[DMO*SMO*BYTE]
        !DM=IR<5:3>H="010", DMOH=0; SM=IR<11:9>H="100", SMOH=0; BYTE H=1
        5645:
        TEST042A:
                P0,     LOAD-ENUA(ZTARGET401),                  !LOAD EXPECTED ADDRESS AFTER "BUT"
                        LOAD-ERROR(TEST042A),                   !ERROR DIRECTORY KEY
                        DCS-CTR(C4,),                           !COMPARE ENUA:TNUA IN 4, MICROWORDS
                NEXT,   J/LOAD042A                              !GO LOAD PATTERN
    (5645)  DCS[1,00,1,0,0,0]  BN[1011,,00,11,,11,00,,000,,001,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,110,100,000]


        5640:
        LOAD042A:
                P2-U,   IR_EMIT,                                !LOAD IR WITH TEST PATTERN
                        EMIT/144020,                            !(144020)
                NEXT,   J/GOBUT042A                             !GO SETUP FOR "BUT"
    (5640)  DCS[0,00,0,0,0,0]  BN[1100,,00,10,,00,00,,010,,000,,,0,0,0,,0,,0,,,1,1010,,,0,,0000,0,,,11,000,,,001,010,001]
```

```
4715
4716
4717    5121:  ;(FREE)
4718    GOBUT042A:
4719            SETUP,  RETURN/TEST0428,                        ;RETURN TO START OF NEXT SUBTEST
4720            NEXT,   GOTO-PAGE(7),                           ;BUT TABLE IS ON PAGE 7
4721                    J/BUTDMSMBYTE                           ;GO DO "BUT"
    (5121)  DCS[0,00,0,0,0,0]  BM[0101,,00,11,,01,00,,011,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,001,011]
4722
4723
4724    | - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
4725
4726    |* PART B *
4727    |TEST-042-B CHECKS THAT BUT[INSTR1] READS THE IR CORRECTLY
4728    |AS CLASS-G=DOP=-SM0; DOP=IR<14:12>=#"100";
4729    |SM=IR<11:9>=#"100"; TARGET# TO (714)
4730    5643:
4731    TEST0428:
4732            P0,     LOAD-ENUA(ZTARGET714),                  ;LOAD EXPECTED ADDRESS AFTER "BUT"
4733                    LOAD-ERROR(TEST0428),                   ;ERROR DIRECTORY KEY
4734                    DCS-CTR(C),,                            ;COMPARE ENUA:TNUA IN 3, MICROWORDS
4735            NEXT,   J/GOBUT042B                             ;GO SETUP FOR "BUT"
    (5643)  DCS[1,00,1,0,0,0]  BM[1100,,00,11,,11,11,,001,,100,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,010,010]
4736
4737
4738    5122:  ;(FREE)
4739    GOBUT042B:
4740            SETUP,  RETURN/SCOPE042,                        ;RETURN TO SCOPE LOOP TEST WORD
4741            P0,     BUMP=VERIFY,                            ;COUNT
4742            NEXT,   GOTO-PAGE(7),                           ;BUT TABLE IS ON PAGE 7
4743                    J/BUTINSTR1                             ;GO DO INSTR1 "BUT"
    (5122)  DCS[0,00,0,0,0,1]  BM[0101,,00,00,,10,10,,011,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,000,110]
4744
4745
4746    5123:  ;(FREE)
4747    SCOPE042:
4748            NEXT,   BUTD[SCOPE],                            ;NO ERROR:  "TEST043A" [+1, WORDS]
4749                    J/TEST043A                              ;   ERROR:  "LOAD043A" [-4, WORDS]
    (5123)  DCS[0,00,0,1,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,110,100,001]
4750
4751
4752
4753
4754
4755
4756    | - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
4757
4758    |*** TEST 043 ***
4759    |TEST-043 USES A DATA PATTERN OF:  "1 110 001 111 000 111" (161707)
4760
4761    | - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
4762
```



```
4763    |* PART A *
4764    |TEST-043-A CHECKS THAT BUT[DMOS&SMOS&BYTE]
4765    |DM=IR<5:3>=#"000", DMOK=1; SM=IR<11:9>=#"001", SMOK=0; BYTE H=0
4766    5641:
4767    TEST043A:
4768            P0,     LOAD-ENUA(ZTARGET404),                  ;LOAD EXPECTED ADDRESS AFTER "BUT"
4769                    LOAD-ERROR(TEST043A),                   ;ERROR DIRECTORY KEY
4770                    DCS-CTR(C4,),                           ;COMPARE ENUA:TNUA IN 4, MICROWORDS
4771            NEXT,   J/LOAD043A                              ;GO LOAD PATTERN
    (5641)  DCS[1,00,1,0,0,0]  BM[1011,,00,11,,11,00,,000,,100,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,110,011,100]
4772
4773
4774    5634:
4775    LOAD043A:
4776            P0,     BUMP=VERIFY,                            ;COUNT
4777            P2=U,   IR_EMIT,                                ;LOAD IR WITH TEST PATTERN
4778                    EMIT/161707,                            ;(161707)
4779            NEXT,   J/GOBUT043A                             ;GO SETUP FOR "BUT"
    (5634)  DCS[0,00,0,0,0,1]  BM[1110,,00,00,,11,11,,000,,111,,,0,0,0,,0,,0,,,1,1010,,,0,,0000,0,,,11,000,,,001,010,100]
4780
4781
4782    5124:  ;(FREE)
4783    GOBUT043A:
4784            SETUP,  RETURN/TEST043B,                        ;RETURN TO START OF NEXT SUBTEST
4785            NEXT,   GOTO-PAGE(7),                           ;BUT TABLE IS ON PAGE 7
4786                    J/BUTDMSMBYTE                           ;GO DO "BUT"
    (5124)  DCS[0,00,0,0,0,0]  BM[0101,,00,11,,00,11,,111,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,001,011]
4787
4788
4789    | - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
4790
4791    |* PART B *
4792    |TEST-043-B CHECKS THAT BUT[INSTR1] READS THE IR CORRECTLY
4793    |AS CLASS-G=DOP=-SM0; DOP=IR<14:12>=#"110";
4794    |SM=IR<11:9>=#"001"; TARGET# TO (711)
4795    5637:
4796    TEST043B:
4797            P0,     LOAD-ENUA(ZTARGET711),                  ;LOAD EXPECTED ADDRESS AFTER "BUT"
4798                    LOAD-ERROR(TEST043B),                   ;ERROR DIRECTORY KEY
4799                    DCS-CTR(C3,),                           ;COMPARE ENUA:TNUA IN 3, MICROWORDS
4800            NEXT,   J/GOBUT043B                             ;GO SETUP FOR "BUT"
    (5637)  DCS[1,00,1,0,0,0]  BM[1100,,00,11,,11,11,,001,,001,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,010,101]
4801
4802
4803    5125:  ;(FREE)
4804    GOBUT043B:
4805            SETUP,  RETURN/SCOPE043,                        ;RETURN TO SCOPE LOOP TEST WORD
4806            NEXT,   GOTO-PAGE(7),                           ;BUT TABLE IS ON PAGE 7
4807                    J/BUTINSTR1                             ;GO DO INSTR1 "BUT"
    (5125)  DCS[0,00,0,0,0,0]  BM[0101,,00,00,,10,10,,110,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,000,110]
4808
4809
```

```
4810    5126:  ;(FREE)
4811    SCOPE043:
4812            P0,     BUMP-VERIFY,                    ;COUNT
4813            NEXT,   BUTD[SCOPE],                    ;NO ERROR:  "TEST044A" [+1, WORDS]
4814                    J/TEST044A                      ;   ERROR:  "LOAD043A" [-4, WORDS]
    (5126)  DCS[0,00,0,1,0,1]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,110,011,101]
4815
4816
4817
4818
4819
4820
4821
4822    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
4823
4824    ;*** TEST 044 ***
4825    ;TEST-044 USES A DATA PATTERN OF:  "0 111 110 010 101 000" (076250)
4826
4827    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
4828
4829    ;* PART A *
4830    ;TEST-044-A CHECKS THAT BUT[INSTR] READS THE IR CORRECTLY
4831    ;AS CLASS=A=XFC; IR<15:9>H="0 111 110"; AND TARGETS TO (447)
4832    5635:
4833    TEST044A:
4834            P0,     LOAD-ENUA(ZTARGET447),          ;LOAD EXPECTED ADDRESS AFTER "BUT"
4835                    LOAD-ERROR(TEST044A),           ;ERROR DIRECTORY KEY
4836                    DCS-CTRIC4,);                   ;COMPARE ENUA;TNUA IN 4, MICROWORDS
4837            NEXT,   J/LOAD044A                      ;GO LOAD PATTERN
    (5635)  DCS[1,00,1,0,0,0]  BM[1011,,00,11,,11,00,,100,,111,,,0,0,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,110,011,010]
4838
4839
4840    5632:
4841    LOAD044A:
4842            P2=U,   IR_EMIT,                        ;LOAD IR WITH TEST PATTERN
4843                    EMIT/076250,                    ;(076250)
4844            NEXT,   J/GOBUT044A                     ;GO SETUP FOR "BUT"
    (5632)  DCS[0,00,0,0,0,0]  BM[0111,,00,11,,00,10,,101,,000,,,0,0,0,,0,,,1,1010,,,0,,0000,0,,,11,000,,,001,010,111]
4845
4846
4847    5127:  ;(FREE)
4848    GOBUT044A:
4849            SETUP,  RETURN/SCOPE044,                ;RETURN TO SCOPE LOOP TEST WORD
4850            NEXT,   GOTO-PAGE(7),                   ;BUT TABLE IS ON PAGE 7
4851                    J/BUTINSTR1                     ;GO DO INSTR; "BUT"
    (5127)  DCS[0,00,0,0,0,0]  BM[0101,,00,00,,10,11,,000,,111,,,0,0,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,000,110]
4852
4853
4854    5130:  ;(FREE)
4855    SCOPE044:
4856            P0,     BUMP-VERIFY,                    ;COUNT
4857            NEXT,   BUTD[SCOPE],                    ;NO ERROR:  "TEST045A" [+1, WORD]
4858                    J/TEST045A                      ;   ERROR:  "LOAD044A" [-2, WORDS]
```



```
    (5130)  DCS[0,00,0,1,0,1]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,110,011,011]
4859
4860
4861
4862
4863
4864
4865
4866    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
4867
4868    ;*** TEST 045 ***
4869    ;TEST-045 USES A DATA PATTERN OF:  "0 000 100 011 000 000" (004300)
4870
4871    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
4872
4873    ;* PART A *
4874    ;TEST-045-A CHECKS THAT BUT[DN=SM=BYTE] READS THE IR CORRECTLY AS:
4875    ;DM M="000", DMOH=1;  SM M="100", SMOH=0;  BYTE M=1  (IR=SMAB, BORT OF)
4876    ;CHECKS THAT SWAB INSTR ASSERTS BYTE M
4877    5633:
4878    TEST045A:
4879            P0,     LOAD-ENUA(ZTARGET405),          ;LOAD EXPECTED ADDRESS AFTER "BUT"
4880                    LOAD-ERROR(TEST045A),           ;ERROR DIRECTORY KEY
4881                    DCS-CTRIC4,);                   ;COMPARE ENUA;TNUA IN 4, MICROWORDS
4882            NEXT,   J/LOAD045A                      ;GO LOAD PATTERN
    (5633)  DCS[1,00,1,0,0,0]  BM[1011,,00,11,,11,00,,000,,101,,,0,0,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,110,011,000]
4883
4884
4885    5630:
4886    LOAD045A:
4887            P2=U,   IR_EMIT,                        ;LOAD IR WITH TEST PATTERN
4888                    EMIT/004300,                    ;(004300)
4889            NEXT,   J/GOBUT045A                     ;GO SETUP FOR "BUT"
    (5630)  DCS[0,00,0,0,0,0]  BM[0000,,00,10,,00,11,,000,,000,,,0,0,0,,0,,,1,1010,,,0,,0000,0,,,11,000,,,001,011,001]
4890
4891
4892    5131:  ;(FREE)
4893    GOBUT045A:
4894            SETUP,  RETURN/SCOPE045,                ;RETURN TO SCOPE LOOP TEST WORD
4895            NEXT,   GOTO-PAGE(7),                   ;BUT TABLE IS ON PAGE 7
4896                    J/BUTDN=SM=BYTE                 ;GO DO BUT ON DN=SM=BYTE
    (5131)  DCS[0,00,0,0,0,0]  BM[010],,00,00,,10,11,,010,,11*,,,0,0,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,001,011]
4897
4898
4899    5132:  ;(FREE)
4900    SCOPE045:

4901            NEXT,   BUTD[SCOPE],                    ;NO ERROR:  "TEST046A"  [+1, WORD]
4902                    J/TEST046A                      ;   ERROR:  "LOAD045A" [-2, WORDS]
    (5132)  DCS[0,00,0,1,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,110,011,001]
4903
4904
4905
4906
```

```
	;- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

	;*** TEST 046 ***
	;TEST-046 USES A DATA PATTERN OF:  "0 000 000 010 000 000" (000200)

	; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

	;* PART A *
	;TEST-046-A CHECKS THAT BUT[INSTRS] READS THE IR CORRECTLY
	;AS ROM ADDRESS=(426) ON THE INSTRS ESS ROM, AND RECEIVES THE VALUE
	;OF (06), TARGETING TO (426) AFTER THE DECODE
	5631:
	TEST046A:
		P0,	LOAD-ENUA(ZTARGET426),			;LOAD EXPECTED ADDRESS AFTER "BUT"
			LOAD-ERROR(TEST046A),			;ERROR DIRECTORY KEY
			DCS-CTR(C4,),				;COMPARE ENUA:TNUA IN 4, MICROWORDS
		NEXT,	J/LOAD046A				;GO LOAD PATTERN
 (5631)  DCS[1,00,1,0,0,0]  BM[1011,,00,11,,11,00,,010,,110,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,101,110,000]


	5560:
	LOAD046A:
		P2-U,	IR_EMIT,				;LOAD IR WITH TEST PATTERN
			EMIT/000200,				;(000200)
		NEXT,	J/GOBUT046A				;GO SETUP FOR "BUT"
 (5560)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,10,,000,,000,,,0,0,0,,0,,0,,,1,1010,,,0,,0000,0,,,11,000,,,001,011,011]


	5133:  ;(FREE)
	GOBUT046A:
		SETUP,	RETURN/TEST046B,			;RETURN TO START OF NEXT SUBTEST
		NEXT,	GOTO-PAGE(7),				;BUT TABLE IS ON PAGE 7
			J/BUTINSTRS				;GO DO INSTRS "BUT"
 (5133)  DCS[0,00,0,0,0,0]  BM[0101,,00,11,,11,11,,000,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,000,001]




	; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

	;* PART B *
	;TEST-046-B CHECKS THAT BUT[IR<11>!FLTPT<3:0>] READS THE "0" IN IR<11>M CORRECTLY,
	;AND THE FLTPT DECODE ROM GETS ADDRESS (040), WHICH IS A STST INSTR,
	;DATA SHOULD BE (01)
	5770:
	TEST046B:
		P0,	LOAD-ENUA(ZTARGET401),			;LOAD EXPECTED ADDRESS AFTER "BUT"
			LOAD-ERROR(TEST046B),			;ERROR DIRECTORY KEY
			DCS-CTR(C3,),				;COMPARE ENUA:TNUA IN 3, MICROWORDS
		NEXT,	J/GOBUT046B				;
 (5770)  DCS[1,00,1,0,0,0]  BM[1100,,00,11,,11,00,,000,,001,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,011,100]
```



```
	5134:  ;(FREE)
	GOBUT046B:
		SETUP,	RETURN/SCOPE046,			;RETURN TO SCOPE LOOP TEST WORD
		NEXT,	GOTO-PAGE(7),				;BUT TABLE IS ON PAGE 7
			J/BUTIR11FLTPT3-0			;GO DO "BUT" ON IR<11>M!FLTPT<3:0>M
 (5134)  DCS[0,00,0,0,0,0]  BM[0101,,00,00,,10,11,,101,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,000,010]




	5135:  ;(FREE)
	SCOPE046:
		NEXT,	BUTD[SCOPE],				;NO ERROR:  "TEST047A" [+1, WORD]
			J/TEST047A				;   ERROR:  "LOAD046A" [-4, WORDS]
 (5135)  DCS[0,00,0,1,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,101,110,001]






	;- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

	;*** TEST 047 ***
	;TEST-047 USES A DATA PATTERN OF:  "1 010 010 101 110 110" (122566)

	; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

	;* PART A *
	;TEST-047-A CHECKS THAT BUT[INSTRS] READS THE IR CORRECTLY
	;AS ROM ADDRESS=(325) ON THE INSTRS 676 ROM, AND RECEIVES THE DIAGNOSTIC VALUE
	;OF (12), TARGETING TO (412) AFTER THE DECODE
	5561:
	TEST047A:
		P0,	LOAD-ENUA(ZTARGET412),			;LOAD EXPECTED ADDRESS AFTER "BUT"
			LOAD-ERROR(TEST047A),			;ERROR DIRECTORY KEY
			DCS-CTR(C4,),				;COMPARE ENUA:TNUA IN 4, MICROWORDS
			BUMP-VERIFY,				;COUNT
		NEXT,	J/LOAD047A				;GO LOAD PATTERN
 (5561)  DCS[1,00,1,0,0,1]  BM[1011,,00,11,,11,00,,001,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,101,101,000]


	5550:
	LOAD047A:
		P2-U,	IR_EMIT,				;LOAD IR WITH TEST PATTERN

			EMIT/122566,				;(122566)
		NEXT,	J/GOBUT047A				;GO SETUP FOR "BUT"
 (5550)  DCS[0,00,0,0,0,0]  BM[1010,,00,01,,01,01,,110,,110,,,0,0,0,,0,,0,,,1,1010,,,0,,0000,0,,,11,000,,,001,011,110]


	5136:  ;(FREE)
	GOBUT047A:
```

```
        SETUP,  RETURN/TEST047B,                        ;RETURN TO START OF NEXT SUBTEST
        NEXT,   GOTO-PAGE(7),                           ;BUT TABLE IS ON PAGE 7
                J/BUTINSTRS                             ;GO DO INSTRS "BUT"
(5136)  DCS[0,00,0,0,0,0]  BM[0101,,00,11,,01,11,,000,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,000,001]




;- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

;* PART B *
;TEST-047-B CHECKS THAT BUT[IR<11>!FLTPT<3:0>] READS THE "0" IN IR<11>H CORRECTLY,
;AND THE FLTPT DECODE ROM GETS ADDRESS (534), WHICH IS A LOAD/MODE-6 INSTR,
;DATA SHOULD BE (13)
5670:
TEST047B:
        P0,     LOAD-ENUA(%TARGET413),                  ;LOAD EXPECTED ADDRESS AFTER "BUT"
                LOAD-ERROR(TEST047B),                   ;ERROR DIRECTORY KEY
                DCS-CTR(C3,),                           ;COMPARE ENUA:THUA IN 3, MICROWORDS
        NEXT,   J/GOBUT047B                             ;
(5670)  DCS[1,00,1,0,0,0]  BM[1100,,00,11,,11,00,,001,,011,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,011,111]


5137:   ;(FREE)
GOBUT047B:
        P0,     BUMP-VERIFY,                            ;COUNT
        SETUP,  RETURN/TEST047C,                        ;RETURN TO START OF NEXT SUBTEST
        NEXT,   GOTO-PAGE(7),                           ;BUT TABLE IS ON PAGE 7
                J/BUTIR11FLTPT3=0                       ;GO DO "BUT" ON IR<11>H&FLTPT<3:0>H
(5137)  DCS[0,00,0,0,0,1]  BM[0101,,00,11,,10,11,,000,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,000,010]


;- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

;* PART C *
;TEST-047-C CHECKS THAT BUT[MOV-DR7&IR<5:3>] READS THE =(-FLTPT*MOV+FLTPT*DR7), IR<5:3>="110"
5730:
TEST047C:
        P0,     LOAD-ENUA(%TARGET406),                  ;LOAD EXPECTED ADDRESS AFTER "BUT"
                LOAD-ERROR(TEST047C),                   ;ERROR DIRECTORY KEY
                DCS-CTR(C3,),                           ;COMPARE ENUA:THUA IN 3, MICROWORDS
        NEXT,   J/GOBUT047C                             ;GO SETUP FOR "BUT"
(5730)  DCS[1,00,1,0,0,0]  BM[1100,,00,11,,11,00,,000,,110,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,100,000]


5140:   ;(FREE)
GOBUT047C:
        SETUP,  RETURN/SCOPE047,                        ;RETURN TO SCOPE LOOP TEST WORD
        NEXT,   GOTO-PAGE(7),                           ;BUT TABLE IS ON PAGE 7
                J/BUTMOVDR7IR5=3                        ;GO DO "BUT" ON (MOV/DR7)&IR<5:3>H
(5140)  DCS[0,00,0,0,0,0]  BM[0101,,00,00,,11,00,,001,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,000,101]
```



```




5141:   ;(FREE)
SCOPE047:
        NEXT,   BUTD[SCOPE],                            ;NO ERROR:  "TEST050A" [+1, WORD]
                J/TEST050A                              ;   ERROR:  "LOAD047A" [-6, WORDS]
(5141)  DCS[0,00,0,1,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,101,101,001]





;- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

;*** TEST 050 ***
;TEST-050-A USES A DATA PATTERN OF:  "1 111 010 101 000 000" (172500)
;TEST-050-B USES A DATA PATTERN OF:  "1 111 101 010 000 000" (175200)

;- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

;* PART A *
;TEST-050-A CHECKS THAT BUT[INSTR5] READS THE IR CORRECTLY
;AS ROM ADDRESS=(725) ON THE INSTR5 E78 ROM, AND RECEIVES THE DIAGNOSTIC VALUE
;OF (05), TARGETING TO (405) AFTER THE DECODE
5551:
TEST050A:
        P0,     LOAD-ENUA(%TARGET405),                  ;LOAD EXPECTED ADDRESS AFTER "BUT"
                LOAD-ERROR(TEST050A),                   ;ERROR DIRECTORY KEY
                DCS-CTR(C4,),                           ;COMPARE ENUA:THUA IN 4, MICROWORDS
                BUMP-VERIFY,                            ;COUNT
        NEXT,   J/LOAD050A                              ;GO LOAD PATTERN
(5551)  DCS[1,00,1,0,0,1]  BM[1011,,00,11,,11,00,,000,,101,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,101,101,010]


5552:
LOAD050A:
        P2-U,   IR_EMIT,                                ;LOAD IR WITH TEST PATTERN
                EMIT/172500,                            ;(172500)
        NEXT,   J/GOBUT050A                             ;GO SETUP FOR "BUT"
(5552)  DCS[0,00,0,0,0,0]  BM[1111,,00,01,,01,01,,000,,000,,,0,0,0,,0,,0,,,1,1010,,,0,,0000,0,,,11,000,,,001,100,010]


5142:   ;(FREE)
GOBUT050A:

        SETUP,  RETURN/TEST050B,                        ;RETURN TO START OF NEXT SUBTEST
        NEXT,   GOTO-PAGE(7),                           ;BUT TABLE IS ON PAGE 7
                J/BUTINSTR5                             ;GO DO INSTR5 "BUT"
(5142)  DCS[0,00,0,0,0,0]  BM[0101,,00,11,,10,10,,110,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,000,001]
```

```
5100
5101    ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
5102
5103    !* PART B *
5104    !TEST-050=8 CHECKS THAT BUT(INSTR=8) READS THE IR CORRECTLY
5105    !AS ROM ADDR=(752) ON THE INSTRS E78 ROM, AND RECEIVES THE DIAGNOSTIC VALUE
5106    !OF (12), TARGETING TO (412) AFTER THE DECODE
5107    5726:
5108    TEST050B:
5109            P0,     LOAD=ENUA(ZTARGET412),                  !LOAD EXPECTED ADDRESS AFTER "BUT"
5110                    LOAD=ERROR(TEST0308),                   !ERROR DIRECTORY KEY
5111                    DCS-CTR(C4.),                           !COMPARE ENUA:TNUA IN 4. MICROWORDS
5112            NEXT,   J/LOAD050B                              !
    (5726)  DCS[1,00,1,0,0,0]  BM[1011,,00,11,,11,00,,001,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,100,011]
5113
5114
5115    5143:  !(FREE)
5116    LOAD050B:
5117            P2=U,   IR_EMIT,                                !LOAD IR WITH TEST PATTERN
5118                    EMIT/175200,                            !(175200)
5119            NEXT,   J/GOBUT050B                             !GO SETUP FOR "BUT"
    (5143)  DCS[0,00,0,0,0,0]  BM[1111,,00,10,,10,10,,000,,000,,,0,0,0,,0,,0,,,1,1010,,,0,,0000,0,,,11,000,,,001,100,100]
5120
5121    5144:  !(FREE)
5122    GOBUT050B:
5123            SETUP,  RETURN/SCOPE050,                        !RETURN TO SCOPE LOOP TEST WORD
5124            NEXT,   GOTO-PAGE(7),                           !BUT TABLE IS ON PAGE 7
5125                    J/BUTINSTRS                             !DO INSTRS BUT
    (5144)  DCS[0,00,0,0,0,0]  BM[0101,,00,00,,11,00,,101,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,000,001]
5126
5127
5128    5145:  !(FREE)
5129    SCOPE050:
5130            P0,     BUSDIN_EMIT=(I),                        !RESET PROC UCON
5131                    IN-CLK-IR(15=00),                       !
5132            NEXT,   BUT0(SCOPE),                            !NO ERROR:  "TEST101A"  (+1, WORD)
5133                    J/TEST101A                              !   ERROR:  "LOAD050A" (-8, WORDS)
    (5145)  DCS[0,00,0,1,0,0]  BM[0000,,00,00,,00,01,,000,,100,,,0,0,0,,0,,0,,,1,1001,,,0,,0000,0,,,11,000,,,101,101,011]
5134
5135
5136
5137    1.PAGE**********************************************************************************************************************
5138
5139    .TOC    *   TEST101:  D => DBUF => IR PATH
5140
5141
5142    !*********************************************************************************
5143    !*
5144    !*      TESTS:  101 - 104                               UWORDS:  057 + 067
5145    !*
5146    !*      FUNCTIONS:
5147    !*
5148    !*      THESE TESTS VERIFY THE "EMIT => CSP => BBUS => D => DBUF => IR" DATAPATH.
```



```
5149    !*
5150    !*      TEST 101 FIRST VERIFIES THE "D => DBUF => IR" DATA PATH, INSURING THAT THE
5151    !*      DBUF LATCH CAN BE WRITTEN WITH ZEROES, AND ENABLED ONTO BUSDIN, TO BE PUT
5152    !*      INTO THE IR AND VERIFIED (VIA INSTRS DECODE, AS A HALT INSTRUCTION).
5153    !*
5154    !*      TESTS 102-104 THEN GO ON TO FURTHER TEST THE FULL DATAPATH FROM EMIT TO
5155    !*      IR, VIA THE EXTENDED ROUTE.  THESE TESTS THEN VERIFY THE CSP WRITE, ADDRESSING AND
5156    !*      DATAPATHS LOGIC.
5157    !*
5158    !*********************************************************************************
5159
5160    5553:
5161    TEST101A:
5162            P0,     LOAD=ENUA(ZTARGET434),                  !INSTRS E48 OUTPUT FOR IR=(000000)
5163                    LOAD=ERROR(TEST101A),                   !ERROR DIRECTORY KEY
5164                    DCS-CTR(C7.),                           !COMPARE AT TARGET
5165            NEXT,   J/LOAD0101A                             !
    (5553)  DCS[1,00,1,0,0,0]  BM[1000,,00,11,,11,00,,011,,100,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,101,010,000]
5166
5167    5520:
5168    LOAD0101A:
5169            P2=T,   D_ZERO,                                 !PUT (000000) IN D
5170            NEXT,   J/GOTEST101A                            !
    (5520)  DCS[0,00,0,0,0,0]  BM[0011,,00,00,,00,00,,000,,000,,,0,1,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,100,111]
5171
5172    5147:  !(FREE)
5173    GOTEST101A:
5174            SETUP,  RETURN/SCOPE101,                        !GOT TO SUBR THAT:
5175            NEXT,   CALL[DINTOIR=5]                         !D => DBUF => IR, THEN BUT(INSTR5)
    (5147)  DCS[0,00,0,0,0,0]  BM[0101,,00,00,,11,01,,000,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,111,011]
5176
5177
5178
5179    5150:  !(FREE)
5180    SCOPE101:
5181            NEXT,   BUT0[SCOPE],                            !NO ERROR:  "TEST102A"  (+1, WORDS)
5182                    J/TEST102A                              !   ERROR:  "LOADCSP101A"  (-3, WORDS)
    (5150)  DCS[0,00,0,1,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,101,010,001]
5183
5184
5185    ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
5186
5187
5188    .TOC    *   TEST102-104:  TESTING CSP ADDRESS/READ/WRITE FUNCTIONS
5189
5190    !THE FOLLOWING SET OF FOUR TESTS VERIFIES THAT THE CSP, AND THE CSP ADDRESS FIELD "CSPADDR"

5191    !HAS NO STUCK ZERO BITS, AND THAT THE EMIT => CSP => D => DBUF => IR DATAPATH
5192    !IS FULLY FUNCTIONAL.
5193    !

5194    !       AFTER TEST 104B COMPLETES, THE CSP WILL LOOK AS FOLLOWS:
5195    !
5196    !       "MAS=CON"       "CSP=ADR"       CSP             INSTRS
5197    !       "C41140>H       U<23120>H       LOCT    -DATA-  TARGET
```

```
5198    !       ----------      ----------      ----    ------  ------
5199    !                         1111          00      000000  E78/434
5200    !                         1110          01      000182  E78/425
5201    !                         1101          02      000125  E78/432
5202    !                         1100          03      ******
5203    !                         1011          04      125200  E88/412
5204    !                         1010          05      ******
5205    !                         1001          06      ******
5206    !                         1000          07      152500  E88/405
5207    !                         0111          10      125200  E88/405
5208    !                         0110          11      ******
5209    !                         0101          12      ******
5210    !                         0100          13      125200  E88/412
5211    !           11            0011          14      ******
5212    !           10            0010          15      000125  E78/432
5213    !           01            0001          16      000182  E78/425
5214    !           00            0000          17      000000  E78/434
5215    !
5216
5217
5218    ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
5219
5220
5221    !TEST 102A VERIFIES THAT CSPD(02) WAS WRITTEN WITH THE UNIQUE INSTR8 PATTERN:
5222    !  (000125); E78 TARGET (432),  LOOKING FOR CSP ADDRESS BIT<01> STUCK ONE/ZERO,
5223    !  OR ERRORS IN DATAPATH FROM EMIT -> CSP -> ALU-B -> D -> DBUF -> IR.
5224    5521:
5225    TEST102A:
5226            P0,     LOAD=ENUA(ETARGET432),                  !INSTR8 E78 OUTPUT
5227                    LOAD=ERROR(TEST102A),                   !ERROR DIRECTORY KEY
5228                    DCS=CTR(C13,),                          !COMPARE AT TARGET
5229            NEXT,   J/LOAD01=102A                           !
    (5521)  DCS[1,00,1,0,0,0]  BM[0011..00.11..11.00..011..010...0,0,0..0..0...0.0000...0..0000.0...11.000...101.011.000]
5230
5231    5530:
5232    LOAD01=102A:
5233            P3,     CSPD[01]_EMIT, EMIT/000182,             !INSTR8 DATA PATTERN:
5234            NEXT,   J/LOAD02=102A                           !  (000182)=E78(425)
    (5530)  DCS[0,00,0,0,0,0]  BM[0000..10.00..00.01..101..010...0,0,0..0..0...0.1110...1..0000.0...11.000...001.101.001]
5235
5236    5141:  !(FREE)
5237    LOAD02=102A:
5238            P3,     CSPD[02]_EMIT, EMIT/000125,             !INSTR8 DATA PATTERN:
5239            NEXT,   J/LOAD04=102A                           !  (000125)=E78(432)
    (5141)  DCS[0,00,0,0,0,0]  BM[0000..10.00..00.01..010..101...0,0,0..0..0...0.1101...1..0000.0...11.000...001.101.010]
5240
5241    5152:  !(FREE)
5242    LOAD04=102A:
5243            P3,     CSPD[04]_EMIT, EMIT/125200,             !INSTR8 DATA PATTERN:
5244            NEXT,   J/LOAD10=102A                           !  (125200)=E88(412)
    (5152)  DCS[0,00,0,0,0,0]  BM[1010..10.10..10.10..000..000...0,0,0..0..0...0.1011...1..0000.0...11.000...001.101.011]
5245
5246    5153:  !(FREE)
5247    LOAD10=102A:
```



```
5248            P3,     CSPD[10]_EMIT, EMIT/152500,             !INSTR8 DATA PATTERN:
5249            NEXT,   J/LOAD00=102A                           !  (152500)=E88(405)
    (5153)  DCS[0,00,0,0,0,0]  BM[1101..10.01..01.01..000..000...0,0,0..0..0...0.0111...1..0000.0...11.000...001.101.100]
5250
5251    5154:  !(FREE)
5252    LOAD00=102A:
5253            P3,     CSPD[00]_EMIT, EMIT/000000,             !INSTR8 DATA PATTERN:
5254                                                            !  (000000)=E78(434)
5255            NEXT,   J/LOAD00102A                            !IF THIS DATA USED, CSP ADDRESSING ERROR
    (5154)  DCS[0,00,0,0,0,0]  BM[0000..10.00..00.00..000..000...0,0,0..0..0...0.1111...1..0000.0...11.000...001.101.101]
5256
5257    5155:  !(FREE)
5258    LOAD00102A:
5259            P2=T,   D_CSPD(D02), BSEL/B17,                  !GET CSP LOC VIA CSPADDR, BASCON FIELD "00"
5260            NEXT,   J/GOTEST102A                            !
    (5155)  DCS[0,00,0,0,0,0]  BM[1010..10.00..00.00..000..000...0,1,0..0..0...0.1101...0..0000.0...11.000...001.101.110]
5261
5262    5156:  !(FREE)
5263    GOTEST102A:
5264            SETUP,  RETURN/TEST102B,                        !GO TO SUBR WHICH:
5265            NEXT,   CALL[DINTOIR=5]                         ! D => DBUF => IR, THEN BUT(INSTR5)
    (5156)  DCS[0,00,0,0,0,0]  BM[0101..00.10..11.11..100..111...0,0,0..0..0...0.0000...0..0000.0...11.100...010.111.011]
5266
5267
5268
5269
5270
5271    ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
5272
5273    !TEST 102B VERIFIES THAT CSPD(10) WAS WRITTEN WITH THE UNIQUE INSTR8 PATTERN:
5274    !  (152500), E88 TARGET (405),  LOOKING FOR CSP ADDRESS BIT<00> STUCK ONE/ZERO,
5275    !  OR ERRORS IN DATAPATH FROM EMIT -> CSP -> ALU-B -> D -> DBUF -> IR.
5276    5574:
5277    TEST102B:
5278            P0,     LOAD=ENUA(ETARGET405),                  !INSTR8 E88 OUTPUT
5279                    LOAD=ERROR(TEST102B),                   !ERROR DIRECTORY KEY
5280                    DCS=CTR(C7,),                           !COMPARE AT TARGET
5281            NEXT,   J/LOAD00102B                            !
    (5574)  DCS[1,00,1,0,0,0]  BM[1000..00.11..11.00..000..101...0,0,0..0..0...0.0000...0..0000.0...11.000...001.101.111]
5282
5283    5157:  !(FREE)
5284    LOAD00102B:
5285            P2=T,   D_CSPD(D10), BSEL/B17,                  !GET CSP LOC VIA CSPADDR, BASCON FIELD "00"
5286            NEXT,   J/GOTEST102B                            !
    (5157)  DCS[0,00,0,0,0,0]  BM[1010..10.00..00.00..000..000...0,1,0..0..0...0.0111...0..0000.0...11.000...001.110.000]
5287

5288    5160:  !(FREE)
5289    GOTEST102B:
5290            SETUP,  RETURN/TEST102C,                        !GO TO SUBR WHICH:
5291            NEXT,   CALL[DINTOIR=5]                         ! D => DBUF => IR, THEN BUT(INSTR5)
    (5160)  DCS[0,00,0,0,0,0]  BM[0101..00.10..11.10..100..111...0,0,0..0..0...0.0000...0..0000.0...11.100...010.111.011]
5292
```

```
5293
5294
5295
5296
5297    ! * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - *
5298
5299    !TEST 102C VERIFIES THAT CSPD(04) WAS WRITTEN WITH THE UNIQUE INSTRS PATTERN:
5300    !  (125200), E88 TARGET (412),  LOOKING FOR CSP ADDRESS BIT<02> STUCK ONE/ZERO,
5301    !  OR ERRORS IN DATAPATH FROM EMIT -> CSP -> ALU-B -> D -> DBUF -> IR.
5302    5564:
5303    TEST102C:
5304            P0,     LOAD-ENUA(ZTARGET412),                  !INSTRS E88 OUTPUT
5305                    LOAD-ERROR(TEST102C),                   !ERROR DIRECTORY KEY
5306                    DC8-CTR(C7,),                           !COMPARE AT TARGET
5307                    BUMP-VERIFY,                            !COUNT
5308            NEXT,   J/LOADD102C                             !
     (5564)  DC8(1,00,1,0,0,1)  BM(1000,,00,11,,11,00,,001,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,110,001)
5309
5310    5161:  !(FREE)
5311    LOADD102C:
5312            P2=T,   D_CSPD(D04), BSEL/B17,                  !GET CSP LOC VIA CSPADDR, BASCON FIELD "00"
5313            NEXT,   J/GOTEST102C                            !
     (5161)  DC8(0,00,0,0,0,0)  BM(1010,,10,00,,00,00,,000,,000,,,0,1,0,,0,,0,,,0,1011,,,0,,0000,0,,,11,000,,,001,110,010)
5314
5315    5162:  !(FREE)
5316    GOTEST102C:
5317            SETUP,  RETURN/TEST102D,                        !GO TO SUBR WHICH:
5318            NEXT,   CALL[DINTOIR=5]                         ! D -> DBUF -> IR, THEN BUT(INSTRS)
     (5162)  DC8(0,00,0,0,0,0)  BM(0101,,00,10,,11,01,,100,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,111,011)
5319
5320
5321
5322
5323
5324    ! * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - *
5325
5326    !TEST 102D VERIFIES THAT CSPD(01) WAS WRITTEN WITH THE UNIQUE INSTRS PATTERN:
5327    !  (000152), E78 TARGET (425),  LOOKING FOR CSP ADDRESS BIT<00> STUCK ONE/ZERO,
5328    !  OR ERRORS IN DATAPATH FROM EMIT -> CSP -> ALU-B -> D -> DBUF -> IR.
5329    5554:
5330    TEST102D:
5331            P0,     LOAD-ENUA(ZTARGET425),                  !INSTRS E88 OUTPUT
5332                    LOAD-ERROR(TEST102D),                   !ERROR DIRECTORY KEY
5333                    DC8-CTR(C7,),                           !COMPARE AT TARGET
5334            NEXT,   J/LOADD102D                             !
     (5554)  DC8(1,00,1,0,0,0)  BM(1000,,00,11,,11,00,,010,,101,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,110,011)
5335
5336    5163:  !(FREE)
5337    LOADD102D:
5338            P2=T,   D_CSPD(D01), BSEL/B17,                  !GET CSP LOC VIA CSPADDR, BASCON FIELD "00"
5339            NEXT,   J/GOTEST102D                            !
     (5163)  DC8(0,00,0,0,0,0)  BM(1010,,10,00,,00,00,,000,,000,,,0,1,0,,0,,0,,,0,1110,,,0,,0000,0,,,11,000,,,001,110,100)
5340
```



```
5341    5164:  !(FREE)
5342    GOTEST102D:
5343            SETUP,  RETURN/SCOPE102,                        !GO TO SUBR WHICH:
5344            NEXT,   CALL[DINTOIR=5]                         ! D -> DBUF -> IR, THEN BUT(INSTRS)
     (5164)  DC8(0,00,0,0,0,0)  BM(0101,,00,00,,11,10,,101,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,111,011)
5345
5346
5347
5348
5349    5165:  !(FREE)
5350    SCOPE102:
5351            NEXT,   BUTD(SCOPE),                            !NO ERROR:  "TEST103A" (+1, WORDS)
5352                    J/TEST103A                              !   ERROR:  "LOAD01=102A"  (-16, WORDS)
     (5165)  DC8(0,00,0,1,0,0)  BM(0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,101,011,001)
5353
5354
5355    ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
5356
5357
5358    !THE FOLLOWING SET OF FOUR TESTS VERIFIES THAT THE CSP, AND THE CSP ADDRESS FIELD "CSPADDR"
5359    !HAS NO STUCK ONE BITS, AND THAT THE EMIT -> CSP -> D -> DBUF -> IR DATAPATH
5360    !IS FULLY FUNCTIONAL.
5361
5362
5363    ! * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - *
5364
5365
5366    !TEST 103A VERIFIES THAT CSPD(13) WAS WRITTEN WITH THE UNIQUE INSTRS PATTERN:
5367    !  (125200), E88 TARGET (412),  LOOKING FOR CSP ADDRESS BIT<02> STUCK ONE/ZERO,
5368    !  OR ERRORS IN DATAPATH FROM EMIT -> CSP -> ALU-B -> D -> DBUF -> IR.
5369    5531:
5370    TEST103A:
5371            P0,     LOAD-ENUA(ZTARGET412),                  !INSTRS E88 OUTPUT
5372                    LOAD-ERROR(TEST103A),                   !ERROR DIRECTORY KEY
5373                    DC8-CTR(C12,),                          !COMPARE AT TARGET
5374            NEXT,   J/LOAD16-103A                           !
     (5531)  DC8(1,00,1,0,0,0)  BM(0011,,00,11,,11,00,,001,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,101,100,110)
5375
5376    5546:
5377    LOAD16-103A:
5378            P0,     BUMP-VERIFY,                            !COUNT
5379            P3,     CSPD(16)_EMIT, EMIT/000152,             !INSTRS DATA PATTERN:
5380            NEXT,   J/LOAD15-103A                           !  (000152)=E78(425)
     (5546)  DC8(0,00,0,0,0,1)  BM(0000,,10,00,,00,01,,101,,010,,,0,0,0,,0,,0,,,0,0001,,,1,,0000,0,,,11,000,,,001,110,110)
5381
5382    5166:  !(FREE)
5383
5384    LOAD15-103A:
5385            P3,     CSPD(15)_EMIT, EMIT/000125,             !INSTRS DATA PATTERN:
5386            NEXT,   J/LOAD13-103A                           !  (000125)=E78(432)
     (5166)  DC8(0,00,0,0,0,0)  BM(0000,,10,00,,00,01,,010,,101,,,0,0,0,,0,,0,,,0,0010,,,1,,0000,0,,,11,000,,,001,110,111)
5387
5388    5167:  !(FREE)
5389    LOAD13-103A:
```

```
5388            P1,     CSPD[13]_EMIT, EMIT/128200,             ;INSTRS DATA PATTERN:
5390            NEXT,   J/LOAD07-103A                           ;  (128200)=XB80(412)
    (5167)  DCS[0,00,0,0,0,0]  BM[1010,,10,1D,,10,10,,000,,000...0,0,0,,0,,0,,0,,0,0160...1,,0000,0,,,11,000,,,001,111,000]
5391
5392    5170:  ;(FREE)
5393    LOAD07-103A:
5394            P3,     CSPD[07]_EMIT, EMIT/192800,             ;INSTRS DATA PATTERN:
5395            NEXT,   J/LOAD17-103A                           ;  (192800)=XE80(400)
    (5170)  DCS[0,00,0,0,0,0]  BM[1101,,10,01,,01,01,,000,,000...0,0,0,,0,,0,,0,,0,1000...1,,0000,0,,,11,000,,,001,111,001]
5396
5397    5171:  ;(FREE)
5398    LOAD17-103A:
5399            P3,     CSPD[17]_EMIT, EMIT/000000,             ;INSTRS DATA PATTERN:
5400                                                            ;  (000000)=X70(434)
5401            NEXT,   J/LOADD103A                             ;IF THIS DATA USED, CSP ADDRESSING ERROR
    (5171)  DCS[0,00,0,0,0,0]  BM[0000,,10,00,,00,00,,000,,000...0,0,0,,0,,0,,0,,0,0000...1,,0000,0,,,11,000,,,001,111,010]
5402
5403    5172:  ;(FREE)
5404    LOADD103A:
5405            P2=T,   D_CSPD(D13), BSEL/B17,                  ;GET CSP LOC VIA CSPADDR, BASCON FIELD "00"
5406            NEXT,   J/GOTEST103A                            ;
    (5172)  DCS[0,00,0,0,0,0]  BM[1010,,10,00,,00,00,,000,,000...0,1,0,,0,,0,,0,,0,0100...0,,0000,0,,,11,000,,,001,111,011]
5407
5408    5173:  ;(FREE)
5409    GOTEST103A:
5410            SETUP,  RETURN/TEST103B,                        ;GO TO SUBR WHICH:
5411            NEXT,   CALL[DINTOIR=5]                         ; D => DBUF => IR, THEN OUT(INSTRS)
    (5173)  DCS[0,00,0,0,0,0]  BM[0101,,00,10,,11,11,,010,,111...0,0,0,,0,,0,,0,,0,0000,,,0,,0000,0,,,11,100,,,010,111,011]
5412
5413
5414
5415
5416
5417    ;  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -
5418
5419    ;TEST 103B VERIFIES THAT CSPD(15) WAS WRITTEN WITH THE UNIQUE INSTRS PATTERN:
5420    ;  (000125), E78 TARGET (432).  LOOKING FOR CSP ADDRESS BIT<01> STUCK ONE/ZERO,
5421    ;  OR ERRORS IN DATAPATH FROM EMIT => CSP => ALU=B => D => DBUF => IR.
5422    5572:
5423    TEST103B:
5424            P0,     LOAD=ENUA(ZTARGET432),                  ;INSTRS E78 OUTPUT
5425                    LOAD=ERROR(TEST103B),                   ;ERROR DIRECTORY KEY
5426                    DCS-CTR(C?,),                           ;COMPARE AT TARGET
5427            NEXT,   J/LOADD103B                             ;
    (5572)  DCS[1,00,1,0,0,0]  BM[1000,,00,11,,11,00,,011,,010...0,0,0,,0,,0,,0,,0,0000...0,,0000,0,,,11,000,,,001,111,100]
5428
5429    5174:  ;(FREE)
5430    LOADD103B:
5431            P2=T,   D_CSPD(D15), BSEL/B17,                  ;GET CSP LOC VIA CSPADDR, BASCON FIELD "00"
5432            NEXT,   J/GOTEST103B                            ;
    (5174)  DCS[0,00,0,0,0,0]  BM[1010,,10,00,,00,00,,000,,000...0,1,0,,0,,0,,0,,0,0010...0,,0000,0,,,11,000,,,001,111,101]
5433
5434    5175:  ;(FREE)
```



```
5435    GOTEST103B:
5436            SETUP,  RETURN/TEST103C,                        ;GO TO SUBR WHICH:
5437            NEXT,   CALL[DINTOIR=5]                         ; D => DBUF => IR, THEN OUT(INSTRS)
    (5175)  DCS[0,00,0,0,0,0]  BM[0101,,00,10,,11,11,,110,,111...0,0,0,,0,,0,,0,,0,0000...0,,0000,0,,,11,100,,,010,111,011]
5438
5439
5440
5441
5442
5443    ;  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -
5444
5445    ;TEST 103C VERIFIES THAT CSPD(16) WAS WRITTEN WITH THE UNIQUE INSTRS PATTERN:
5446    ;  (000152), E78 TARGET (425).  LOOKING FOR CSP ADDRESS BIT<00> STUCK ONE/ZERO,
5447    ;  OR ERRORS IN DATAPATH FROM EMIT => CSP => ALU=B => D => DBUF => IR.
5448    5576:
5449    TEST103C:
5450            P0,     LOAD=ENUA(ZTARGET425),                  ;INSTRS E78 OUTPUT
5451                    LOAD=ERROR(TEST103C),                   ;ERROR DIRECTORY KEY
5452                    DCS-CTR(C7,),                           ;COMPARE AT TARGET
5453                    BUMP=VERIFY,                            ;COUNT
5454            NEXT,   J/LOADD103C                             ;
    (5576)  DCS[1,00,1,0,0,1]  BM[1000,,00,11,,11,00,,010,,101...0,0,0,,0,,0,,0,,0,0000...0,,0000,0,,,11,000,,,001,111,110]
5455
5456    5176:  ;(FREE)
5457    LOADD103C:
5458            P2=T,   D_CSPD(D16), BSEL/B17,                  ;GET CSP LOC VIA CSPADDR, BASCON FIELD "00"
5459            NEXT,   J/GOTEST103C                            ;
    (5176)  DCS[0,00,0,0,0,0]  BM[1010,,10,00,,00,00,,000,,000...0,1,0,,0,,0,,0,,0,0001...0,,0000,0,,,11,000,,,001,111,111]
5460
5461    5177:  ;(FREE)
5462    GOTEST103C:
5463            SETUP,  RETURN/TEST103D,                        ;GO TO SUBR WHICH:
5464            NEXT,   CALL[DINTOIR=5]                         ; D => DBUF => IR, THEN OUT(INSTRS)
    (5177)  DCS[0,00,0,0,0,0]  BM[0101,,00,11,,01,11,,100,,111...0,0,0,,0,,0,,0,,0,0000...0,,0000,0,,,11,100,,,010,111,011]
5465
5466
5467
5468
5469
5470    ;  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -
5471
5472    ;TEST 103D VERIFIES THAT CSPD(07) WAS WRITTEN WITH THE UNIQUE INSTRS PATTERN:
5473    ;  (152800), E88 TARGET (405).  LOOKING FOR CSP ADDRESS BIT<03> STUCK ONE/ZERO,
5474    ;  OR ERRORS IN DATAPATH FROM EMIT => CSP => ALU=B => D => DBUF => IR.
5475    5674:
5476    TEST103D:

5477            P0,     LOAD=ENUA(ZTARGET405),                  ;INSTRS E88 OUTPUT
5478                    LOAD=ERROR(TEST103D),                   ;ERROR DIRECTORY KEY
5479                    DCS-CTR(C7,),                           ;COMPARE AT TARGET

5480            NEXT,   J/LOADD103D                             ;
    (5674)  DCS[1,00,1,0,0,0]  BM[1000,,00,11,,11,00,,000,,101...0,0,0,,0,,0,,0,,0,0000...0,,0000,0,,,11,000,,,010,000,000]
5481
```

```
    5200: ;(FREE)
    LOADD103D:
            P2-T,   D_CSPD(D07), BSEL/B17,                    ;GET CSP LOC VIA CSPADDR, BASCON FIELD "00"
            NEXT,   J/GOTEST103D                              ;
  (5200)  DCS[0,00,0,0,0,0]  BM[1010,,10,00,,00,00,,000,,000,,,0,1,0,,0,,0,,,0,1000,,,0,,0000,0,,,11,000,,,010,000,001]

    5201: ;(FREE)
    GOTEST103D:
            SETUP,  RETURN/SCOPE103,                          ;GO TO SUBR WHICH:
            NEXT,   CALL[DINTOIR-5]                           ; D => DBUF => IR, THEN BUT(INSTRS)
  (5201)  DCS[0,00,0,0,0,0]  BM[0101,,00,01,,00,00,,010,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,111,011]




    5202: ;(FREE)
    SCOPE103:
            NEXT,   BUTD[SCOPE],                              ;NO ERROR:  "TEST104A" (+1, WORDS)
                    J/TEST104A                                ;   ERROR:  "LOAD16-103A"  (-16, WORDS)
  (5202)  DCS[0,00,0,1,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,101,100,111]


    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

    ;THE FOLLOWING SET OF TWO TESTS VERIFIES THAT THE CSP, AND THE CSP ADDRESS FIELD "BASCON"
    ;HAS NO STUCK ZERO/ONE BITS, AND THAT THE EMIT => CSP => D => DBUF => IR DATAPATH
    ;IS FULLY FUNCTIONAL.


    ;  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -


    ;TEST 104A VERIFIES THAT CSPB(16) WAS WRITTEN WITH THE UNIQUE INSTRS PATTERN:
    ;  (000152), E78 TARGET (425).  LOOKING FOR CSP ADDRESS BIT<00> STUCK ONE/ZERO,
    ;  OR ERRORS IN DATAPATH FROM EMIT => CSP => ALU-B => D => DBUF => IR.
    ;THIS TEST USES THE "BASCON" ADDRESS MODE FOR THE CSP.
    5547:
    TEST104A:
            P0,     LOAD-ENUA[ZTARGET425],                    ;INSTRS E78 OUTPUT
                    LOAD-ERROR(TEST104A),                     ;ERROR DIRECTORY KEY
                    DCS-CTR(C7,),                             ;COMPARE AT TARGET
            NEXT,   J/LOADD104A                               ;
  (5547)  DCS[1,00,1,0,0,0]  BM[1000,,00,11,,11,00,,010,,101,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,101,100,000]

    5540:
    LOADD104A:
            P2-T,   D_CSPB(B16), CSPADDR/D17,                 ;GET CSP LOC VIA BASCON, CSPADDR FIELD "0000"
            NEXT,   J/GOTEST104A                              ;
  (5540)  DCS[0,00,0,0,0,0]  BM[1010,,11,01,,00,00,,000,,000,,,0,1,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,010,000,011]

    5203: ;(FREE)
    GOTEST104A:
            SETUP,  RETURN/TEST104B,                          ;GO TO SUBR WHICH:
```



```
            NEXT,   CALL[DINTOIR-5]                           ; D => DBUF => IR, THEN BUT(INSTRS)
  (5203)  DCS[0,00,0,0,0,0]  BM[0101,,00,11,,01,00,,110,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,111,011]





    ;  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -

    ;TEST 104B VERIFIES THAT CSPB(15) WAS WRITTEN WITH THE UNIQUE INSTRS PATTERN:
    ;  (000125), E78 TARGET (432).  LOOKING FOR CSP ADDRESS BIT<01> STUCK ONE/ZERO,
    ;  OR ERRORS IN DATAPATH FROM EMIT => CSP => ALU-B => D => DBUF => IR.
    ;THIS TEST USES THE "BASCON" ADDRESS MODE FOR THE CSP.
    5646:
    TEST104B:
            P0,     LOAD-ENUA[ZTARGET432],                    ;INSTRS E78 OUTPUT
                    LOAD-ERROR(TEST104B),                     ;ERROR DIRECTORY KEY
                    DCS-CTR(C7,),                             ;COMPARE AT TARGET
            NEXT,   J/LOADD104B                               ;
  (5646)  DCS[1,00,1,0,0,0]  BM[1000,,00,11,,11,00,,011,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,010,000,100]

    5204: ;(FREE)
    LOADD104B:
            P2-T,   D_CSPB(B15), CSPADDR/D17,                 ;GET CSP LOC VIA BASCON, CSPADDR FIELD "0000"
            NEXT,   J/GOTEST104B                              ;
  (5204)  DCS[0,00,0,0,0,0]  BM[1010,,11,10,,00,00,,000,,000,,,0,1,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,010,000,101]

    5205: ;(FREE)
    GOTEST104B:
            SETUP,  RETURN/SCOPE104,                          ;GO TO SUBR WHICH:
            NEXT,   CALL[DINTOIR-5]                           ; D => DBUF => IR, THEN BUT(INSTRS)
  (5205)  DCS[0,00,0,0,0,0]  BM[0101,,00,01,,00,00,,110,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,111,011]




    5206: ;(FREE)
    SCOPE104:
            P0,     BUSDIN_EMIT-[I],                          ;RESET PROC UCON
                    EN-CLK-IR[15-00],                         ;
            NEXT,   BUTD[SCOPE],                              ;NO ERROR:  "TEST105A" (+1, WORDS)
                    J/TEST105A                                ;   ERROR:  "LOADD104A"  (-5, WORDS)
  (5206)  DCS[0,00,0,1,0,0]  BM[0000,,00,00,,00,01,,000,,100,,,0,0,0,,0,,0,,,1,1001,,,0,,0000,0,,,11,000,,,101,100,001]



    ;.PAGE=============================================================================================

    .TOC    *   TEST105:  SR CAN LOAD/STORE AS A REGISTER

    ;*****************************************************************************
    ;*
```

```
        !*      TESTS:  105                                              UWORDS:  033 + 032
        !*
        !*      FUNCTIONS:
        !*
        !*      THE FOLLOWING TESTS VERIFY THE VALIDITY OF THE SR AS A TEMPORARY REGISTER,
        !*      (IE, IT CAN BE LOADED/READ) IN ALL BIT POSITIONS, AND THAT THE ALU-A SIDE CAN
        !*      PASS DATA INTO D.
        !*
        !*****************************************************************************************


        !TEST 105A VERIFIES THAT THE SR CAN BE LOADED/READ WITH THE INSTRS PATTERN:
        !  (000125), E78 TARGET (432), AND TO VERIFY THE:
        !  DATAPATH FROM SR => ALU-A => D -> DBUF => IR.
        5541:
        TEST105A:
                PO,     LOAD=ENUA(ZTARGET432),                   !INSTRS E78 OUTPUT
                        LOAD-ERROR(TEST105A),                    !ERROR DIRECTORY KEY
                        DCS-CTR(C8,),                            !COMPARE AT TARGET
                        BUMP=VERIFY,                             !COUNT
                NEXT,   J/LOADSR105A                             !
 (5541)  DCS[1,00,1,0,0,1]  BN[0111,,00,11,,11,00,,011,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,101,100,010]

          5542:
        LOADSR105A:
                P2,     RES_CSPD(002),                           !BITS<13:11>=00/0, WHICH IS;  SR/LOAD, GUARD/DISABLED
                P3-T,   SR_CSPD(D02), BSEL/B17,                  !DATA IS (000125) = INSTRS E78 (432)
                NEXT,   J/GOTEST105A                             !
 (5542)  DCS[0,00,0,0,0,0]  BN[1010,,10,00,,00,00,,000,,000,,,1,0,1,,0,,0,,,0,1101,,,0,,1000,1,,,11,000,,,010,000,111]

          5207:  !(FREE)
        GOTEST105A:
                PO,     BUMP-VERIFY,                             !COUNT
                SETUP,  RETURN/TEST105A1,                        !GO TO SUBR WHICH:
                NEXT,   CALL[SRINTOIR-5]                         ! SR => D => DBUF => IR, THEN BUT(INSTRS)
 (5207)  DCS[0,00,0,0,0,1]  BN[0101,,00,11,,00,01,,000,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,111,010]





        !  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -

        !TEST 105A1 VERIFIES THAT THE BUT(SR3-0) SEES THE "0101" IN THE SR.
        5610:
        TEST105A1:
                PO,     LOAD=ENUA(ZTARGET405),                   !BIT<3:0> = "0101"
                        LOAD-ERROR(TEST105A1),                   !ERROR DIRECTORY KEY
                        DCS-CTR(C3,),                            !COMPARE AT TARGET
                NEXT,   J/GOBUT105A1                             !
 (5610)  DCS[1,00,1,0,0,0]  BN[1100,,00,11,,11,00,,000,,101,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,010,001,000]

          5210:  !(FREE)
```



```
        GOBUT105A1:
                PO,     BUMP-VERIFY,                             !COUNT
                SETUP,  RETURN/TEST105B,                         !RETURN TO START OF NEXT SUBTEST
                NEXT,   GOTO-PAGE(7),                            !BUT TABLE
                        J/BUTSR3-0                               !SR<3:0> IN BIT<3:0>
 (5210)  DCS[0,00,0,0,0,1]  BN[0101,,00,11,,01,01,,010,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,111,110]





        !  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -

        !TEST 105B VERIFIES THAT THE SR CAN BE LOADED/READ WITH THE INSTRS PATTERN:
        !  (000152), E78 TARGET (425), AND TO VERIFY THE:
        !  DATAPATH FROM SR -> ALU-A => D -> DBUF => IR.
        5652:
        TEST105B:
                PO,     LOAD-ENUA(ZTARGET425),                   !INSTRS E78 OUTPUT
                        LOAD-ERROR(TEST105B),                    !ERROR DIRECTORY KEY
                        DCS-CTR(C8,),                            !COMPARE AT TARGET
                NEXT,   J/LOADSR105B                             !
 (5652)  DCS[1,00,1,0,0,0]  BN[0111,,00,11,,11,00,,010,,101,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,010,001,001]

          5211:  !(FREE)
        LOADSR105B:
                P2-T,   SR_CSPD(D01), BSEL/B17,                  !DATA IS (000152) = INSTRS E78 (425)
                NEXT,   J/GOTEST105B                             !
 (5211)  DCS[0,00,0,0,0,0]  BN[1010,,10,00,,00,00,,000,,000,,,0,0,1,,0,,0,,,0,1110,,,0,,0000,0,,,11,000,,,010,001,010]

          5212:  !(FREE)
        GOTEST105B:
                SETUP,  RETURN/TEST105B1,                        !GO TO SUBR WHICH:
                NEXT,   CALL[SRINTOIR-5]                         ! SR => D => DBUF => IR, THEN BUT(INSTRS)
 (5212)  DCS[0,00,0,0,0,0]  BN[0101,,00,11,,00,00,,000,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,111,010]





        !  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -

        !TEST 105B1 VERIFIES THAT THE BUT(SR3-0) SEES THE "1010" IN THE SR.
        5600:
        TEST105B1:
                PO,     LOAD-ENUA(ZTARGET412),                   !BIT<3:0> = "1010"

                        LOAD-ERROR(TEST105B1),                   !ERROR DIRECTORY KEY
                        DCS-CTR(C3,),                            !COMPARE AT TARGET
                        BUMP-VERIFY,                             !COUNT
                NEXT,   J/GOBUT105B1                             !
 (5600)  DCS[1,00,1,0,0,1]  BN[1100,,00,11,,11,00,,001,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,010,001,011]
```

```
5673    5213:  ;(FREE)
5674    GOBUT105B1:
5675            SETUP,  RETURN/TEST105C,                        ;RETURN TO START OF NEXT SUBTEST
5676            NEXT,   GOTO=PAGE(7),                           ;BUT TABLE
5677                    J/BUTBR3=0                              ;BR<3:0> IN BIT<3:0>
    (5213)  DCS[0,00,0,0,0,0]  BM[0101,,00,11,,01,00,,010,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,111,110]
5678
5679
5680
5681
5682
5683
5684    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
5685
5686    ;TEST 105C VERIFIES THAT THE SR CAN BE LOADED/READ WITH THE INSTR5 PATTERN:
5687    ;  (125200), E86 TARGET (412), AND TO VERIFY THE:
5688    ;  DATAPATH FROM SR => ALU=A => D => DBUF => IR.
5689    5642:
5690    TEST105C:
5691            P0,     LOAD=ENUA(ZTARGET412),                  ;INSTR5 E86 OUTPUT
5692                    LOAD=ERROR(TEST105C),                   ;ERROR DIRECTORY KEY
5693                    DC5=CTR(C8,),                           ;COMPARE AT TARGET
5694            NEXT,   J/LOADSR105C                            ;
    (5642)  DCS[1,00,1,0,0,0]  BM[0111,,00,11,,11,00,,001,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,010,001,100]
5695
5696    5214:  ;(FREE)
5697    LOADSR105C:
5698            P2=T,   SR_CSPD(004), BSEL/B17,                 ;DATA IS (125200) = INSTR5 E86 (412)
5699            NEXT,   J/GOTEST105C                            ;
    (5214)  DCS[0,00,0,0,0,0]  BM[1010,,10,00,,00,00,,000,,000,,,0,0,1,,0,,0,,,0,1011,,,0,,0000,0,,,11,000,,,010,001,101]
5700
5701    5215:  ;(FREE)
5702    GOTEST105C:
5703            SETUP,  RETURN/TEST105D,                        ;GO TO SUBR WHICH:
5704            NEXT,   CALL[SRINTOIR=5]                        ; SR => D => DBUF => IR, THEN BUT[INSTR5]
    (5215)  DCS[0,00,0,0,0,0]  BM[0101,,00,11,,00,11,,110,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,111,010]
5705
5706
5707
5708
5709
5710    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
5711
5712    ;TEST 105D VERIFIES THAT THE SR CAN BE LOADED/READ WITH THE INSTR5 PATTERN:
5713    ;  (152500), E86 TARGET (405), AND TO VERIFY THE:
5714    ;  DATAPATH FROM SR => ALU=A => D => DBUF => IR.
5715    5636:
5716    TEST105D:
5717            P0,     LOAD=ENUA(ZTARGET405),                  ;INSTR5 E86 OUTPUT
5718                    LOAD=ERROR(TEST105D),                   ;ERROR DIRECTORY KEY
5719                    DC5=CTR(C9,),                           ;COMPARE AT TARGET
5720            NEXT,   J/LOADSR105D                            ;
```



```
    (5636)  DCS[1,00,1,0,0,0]  BM[0111,,00,11,,11,00,,000,,101,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,010,001,110]
5721
5722    5216:  ;(FREE)
5723    LOADSR105D:
5724            P0,     BUMP=VERIFY,                            ;COUNT
5725            P2=T,   SR_CSPD(D10), BSEL/B17,                 ;DATA IS (152500) = INSTR5 E86 (405)
5726            NEXT,   J/GOTEST105D                            ;
    (5216)  DCS[0,00,0,0,0,1]  BM[1010,,10,00,,00,00,,000,,000,,,0,0,1,,0,,0,,,0,0111,,,0,,0000,0,,,11,000,,,010,001,111]
5727
5728    5217:  ;(FREE)
5729    GOTEST105D:
5730            SETUP,  RETURN/TEST105E,                        ;GO TO SUBR WHICH:
5731            NEXT,   CALL[SRINTOIR=5]                        ; SR => D => DBUF => IR, THEN BUT[INSTR5]
    (5217)  DCS[0,00,0,0,0,0]  BM[0101,,00,11,,00,10,,000,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,111,010]
5732
5733
5734
5735
5736
5737    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
5738
5739    ;TEST 105E VERIFIES THE DATAPATH FROM:
5740    ;  EMIT => CSP => ALU=B => SR => ALU=A => D => DBUF => IR,
5741    ;  IS A VALID "1101" IN AIPS<15:12>,  SR VALUE = (152500) FROM PREVIOUS TEST.
5742    5620:
5743    TEST105E:
5744            P0,     LOAD=ENUA(ZTARGET415),                  ;BIT<15:12> = "1101"
5745                    LOAD=ERROR(TEST105E),                   ;ERROR DIRECTORY KEY
5746                    DC5=CTR(C3,),                           ;COMPARE AT TARGET
5747            NEXT,   J/GOTEST105E                            ;
    (5620)  DCS[1,00,1,0,0,0]  BM[1100,,00,11,,11,00,,001,,101,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,010,010,000]
5748
5749    5220:  ;(FREE)
5750    GOTEST105E:
5751            SETUP,  RETURN/SCOPE105,                        ;RETURN TO SCOPE LOOP TEST WORD
5752            NEXT,   GOTO=PAGE(7),                           ;BUT TABLE
5753                    J/BUTIR15-12                            ;BIT<15:12> TEST
    (5220)  DCS[0,00,0,0,0,0]  BM[0101,,00,01,,00,10,,001,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,000,000]
5754
5755
5756
5757    5221:  ;(FREE)
5758    SCOPE105:
5759            P0,     BUSDIN_EMIT=(1),                        ;RESET PROC UCON
5760                    EN-CLK-IR[15=00],                       ;
5761            NEXT,   BUTD[SCOPE],                            ;NO ERROR:  "SETUPCSP17A" (+1, WORDS)
5762                                                            ;
5763                    J/SETUPCSP17A                           ;   ERROR:  "LOADSR105A"  (-17, WORDS)
    (5221)  DCS[0,00,0,1,0,0]  BM[0000,,00,00,,00,01,,000,,100,,,0,0,0,,0,,0,,,1,1001,,,0,,0000,0,,,11,000,,,101,100,011]
5764
5765
5766
```

```
;.PAGE=========================================================================================================================

.TOC    *    TEST114-121:  ALU LOGIC TESTS / D[C] TESTS

;**************************************************************************************************
;*
;*      TESTS:  114 - 121                                        UWORDS:  250 + 300
;*
;*      FUNCTIONS:
;*
;*      THESE TESTS TEST THE ALU LOGIC FUNCTIONS.
;*
;**************************************************************************************************



;
;       SUMMARY OF ALU LOGIC / D[C] TESTS:
;
;       TEST            ALU             OPERANDS                D[C]
;       NUMB            FUNCTION         A/B=D                  FUNCTION
;       ----            --------        --------                --------
;
;       114A            ZERO             1/1=0                  CIN=PS[C]=0
;
;       115A            NOT-A            1/1=0                  CIN=1
;       115B            NOT-A            0/1=1                  PS[C]=0
;       115C            NOT-A            0/0=1
;       115D            NOT-A            1/0=0
;
;       116A            NOT-A-AND-B      0/1=1                  ALU15=1, D[C]=1
;       116B            ZERO             1/0=0
;       116C            NOT-A-AND-B      0/0=0                  ALU15=0, D[C]=0
;       116D            ZERO             0/1=0
;
;       117A            A-AND-NOT-B      1/0=1                  CIN=D[C]=0
;       117B            A-AND-NOT-B      1/1=0                  (ALU15=1)
;       117C            A-AND-NOT-B      0/0=0                  CIN=D[C]=1
;
;       120A            A-AND-B          0/1=0
;       120B            A-AND-B          1/1=1                  CIN=0
;
;       121A            A-XOR-B          0/0=0                  ALU00=1
;       121B            A-XOR-B          1/0=1                  ALU00=0
;       121C            A-XOR-B          0/1=1                  ALU07=1
;       121D            A-XOR-B          1/1=0                  ALU07=0
;
;       122A            A-IOR-B          0/0=0                  CIN=1
;       122A3           BUT(D<14-00>=ZERO*D15) W/ D=(000000)
;       122A4           BUT(D<14-00>=ZERO*D15) W/ D=(125252)
;
```



```
;

;FOR THE ALU LOGIC TESTS FOLLOWING, THE REQUIRED CONSTANTS
;IN THE CSP ARE:
5543:
SETUPCSP17A:
        P3,     CSPD[15]_EMIT, EMIT/000077,                     ;MASK FOR BITS<05:00>
        NEXT,   GOTO-PAGE(7),                                   ;XFR
                J/SETUPCSP16A                                   ;
(5543)  DCS[0,00,0,0,0,0]  BM[0000,,10,00,,00,00,,111,,111,,,0,0,0,,0,,0,,,0,0010,,,1,,0000,0,,,11,100,,,000,000,010]

7002:  ;(FREE)
SETUPCSP16A:
        P3,     CSPD[16]_EMIT, EMIT/170000,                     ;BITS<15:12> SET
        NEXT,   J/SETUPCSP15A                                   ;
(7002)  DCS[0,00,0,0,0,0]  BM[1111,,10,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0001,,,1,,0000,0,,,11,000,,,000,001,000]

7010:  ;(FREE)
SETUPCSP15A:
        P3,     CSPD[17]_EMIT, EMIT/007700,                     ;MASK FOR BITS<11:06>
        NEXT,   J/SETUPCSP14A                                   ;
(7010)  DCS[0,00,0,0,0,0]  BM[0000,,10,11,,11,11,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,1,,0000,0,,,11,000,,,000,001,001]

7011:  ;(FREE)
SETUPCSP14A:
        P3,     CSPD[14]_EMIT, EMIT/000100,                     ;BIT<06> SET
        NEXT,   J/SETUPCSP12A                                   ;
(7011)  DCS[0,00,0,0,0,0]  BM[0000,,10,00,,00,01,,000,,000,,,0,0,0,,0,,0,,,0,0011,,,1,,0000,0,,,11,000,,,000,001,010]

7012:  ;(FREE)
SETUPCSP12A:
        P3,     CSPD[11]_EMIT, EMIT/125252,                     ;PATTERN: "1010 1010 1010 1010"
        NEXT,   J/SETUPCSP05A                                   ;
(7012)  DCS[0,00,0,0,0,0]  BM[1010,,10,10,,10,10,,101,,010,,,0,0,0,,0,,0,,,0,0110,,,1,,0000,0,,,11,000,,,000,001,011]

7013:  ;(FREE)
SETUPCSP05A:
        P3,     CSPD[10]_EMIT, EMIT/052525,                     ;PATTERN: "0101 0101 0101 0101"
        NEXT,   J/SETUPCSP07A                                   ;
(7013)  DCS[0,00,0,0,0,0]  BM[0101,,10,01,,01,01,,010,,101,,,0,0,0,,0,,0,,,0,0111,,,1,,0000,0,,,11,000,,,000,001,100]

7014:  ;(FREE)
SETUPCSP07A:
        P3,     CSPD[12]_EMIT, EMIT/177777,                     ;PATTERN: "1111 1111 1111 1111 "
        NEXT,   J/SETUPCSP00A                                   ;
(7014)  DCS[0,00,0,0,0,0]  BM[1111,,10,11,,11,11,,111,,111,,,0,0,0,,0,,0,,,0,0101,,,1,,0000,0,,,11,000,,,000,001,101]


7015:  ;(FREE)
SETUPCSP00A:
        P3,     CSPD[13]_EMIT, EMIT/000000,                     ;PATTERN: "0000 0000 0000 0000"
        NEXT,   GOTO-PAGE(0),                                   ;SAME AS (4)
                J/TEST114A                                      ;
```

```
(7015)  DCS[0,00,0,0,0,0]  BM[0000,,10,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0100,,,1,,0000,0,,,11,100,,,110,000,111]




        ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        ;VERIFY THAT WITH: ALU=(ZERO), A=(177777), B=(177777), THEN D=(000000), AND D[C]=CIN=PS[C]=(0)
        4607:
        TEST114A:
                PO,     LOAD-ENUA(ZTARGET434),                  ;INSTRS FOR IN=(000000)
                        LOAD-ERROR(TEST114A),                   ;ERROR DIRECTORY KEY
                        DCS-CTR(C8,);                           ;COMPARE AT TARGET
                NEXT,   J/GETONES114A                           ;
(4607)  DCS[1,00,1,0,0,0]  BM[0111,,00,11,,11,00,,011,,100,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,110,000,000]

        4600:
        GETONES114A:
                P2-T,   D_CSPD(C177777), D[C]_ALU15,            ;ALL ONES STORED HERE
                P3,     A+BSPHI[C177777]_D,                     ;WRITE INTO ASP,BSP
                NEXT,   J/ALU114A
(4600)  DCS[0,00,0,0,0,0]  BM[1010,,10,11,,00,01,,101,,100,,,0,1,0,,0,,0,,,0,0101,,,0,,1111,0,,,11,000,,,000,000,011]

        4003:  ;(FREE)
        ALU114A:
                P2-T,   D_ZERO, D[C]_CINMUX,                    ;ALU=(ZERO), D[C]=CIN=PS[C]=(0)
                        BUS-A_ASPHI(C177777),                   ;A=(177777)
                        BUS-B_CSPD(C177777),                    ;B=(177777)
                NEXT,   J/GETZEROES114A                         ;D=(000000)
(4003)  DCS[0,00,0,0,0,0]  BM[0011,,10,00,,11,01,,101,,000,,,0,1,0,,0,,0,,,0,0101,,,0,,0000,0,,,11,000,,,000,000,100]

        4004:  ;(FREE)
        GETZEROES114A:
                P3,     A+BSPHI[C000000]_D,                     ;ALL ZEROS STORED HERE IN ASP, BSP
                NEXT,   J/GOBUT114A                             ;
(4004)  DCS[0,00,0,0,0,0]  BM[0000,,00,11,,00,01,,100,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,1111,0,,,11,000,,,000,000,101]

        4005:  ;(FREE)
        GOBUT114A:
                SETUP,  RETURN/TEST114A2,                       ;EXEC SUBR WHICH:
                                                                ;(1) D -> IR
                NEXT,   CALL[DZERO]                             ;(2) BUT(INSTRS) INTO ZTARGET---
(4005)  DCS[0,00,0,0,0,0]  BM[0100,,00,11,,00,00,,011,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,100,010]

        ;CHECK THAT D[C] GOT A (0) FROM "CINMUX=PS[C]", ABOVE
        4603:
        TEST114A2:
                PO,     LOAD-ENUA(ZTARGET402),                  ;BIT <00> = D[C] = (0)
                        LOAD-ERROR(TEST114A2),                  ;ERROR DIRECTORY KEY
                        DCS-CTR(C3,),                           ;COMPARE AT TARGET
                NEXT,   J/GOBUT114A2                            ;
(4603)  DCS[1,00,1,0,0,0]  BM[1100,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,000,110]
```



```
        4006:  ;(FREE)
        GOBUT114A2:
                SETUP,  RETURN/SCOPE114A,                       ;RETURN TO SCOPE LOOP TEST WORD
                NEXT,   GOTO-PAGE(7),                           ;BUT TABLE
                        J/BUTD[C]A                              ;D[C]N IN BIT<00>
(4006)  DCS[0,00,0,0,0,0]  BM[0100,,00,00,,00,00,,111,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,100,100]


        4007:  ;(FREE)
        SCOPE114A:
                P2-T,   D_CSPD(C052525), SAVE-D[C],             ;STORE A CONSTANT
                P3,     A+BSPHI[C052525]_D,                     ;
                NEXT,   BUTD[SCOPE],                            ;NO ERROR:  "TEST115A1"   (+1, WORDS)
                        J/TEST115A1                             ;   ERROR:  "GETONES114A" (-6,WORDS)
(4007)  DCS[0,00,0,1,0,0]  BM[1010,,10,11,,00,01,,111,,111,,,0,1,0,,0,,0,,,0,0111,,,0,,1111,0,,,11,000,,,110,000,001]





        ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        ;THIS NEXT SET OF 12. TESTS EXERCISES THE "NOT-A" ALU FUNCTION

        ;  -   -   -   -   -   -   -   -   -   -   -   -   -   -   -   -   -   -   -   -   -   -   -

        ;TESTS 115A 1-3 VERIFIES THAT WITH:
        ;ALU=(NOT-A), A=(052525), B=(177777), THEN D=(125252), AND D[C]=CINMUX=(1)
        4601:
        TEST115A1:
                PO,     LOAD-ENUA(ZTARGET412),                  ;BIT<15:12> = "1010"
                        LOAD-ERROR(TEST115A1),                  ;ERROR DIRECTORY KEY
                        DCS-CTR(C7,1,                           ;COMPARE AT TARGET
                NEXT,   J/ALU115A1                              ;
(4601)  DCS[1,00,1,0,0,0]  BM[1000,,00,11,,11,00,,001,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,110,001,000]

        4610:
        ALU115A1:
                PO,     BUMP-VERIFY,                            ;COUNT
                P2-T,   D_NOT-A, D[C]_CINMUX,                   ;ALU=(NOT-A) D[C]=CINMUX=(1)
                        BUS-A_ASPHI(C052525),                   ;A=(052525)
                        BUS-B_CSPD(C177777),                    ;B=(177777)
                NEXT,   J/GETALTN115A1                          ;D=(125252)
(4610)  DCS[0,00,0,0,0,1]  BM[0000,,10,00,,11,01,,111,,000,,,0,1,0,,0,,0,,,0,0101,,,0,,0000,0,,,11,000,,,000,001,000]

        4010:  ;(FREE)

        GETALTN115A1:
                P3,     A+BSPHI[C125252]_D,                     ;STORE CONSTANT (125252), HOPEFULLY
                NEXT,   J/GOBUT115A1                            ;
(4010)  DCS[0,00,0,0,0,0]  BM[0000,,00,11,,00,01,,110,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,1111,0,,,11,000,,,000,001,001]

        4011:  ;(FREE)
        GOBUT115A1:
```

```
            SETUP,  RETURN/TEST115A4,                      ;EXEC SUBR WHICH:
                                                           ;(1) D<15:12> => IR<15:12>
            NEXT,   CALL[D15-12]                           ;(2) BUT(IR15-12) INTO ZTARGET---
  (4011)  DCS[0,00,0,0,0,0]  BM[0100,,00,11,,00,00,,100,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,011,100]


    ;CHECK THAT D(C) GOT A (1) FROM "CINMUXm(1)", ABOVE
    4604:
    TEST115A4:
            P0,     LOAD-ENUA(ZTARGET403),                 ;BIT <00> = D[C] = (1)
                    LOAD-ERROR(TEST115A4),                 ;ERROR DIRECTORY KEY
                    DCS-CTR(C3,),                          ;COMPARE AT TARGET
                    BUMP-VERIFY,                           ;COUNT
            NEXT,   J/GOBUT115A4                           ;
  (4604)  DCS[1,00,1,0,0,1]  BM[1100,,00,11,,11,00,,000,,011,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,001,010]

    4012:  ;(FREE)
    GOBUT115A4:
            SETUP,  RETURN/TEST115A2,                      ;RETURN TO START OF NEXT SUBTEST
            NEXT,   GOTO-PAGE(7),                          ;BUT TABLE
                    J/BUTD[C]A                             ;D[C]M IN BIT<00>
  (4012)  DCS[0,00,0,0,0,0]  BM[0100,,00,11,,00,01,,110,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,100,100]


    ;CHECK BIT<11:06> = "1010 10"
    4616:
    TEST115A2:
            P0,     LOAD-ENUA(ZTARGET412),                 ;INSTRS-ETS OUTPUT FOR BIT <11:06>="101 010"
                    LOAD-ERROR(TEST115A2),                 ;ERROR DIRECTORY KEY
                    DCS-CTR(C6,),                          ;COMPARE AT TARGET
                    BUMP-VERIFY,                           ;COUNT
            NEXT,   J/GOBUT115A2                           ;
  (4616)  DCS[1,00,1,0,0,1]  BM[1001,,00,11,,11,00,,001,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,001,011]

    4013:  ;(FREE)
    GOBUT115A2:
            SETUP,  RETURN/TEST115A3,                      ;EXEC SUBR WHICH:
                                                           ;(1) D<11:06> => IR<11:06>
            NEXT,   CALL[D11-06]                           ;(2) BUT(INSTRS) INTO ZTARGET---
  (4013)  DCS[0,00,0,0,0,0]  BM[0100,,00,11,,00,01,,101,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,011,110]



    ;CHECK BIT<05:00> = "10 1010"
    4615:
    TEST115A3:

            P0,     LOAD-ENUA(ZTARGET425),                 ;INSTRS-ESS OUTPUT FOR BIT <05:00> = "101 010"
                    LOAD-ERROR(TEST115A3),                 ;ERROR DIRECTORY KEY
                    DCS-CTR(C6,),                          ;COMPARE AT TARGET
            NEXT,   J/GOBUT115A3                           ;
```



```
  (4615)  DCS[1,00,1,0,0,0]  BM[1001,,00,11,,11,00,,010,,101,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,001,100]

    4014:  ;(FREE)
    GOBUT115A3:
            SETUP,  RETURN/TEST115B1,                      ;EXEC SUBR WHICH:
                                                           ;(1) D<05:00> => IR<05:00>
            NEXT,   CALL[D05-00]                           ;(2) BUT(INSTRS) INTO ZTARGET---
  (4014)  DCS[0,00,0,0,0,0]  BM[0100,,00,11,,00,01,,100,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,100,000]




    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

    ;TESTS 115B 1-3 VERIFIES THAT WITH:
    ;ALU=(NOT-A), A=(125252), B=(177777), THEN D=(052525), AND D(C)_PS[C]=(0)
    4614:
    TEST115B1:
            P0,     LOAD-ENUA(ZTARGET405),                 ;BIT <15:12> = "0101"
                    LOAD-ERROR(TEST115B1),                 ;ERROR DIRECTORY KEY
                    DCS-CTR(C6,),                          ;COMPARE AT TARGET
            NEXT,   J/ALU115B1                             ;
  (4614)  DCS[1,00,1,0,0,0]  BM[1001,,00,11,,11,00,,000,,101,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,001,101]

    4015:  ;(FREE)
    ALU115B1:
            P0,     BUMP-VERIFY,                           ;COUNT
            P2-7,   D_NOT-A, D[C]_PS[C],                   ;ALU=(NOT-A), D[C]=PS[C]=(0)
                    BUS-A_ASPHI(C125252),                  ;A=(125252)
                    BUS-B_CSPD(C177777),                   ;B=(177777)
            NEXT,   J/GOBUT115B1                           ;D=(052525)
  (4015)  DCS[0,00,0,0,0,1]  BM[0000,,10,00,,11,01,,110,,001,,,0,1,0,,0,,0,,,0,0101,,,0,,0000,0,,,11,000,,,000,001,110]

    4016:  ;(FREE)
    GOBUT115B1:
            SETUP,  RETURN/TEST115B4,                      ;EXEC SUBR WHICH:
                                                           ;(1) D<15:12> => IR<15:12>
            NEXT,   CALL[D15-12]                           ;(2) BUT(IR15-12) INTO ZTARGET---
  (4016)  DCS[0,00,0,0,0,0]  BM[0100,,00,11,,00,00,,010,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,011,100]


    ;CHECK THAT D[C] GOT A (0) FROM "PS[C]", ABOVE
    4602:
    TEST115B4:
            P0,     LOAD-ENUA(ZTARGET402),                 ;BIT <00> = D[C] = (0)
                    LOAD-ERROR(TEST115B4),                 ;ERROR DIRECTORY KEY

                    DCS-CTR(C3,),                          ;COMPARE AT TARGET
                    BUMP-VERIFY,                           ;COUNT
            NEXT,   J/GOBUT115B4                           ;
  (4602)  DCS[1,00,1,0,0,1]  BM[1100,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,001,111]

    4017:  ;(FREE)
```

```
GOBUT115B4:
        SETUP,  RETURN/TEST115B2,                   ;RETURN TO START OF NEXT SUBTEST
        NEXT,   GOTO-PAGE(7),                       ;BUT TABLE
                J/BUTD[C]A                          ;D[C]A IN BIT<00>
(4017)  DCS[0,00,0,0,0,0]  BM[0100,,00,11,,00,01,,011,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,100,100]


;CHECK BIT<11:06> = "0101 01"
4613:
TEST115B2:
        P0,     LOAD-ENUA(ZTARGET406),              ;INSTRS-E70 OUTPUT FOR BIT <11:06> = "010 101"
                LOAD-ERROR(TEST115B2),              ;ERROR DIRECTORY KEY
                DCS-CTR(C6,),                       ;COMPARE AT TARGET
                BUMP-VERIFY,                        ;COUNT
        NEXT,   J/GOBUT115B2                        ;
(4613)  DCS[1,00,1,0,0,1]  BM[1001,,00,11,,11,00,,000,,101,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,010,000]

4020:   ;(FREE)
GOBUT115B2:
        SETUP,  RETURN/TEST115B3,                   ;EXEC SUBR WHICH:
                                                    ;(1) D<11:06> => IR<11:06>
        NEXT,   CALL[D11-06]                        ;(2) BUT(INSTRS) INTO ZTARGET---
(4020)  DCS[0,00,0,0,0,0]  BM[0100,,00,11,,00,01,,010,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,011,110]



;CHECK BIT<05:00> = "01 0101"
4612:
TEST115B3:

        P0,     LOAD-ENUA(ZTARGET432),              ;INSTRS-E80 OUTPUT FOR BIT<05:00> = "010 101"
                LOAD-ERROR(TEST115B3),              ;ERROR DIRECTORY KEY
                DCS-CTR(C6,),                       ;COMPARE AT TARGET
        NEXT,   J/GOBUT115B3                        ;
(4612)  DCS[1,00,1,0,0,0]  BM[1001,,00,11,,11,00,,011,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,010,001]

4021:   ;(FREE)
GOBUT115B3:
        SETUP,  RETURN/SCOPE115B,                   ;EXEC SUBR WHICH:
                                                    ;(1) D<05:00> => IR<05:00>
        NEXT,   CALL[D05-00]                        ;(2) BUT(INSTRS) INTO ZTARGET---
(4021)  DCS[0,00,0,0,0,0]  BM[0100,,00,00,,00,10,,010,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,100,000]


4022:   ;(FREE)
SCOPE115B:
        NEXT,   BUTD(SCOPE),                        ;NO ERROR:  "TEST115C1"   (+1, WORDS)
                J/TEST115C1                         ;   ERROR:  "ALU115A1" (-17,WORDS)
(4022)  DCS[0,00,0,1,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,110,001,001]
```



```
;  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -

;TESTS 115C 1-3 VERIFIES THAT WITH:
;ALU=(NOT-A), A=(052525), B=(000000), THEN D=(125252)
4611:
TEST115C1:
        P0,     LOAD-ENUA(ZTARGET412),              ;BIT <15:12> = "1010"
                LOAD-ERROR(TEST115C1),              ;ERROR DIRECTORY KEY
                DCS-CTR(C6,),                       ;COMPARE AT TARGET
        NEXT,   J/ALU115C1                          ;
(4611)  DCS[1,00,1,0,0,0]  BM[1001,,00,11,,11,00,,001,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,110,010,010]

4622:
ALU115C1:
        P0,     BUMP-VERIFY,                        ;COUNT
        P2-T,   D_NOT-A, SAVE-D(C),                 ;ALU=(NOT-A)
                BUS-A_ASPHI(C052525),               ;A=(052525)
                BUS-B_CSPD(C000000),                ;B=(000000)
        NEXT,   J/GOBUT115C1                        ;D=(125252)
(4622)  DCS[0,00,0,0,0,1]  BM[0000,,10,00,,11,01,,111,,111,,,0,1,0,,0,,0,,,0,0100,,,0,,0000,0,,,11,000,,,000,010,011]

4023:   ;(FREE)
GOBUT115C1:
        SETUP,  RETURN/TEST115C2,                   ;EXEC SUBR WHICH:
                                                    ;(1) D<15:12> => IR<15:12>
        NEXT,   CALL[D15-12]                        ;(2) BUT(IR15-12) INTO ZTARGET---
(4023)  DCS[0,00,0,0,0,0]  BM[0100,,00,11,,00,01,,111,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,011,100]

;CHECK BIT<11:06> = "1010 10"
4617:
TEST115C2:
        P0,     LOAD-ENUA(ZTARGET412),              ;INSTRS-E70 OUTPUT FOR BIT <11:06> = "101 010"
                LOAD-ERROR(TEST115C2),              ;ERROR DIRECTORY KEY
                DCS-CTR(C6,),                       ;COMPARE AT TARGET
                BUMP-VERIFY,                        ;COUNT
        NEXT,   J/GOBUT115C2                        ;
(4617)  DCS[1,00,1,0,0,1]  BM[1001,,00,11,,11,00,,001,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,010,100]

4024:   ;(FREE)
GOBUT115C2:
        SETUP,  RETURN/TEST115C3,                   ;EXEC SUBR WHICH:
                                                    ;(1) D<11:06> => IR<11:06>
        NEXT,   CALL[D11-06]                        ;(2) BUT(INSTRS) INTO ZTARGET---
(4024)  DCS[0,00,0,0,0,0]  BM[0100,,00,11,,00,10,,111,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,011,110]




;CHECK BIT<05:00> = "10 1010"
4627:
TEST115C3:
```

```
        P0,     LOAD-ENUA(ZTARGET425),                         ;INSTRS-E88 OUTPUT FOR BIT<05:00> = "101 010"
                LOAD-ERROR(TEST115C3),                         ;ERROR DIRECTORY KEY
                DCS-CTR(C6,),                                  ;COMPARE AT TARGET
        NEXT,   J/GOBUT115C3                                   ;
  (4627)  DCS[1,00,1,0,0,0]  BM[1001,,00,11,,11,00,,010,,101,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,010,101]

    4025:  ;(FREE)
    GOBUT115C3:
        SETUP,  RETURN/TEST115D1,                              ;EXEC SUBR WHICH:
                                                               ;(1) D<05:00> -> IR<05:00>
        NEXT,   CALL(D05-00)                                   ;(2) BUT(INSTRS) INTO ZTARGET---
  (4025)  DCS[0,00,0,0,0,0]  BM[0100,,00,11,,00,10,,110,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,100,000]




    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

    ;TESTS 115D 1-3 VERIFIES THAT WITH:
    ;ALU=(NOT-A, A=(125252), B=(000000), THEN D=(052525)
    4626:
    TEST115D1:
        P0,     LOAD-ENUA(ZTARGET405),                         ;BIT<15:12> = "0101"
                LOAD-ERROR(TEST115D1),                         ;ERROR DIRECTORY KEY
                DCS-CTR(C6,),                                  ;COMPARE AT TARGET
        NEXT,   J/ALU115D1                                     ;
  (4626)  DCS[1,00,1,0,0,0]  BM[1001,,00,11,,11,00,,000,,101,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,010,110]

    4026:  ;(FREE)
    ALU115D1:
        P0,     BUMP-VERIFY,                                   ;COUNT
        P2-T,   D_NOT-A, SAVE-D[C],                            ;ALU=(NOT-A), D[C]_0
                BUS-A_ASPHI(C125252),                          ;A=(125252)
                BUS-B_CSPD(C000000),                           ;B=(000000)
        NEXT,   J/GOBUT115D1                                   ;D=(052525)
  (4026)  DCS[0,00,0,0,0,1]  BM[0000,,10,00,,11,01,,110,,111,,,0,1,0,,0,,0,,,0,0100,,,0,,0000,0,,,11,000,,,000,010,111]

    4027:  ;(FREE)
    GOBUT115D1:
        SETUP,  RETURN/TEST115D2,                              ;EXEC SUBR WHICH:
                                                               ;(1) D<15:12> -> IR<15:12>
        NEXT,   CALL(D15-12)                                   ;(2) BUT(IR15-12) INTO ZTARGET---
  (4027)  DCS[0,00,0,0,0,0]  BM[0100,,00,11,,00,10,,101,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,011,100]

    ;CHECK BIT<11:06> = "0101 01"
    4625:
    TEST115D2:
        P0,     LOAD-ENUA(ZTARGET405),                         ;INSTRS-E78 OUTPUT FOR BIT<11:06> = "010 101"
                LOAD-ERROR(TEST115D2),                         ;ERROR DIRECTORY KEY
                DCS-CTR(C6,),                                  ;COMPARE AT TARGET
                BUMP-VERIFY,                                   ;COUNT
```



```
        NEXT,   J/GOBUT115D2                                   ;
  (4625)  DCS[1,00,1,0,0,1]  BM[1001,,00,11,,11,00,,000,,101,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,011,000]

    4030:  ;(FREE)
    GOBUT115D2:
        SETUP,  RETURN/TEST115D3,                              ;EXEC SUBR WHICH:
                                                               ;(1) D<11:06> -> IR<11:06>
        NEXT,   CALL(D11-06)                                   ;(2) BUT(INSTRS) INTO ZTARGET---
  (4030)  DCS[0,00,0,0,0,0]  BM[0100,,00,11,,00,10,,100,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,011,110]



    ;CHECK BIT<05:00> = "01 0101"
    4624:
    TEST115D3:

        P0,     LOAD-ENUA(ZTARGET432),                         ;INSTRS-E88 OUTPUT FOR BIT<05:00> = "010 101"
                LOAD-ERROR(TEST115D3),                         ;ERROR DIRECTORY KEY
                DCS-CTR(C6,),                                  ;CONAPRE AT TARGET
        NEXT,   J/GOBUT115D3                                   ;
  (4624)  DCS[1,00,1,0,0,0]  BM[1001,,00,11,,11,00,,011,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,011,001]

    4031:  ;(FREE)
    GOBUT115D3:
        SETUP,  RETURN/SCOPE115D,                              ;EXEC SUBR WHICH:
                                                               ;(1) D<05:00> -> IR<05:00>
        NEXT,   CALL(D05-00)                                   ;(2) BUT(INSTRS) INTO ZTARGET---
  (4031)  DCS[0,00,0,0,0,0]  BM[0100,,00,00,,00,11,,010,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,100,000]

    4032:  ;(FREE)
    SCOPE115D:
        P0,     BUMP-VERIFY,                                   ;COUNT
        NEXT,   BUTO[SCOPE],                                   ;NO ERROR: "TEST116A1" (+1, WORDS)
                J/TEST116A1                                    ;   ERROR: "ALU115C1"  (-13, WORDS)
  (4032)  DCS[0,00,0,1,0,1]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,110,010,011]




    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

    ;THIS NEXT SET OF 12, TESTS EXERCISES THE ALU FUNCTIONS
    ;"ZERO" AND "NOT-A-AND-B", AND THE D(C) INPUTS "ALU15" AND "D[C]"

    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -


    ;TESTS 116A 1-5 VERIFIES THAT WITH:
    ;ALU=(NOT-A-AND-B), A=(000000), B=(125252), THEN D=(125252)
    4623:
    TEST116A1:
        P0,     LOAD-ENUA(ZTARGET412),                         ;BIT<15:12> = "1010"
                LOAD-ERROR(TEST116A1),                         ;ERROR DIRECTORY KEY
```

```
                DCS-CTR(C6,),                            ;COMPARE AT TARGET
        NEXT,   J/ALU116A1                               ;
  (4623)  DCS[1,00,1,0,0,0]  BM[1001,,00,11,,11,00,,001,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,110,011,100]

    4634:
    ALU116A1:
        P0,     BUMP-VERIFY,                             ;COUNT
        P2-T,   D_NOT-A-AND-B, D[C]_ALU15,               ;ALU=(NOT-A-AND-B), D[C]=(1)
                BUS-A_ASPHI(C000000),                    ;A=(000000)
                BUS-B_CSPD(C125252),                     ;B=(125252)
        NEXT,   J/GOBUT116A1                             ;D=(125252)
  (4634)  DCS[0,00,0,0,0,1]  BM[0010,,10,00,,11,01,,100,,100,,,0,1,0,,0,,0,,,0,0110,,,0,,0000,0,,,11,000,,,000,011,011]

    4033:  ;(FREE)
    GOBUT116A1:
        SETUP,  RETURN/TEST116A2,                        ;EXEC SUBR WHICH:
                                                         ;(1) D<15:12> => IR<15:12>
        NEXT,   CALL[D15-12]                             ;(2) BUT (IR15-12) INTO ZTARGET---
  (4033)  DCS[0,00,0,0,0,0]  BM[0100,,00,11,,00,10,,001,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,011,100]

    ;CHECK THAT D[C] GOT A (1) FROM "ALU15," ABOVE
    4621:
    TEST116A2:
        P0,     LOAD-ENUA(ZTARGET403),                   ;BIT <00> = D(C) = (1)
                LOAD-ERROR(TEST116A2),                   ;ERROR DIRECTORY KEY
                DCS-CTR(C3,),                            ;COMPARE AT TARGET
                BUMP-VERIFY,                             ;COUNT
        NEXT,   J/GOBUT116A2                             ;
  (4621)  DCS[1,00,1,0,0,1]  BM[1100,,00,11,,11,00,,000,,011,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,011,100]
    4034:  ;(FREE)
    GOBUT116A2:
        SETUP,  RETURN/TEST116A3,                        ;RETURN TO START OF NEXT SUBTEST
        NEXT,   GOTO-PAGE(7),                            ;BUT TABLE
                J/BUTD[C]A                               ;D[C]N IN BIT <00>
  (4034)  DCS[0,00,0,0,0,0]  BM[0100,,00,11,,00,10,,000,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,100,100]

    ;CHECK BIT<11:06> = "1010 10"
    4620:
    TEST116A3:
        P0,     LOAD-ENUA(ZTARGET412),                   ;INSTRS-ETS OUTPUT FOR BIT <11:06> = "101 010"
                LOAD-ERROR(TEST116A3),                   ;ERROR DIRECTORY KEY
                DCS-CTR(C6,),                            ;COMPARE AT TARGET
                BUMP-VERIFY,                             ;COUNT
        NEXT,   J/GOBUT116A3                             ;
  (4620)  DCS[1,00,1,0,0,1]  BM[1001,,00,11,,11,00,,001,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,011,101]

    4035:  ;(FREE)
    GOBUT116A3:
        SETUP,  RETURN/TEST116A4,                        ;EXEC SUBR WHICH:
                                                         ;(1) D<11:06> => IR<11:06>
        NEXT,   CALL[D11-06]                             ;(2) BUT(INSTRS) INTO ZTARGET---
```



```
  (4035)  DCS[0,00,0,0,0,0]  BM[0100,,00,11,,00,11,,111,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,011,110]



    ;CHECK BIT<05:00> = "10 1010"
    4637:
    TEST116A4:

        P0,     LOAD-ENUA(ZTARGET425),                   ;INSTRS-ESS OUTPUT FOR BIT <05:00> ="101 010"
                LOAD-ERROR(TEST116A4),                   ;ERROR DIRECTORY KEY
                DCS-CTR(C6,),                            ;COMPARE AT TARGET
        NEXT,   J/GOBUT116A4                             ;
  (4637)  DCS[1,00,1,0,0,0]  BM[1001,,00,11,,11,00,,010,,101,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,011,110]

    4036:  ;(FREE)
    GOBUT116A4:
        SETUP,  RETURN/TEST116A5,                        ;EXEC SUBR WHICH:
                                                         ;(1) D<05:00> => IR<05:00>
        NEXT,   CALL[D05-00]                             ;(2) BUT(INSTRS) INTO ZTARGET---
  (4036)  DCS[0,00,0,0,0,0]  BM[0100,,00,11,,00,11,,110,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,100,000]



    ;CHECK THAT D[C] WAS PROPOGATED UNCHANGED AS A (1), VIA D[C]_D[C]
    4636:
    TEST116A5:
        P0,     LOAD-ENUA(ZTARGET403),                   ;BIT<01> = D(C) = (1)
                LOAD-ERROR(TEST116A5),                   ;ERROR DIRECTORY KEY
                DCS-CTR(C3,),                            ;COMPARE AT TARGET
        NEXT,   J/GOBUT116A5                             ;
  (4636)  DCS[1,00,1,0,0,0]  BM[1100,,00,11,,11,00,,000,,011,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,011,111]

    4037:  ;(FREE)
    GOBUT116A5:
        SETUP,  RETURN/TEST116B,                         ;RETURN TO START OF NEXT SUBTEST
        NEXT,   GOTO-PAGE(7),                            ;BUT TABLE
                J/BUTD[C]B                               ;D[C]N IN BIT <01>
  (4037)  DCS[0,00,0,0,0,0]  BM[0100,,00,11,,00,00,,110,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,101,000]





    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

    ;TEST 116B VERIFIES THAT WITH:

    ;ALU=(ZERO), A=(125252), B=(052525), THEN D=(000000)
    4606:
    TEST116B:
        P0,     LOAD-ENUA(ZTARGET434),                   ;INSTRS FOR IR=(000000)
                LOAD-ERROR(TEST116B),                    ;ERROR DIRECTORY KEY
                DCS-CTR(C6,),                            ;COMPARE AT TARGET
```

```
                NEXT,   J/ALU116B                                    ;
      (4606)  DCS[1,00,1,0,0,0]  BN[1001,,00,11,,11,00,,011,,100,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,100,000]

        4040:  ;(FREE)
        ALU116B:
                P2-T,   D_ZERO, SAVE-D[C],                           ;ALU=(ZERO), D[C]=D[C]=(1)
                        BUS-A_ASPHI[C125252],                        ;A=(125252)
                        BUS-B_CSPD(C052525),                         ;B=(052525)
                NEXT,   J/GOBUT116B                                  ;D=(000000)
      (4040)  DCS[0,00,0,0,0,0]  BN[0011,,10,00,,11,01,,110,,111,,,0,1,0,,0,,0,,,0,0111,,,0,,0000,0,,,11,000,,,000,100,001]

        4041:  ;(FREE)
        GOBUT116B:
                SETUP,  RETURN/SCOPE116C,                            ;EXEC SUBR WHICH:
                                                                     ;(1) D=> IR
                NEXT,   CALL[DZERO]                                  ;(2) BUT(INSTRS) INTO ZTARGET---
      (4041)  DCS[0,00,0,0,0,0]  BN[0100,,00,00,,01,00,,010,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,100,010]



        4042:  ;(FREE)
        SCOPE116C:
                NEXT,   BUTD[SCOPE],                                 ;NO ERROR:  "TEST116C1"   (+1, WORDS)
                        J/TEST116C1                                  ;   ERROR:  "ALU116A1"  (-11, WORDS)
      (4042)  DCS[0,00,0,1,0,0]  BN[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,110,011,101]



        ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        ;TEST 116C 1-5 VERIFIES THAT WITH:
        ;ALU=(NOT-A-AND-B), A=(000000), B=(052525), THEN D=(052525)
        4635:
        TEST116C1:
                P0,     LOAD-ENUA(ZTARGET405),                       ;BIT<15:12> = "0101"
                        LOAD-ERROR(TEST116C1),                       ;ERROR DIRECTORY KEY
                        DCS-CTR(C6,),                                ;COMPARE AT TARGET
                NEXT,   J/ALU116C1                                   ;
      (4635)  DCS[1,00,1,0,0,0]  BN[1001,,00,11,,11,00,,000,,101,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,110,100,110]

        4646:
        ALU116C1:
                P0,     BUMP-VERIFY,                                 ;COUNT
                P2-T,   D_NOT-A-AND-B, D[C]_ALU15,                   ;ALU=(NOT-A-AND-B), D[C]=(0)
                        BUS-A_ASPHI[C000000],                        ;A=(000000)
                        BUS-B_CSPD(C052525),                         ;B=(052525)
                NEXT,   J/GOBUT116C1                                 ;D=(052525)
      (4646)  DCS[0,00,0,0,0,1]  BN[0010,,10,00,,11,01,,100,,100,,,0,1,0,,0,,0,,,0,0111,,,0,,0000,0,,,11,000,,,000,100,011]

        4043:  ;(FREE)
        GOBUT116C1:
                SETUP,  RETURN/TEST116C2,                            ;EXEC SUBR WHICH:
                                                                     ;(1) D<15:12> => IR<15:12>
```



```
                NEXT,   CALL[D15-12]                                 ;(2) BUT(IR15-12) INTO ZTARGET---
      (4043)  DCS[0,00,0,0,0,0]  BN[0100,,00,11,,00,11,,011,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,011,100]

        ;CHECK THAT D[C] GOT A (0) FROM "ALU15", ABOVE
        4633:
        TEST116C2:
                P0,     LOAD-ENUA(ZTARGET402),                       ;BIT <00> = D[C] = (0)
                        LOAD-ERROR(TEST116C2),                       ;ERROR DIRECTORY KEY
                        DCS-CTR(C3,),                                ;COMPARE AT TARGET
                        BUMP-VERIFY,                                 ;COUNT
                NEXT,   J/GOBUT116C2                                 ;
      (4633)  DCS[1,00,1,0,0,1]  BN[1100,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,100,100]

        4044:  ;(FREE)
        GOBUT116C2:
                SETUP,  RETURN/TEST116C3,                            ;RETURN TO START OF NEXT SUBTEST
                NEXT,   GOTO-PAGE(7),                                ;BUT TABLE
                        J/BUTD[C]A                                   ;D[C]N IN BIT<00>
      (4044)  DCS[0,00,0,0,0,0]  BN[0100,,00,11,,00,11,,010,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,100,100]

        ;CHECK BIT<11:06> = "0101 01"
        4632:
        TEST116C3:
                P0,     LOAD-ENUA(ZTARGET405),                       ;INSTRS-E78 OUTPUT FOR BIT<11:06>="010 101"
                        LOAD-ERROR(TEST116C3),                       ;ERROR DIRECTORY KEY
                        DCS-CTR(C6,),                                ;COMPARE AT TARGET
                        BUMP-VERIFY,                                 ;COUNT
                NEXT,   J/GOBUT116C3                                 ;
      (4632)  DCS[1,00,1,0,0,1]  BN[1001,,00,11,,11,00,,000,,101,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,100,101]

        4045:  ;(FREE)
        GOBUT116C3:
                SETUP,  RETURN/TEST116C4,                            ;EXEC SUBR WHICH:
                                                                     ;(1) D<11:06>=> IR<11:06>
                NEXT,   CALL[D11-06]                                 ;(2) BUT(INSTRS) INTO ZTARGET---
      (4045)  DCS[0,00,0,0,0,0]  BN[0100,,00,11,,00,11,,001,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,011,110]



        ;CHECK BIT <05:00> = "01 0101"
        4631:
        TEST116C4:

                P0,     LOAD-ENUA(ZTARGET432),                       ;INSTRS-E86 OUTPUT FOR BIT <05:00> = "010 101"
                        LOAD-ERROR(TEST116C4),                       ;ERROR DIRECTORY KEY
                NEXT,   J/GOBUT116C4                                 ;

      (4631)  DCS[1,00,0,0,0,0]  BN[0000,,00,11,,11,00,,011,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,100,110]

        4046:  ;(FREE)
        GOBUT116C4:
                SETUP,  RETURN/TEST116C5,                            ;EXEC SUBR WHICH:
```

```
                                                                ;(1) D<05:00> => IR<05:00>
            NEXT,   CALL[D05-00]                                ;(2) BUT(INSTRS) INTO ZTARGET---
   (4046)  DCS[0,00,0,0,0,0]  BM[0100,,00,11,,00,11,,000,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,100,000]



    ;CHECK THAT D(C) WAS PROPOGATED UNCHANGED AS A (0), VIA D[C]_D[C]
    4630:
    TEST116C5:
            P0,     LOAD-ENUA(ZTARGET401),                      ;BIT<01> = D[C] = (0)
                    LOAD-ERROR(TEST116C5),                      ;ERROR DIRECTORY KEY
                    DCS-CTR(C3.),                               ;COMPARE AT TARGET
            NEXT,   J/GOBUT116C5                                ;
   (4630)  DCS[1,00,1,0,0,0]  BM[1100,,00,11,,11,00,,000,,001,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,100,111]

    4047:  ;(FREE)
    GOBUT116C5:

            SETUP,  RETURN/TEST116D,                            ;RETURN TO START OF NEXT SUBTEST
            NEXT,   GOTO-PAGE(7),                               ;BUT TABLE
                    J/BUTO[C]B                                  ;D[C]N IN BIT <01>
   (4047)  DCS[0,00,0,0,0,0]  BM[0100,,00,11,,01,00,,101,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,101,000]






    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

    ;TEST 116D VERIFIES THAT WITH:
    ;ALU=(ZERO), A=(052525), B=(125252), THEN D=(000000)
    4645:
    TEST116D:
            P0,     LOAD-ENUA(ZTARGET434),                      ;INSTRS FOR IR=(000000)
                    LOAD-ERROR(TEST116D),                       ;ERROR DIRECTORY KEY
                    DCS-CTR(C6.),                               ;COMPARE AT TARGET
            NEXT,   J/ALU116D                                   ;
   (4645)  DCS[1,00,1,0,0,0]  BM[1001,,00,11,,11,00,,011,,100,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,101,000]

    4050:  ;(FREE)
    ALU116D:
            P2-T,   D_ZERO, SAVE-D[C],                          ;ALU=(ZERO), D[C]=D[C]=(0)
                    BUS-A_ASPHI(C052525),                       ;A=(052525)
                    BUS-B_CSPD(C125252),                        ;B=(125252)
            NEXT,   J/GOBUT116D                                 ;D=(000000)
   (4050)  DCS[0,00,0,0,0,0]  BM[0011,,10,00,,11,01,,111,,111,,,0,1,0,,0,,0,,,0,0110,,,0,,0000,0,,,11,000,,,000,101,001]

    4051:  ;(FREE)
    GOBUT116D:
            SETUP,  RETURN/SCOPE116D,                           ;EXEC SUBR WHICH:
                                                                ;(1) 0  => IR
```



```
            NEXT,   CALL[DZERO]                                 ;(2) BUT(INSTRS) INTO ZTARGET---
   (4051)  DCS[0,00,0,0,0,0]  BM[0100,,00,00,,01,01,,010,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,100,010]



    4052:  ;(FREE)
    SCOPE116D:
            NEXT,   BUTO[SCOPE],                                ;NO ERROR: "TEST117A1" (+1, WORDS)
                    J/TEST117A1                                 ;   ERROR: "ALU116C1" (-13, WORDS)
   (4052)  DCS[0,00,0,1,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,110,100,111]





    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

    ;THIS NEXT SET OF 9. TESTS EXERCIZES THE ALU FUNCTION
    ;"A-AND-NOT-B", AND THE CARRYOUT OF "CINMUX=D[C]" INTO D(C)




    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

    ;TESTS 117A 1-4 VERIFIES THAT WITH:
    ;ALU=(A-AND-NOT-B), A=(177777), B=(125252), THEN D=(052525)
    4647:
    TEST117A1:
            P0,     LOAD-ENUA(ZTARGET405),                      ;BIT<15:12> = "0101"
                    LOAD-ERROR(TEST117A1),                      ;ERROR DIRECTORY KEY
                    DCS-CTR(C6.),                               ;COMPARE AT TARGET
            NEXT,   J/ALU117A1                                  ;
   (4647)  DCS[1,00,1,0,0,0]  BM[1001,,00,11,,11,00,,000,,101,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,110,110,110]

    4666:
    ALU117A1:
            P0,     BUMP-VERIFY,                                ;COUNT
            P2-T,   D_A-AND-NOT-B, D(C)_CINMUX,                 ;ALU=(A-AND-NOT-B), D[C]=CIN=D[C]=(0)
                    BUS-A_ASPHI(C177777),                       ;A=(177777)
                    BUS-B_CSPD(C125252),                        ;B=(125252)
            NEXT,   J/GOBUT117A1                                ;D=(052525)
   (4666)  DCS[0,00,0,0,0,1]  BM[0111,,10,00,,11,01,,101,,000,,,0,1,0,,0,,0,,,0,0110,,,0,,0000,0,,,11,000,,,000,101,011]

    4053:  ;(FREE)
    GOBUT117A1:

            SETUP,  RETURN/TEST117A2,                           ;EXEC SUBR WHICH:
                                                                ;(1) D<15:12> => IR<15:12>
            NEXT,   CALL[D15-12]                                ;(2) BUT(IR15-12) INTO ZTARGET---
   (4053)  DCS[0,00,0,0,0,0]  BM[0100,,00,11,,01,00,,100,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,011,100]

    ;CHECK BIT<11:06> = "0101 01"
```

```
	4644:
	TEST117A2:
		P0,	LOAD-ENUA(ZTARGET405),			;INSTRS-E70 OUTPUT FOR BIT<11:06>="010 101"
			LOAD-ERROR(TEST117A2),			;ERROR DIRECTORY KEY
			DCS-CTR(C6,),				;COMPARE AT TARGET
			BUMP-VERIFY,				;COUNT
		NEXT,	J/GOBUT117A2				;
    (4644)  DCS[1,00,1,0,0,1]  BM[1001,,00,11,,11,00,,000,,101,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,101,100]

	4054:	;(FREE)
	GOBUT117A2:
		SETUP,	RETURN/TEST117A3,			;EXEC SUBR WHICH:
								;(1) D<11:06> => IR<11:06>
		NEXT,	CALL[D11-06]				;(2) BUT(INSTRS) INTO ZTARGET---
    (4054)  DCS[0,00,0,0,0,0]  BM[0100,,00,11,,01,00,,011,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,011,110]



	;CHECK BIT<05:00> = "01 0101"
	4643:
	TEST117A3:

		P0,	LOAD-ENUA(ZTARGET432),			;INSTRS-E80 OUTPUT FOR BIT <05:00>="010 101"
			LOAD-ERROR(TEST117A3),			;ERROR DIRECTORY KEY
			DCS-CTR(C6,),				;COMPARE AT TARGET
		NEXT,	J/GOBUT117A3				;
    (4643)  DCS[1,00,1,0,0,0]  BM[1001,,00,11,,11,00,,011,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,101,101]

	4055:	;(FREE)
	GOBUT117A3:
		SETUP,	RETURN/TEST117A4,			;EXEC SUBR WHICH:
								;(1) D<05:00> => IR<05:00>
		NEXT,	CALL[D05-00]				;(2) BUT(INSTRS) INTO ZTARGET---
    (4055)  DCS[0,00,0,0,0,0]  BM[0100,,00,11,,01,00,,010,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,100,000]

	;CHECK THAT D[C] GOT A (0) FROM CINMUX, ABOVE
	4642:
	TEST117A4:
		P0,	LOAD-ENUA(ZTARGET413),			;BIT<03> = D[C] = (0)
			LOAD-ERROR(TEST117A4),			;ERROR DIRECTORY KEY
			DCS-CTR(C3,),				;COMPARE AT TARGET
			BUMP-VERIFY,				;COUNT
		NEXT,	J/GOBUT117A4				;
    (4642)  DCS[1,00,1,0,0,1]  BM[1100,,00,11,,11,00,,001,,011,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,101,110]

	4056:	;(FREE)
	GOBUT117A4:
		SETUP,	RETURN/TEST117B1,			;RETURN TO START OF NEXT SUBTEST
		NEXT,	GOTO-PAGE(7),				;BUT TABLE
			J/BUTD[C]C				;D[C]=  IN BIT<03>
    (4056)  DCS[0,00,0,0,0,0]  BM[0100,,00,11,,01,00,,001,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,001,000]
```



```




	; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

	;TESTS 117B 1-3 VERIFIES THAT WITH:
	;ALU=(A-AND-NOT-B), A=(177777), B=(052525), THEN D=(125252)
	4641:
	TEST117B1:
		P0,	LOAD-ENUA(ZTARGET412),			;BIT<15:12> = "1010"
			LOAD-ERROR(TEST117B1),			;ERROR DIRECTORY KEY
			DCS-CTR(C6,),				;COMPARE AT TARGET
		NEXT,	J/ALU117B1				;
    (4641)  DCS[1,00,1,0,0,0]  BM[1001,,00,11,,11,00,,001,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,101,111]

	4057:	;(FREE)
	ALU117B1:
		P0,	BUMP-VERIFY,				;COUNT
		P2-T,	D_A-AND-NOT-B, D[C]_ALU15,		;ALU=(A-AND-NOT-B), D[C]=ALU15=(1)
			BUS-A_ASPHI(C177777),			;A=(177777)
			BUS-B_CSPD(C052525),			;B=(052525)
		NEXT,	J/GOBUT117B1				;D=(125252)
    (4057)  DCS[0,00,0,0,0,1]  BM[0111,,10,00,,11,01,,101,,100,,,0,1,0,,0,,0,,,0,0111,,,0,,0000,0,,,11,000,,,000,110,000]

	4060:	;(FREE)
	GOBUT117B1:
		SETUP,	RETURN/TEST117B2,			;EXEC SUBR WHICH:
								;(1) D<15:12> => IR<15:12>
		NEXT,	CALL[D15-12]				;(2) BUT(IR15-12) INTO ZTARGET---
    (4060)  DCS[0,00,0,0,0,0]  BM[0100,,00,11,,01,00,,000,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,011,100]

	;CHECK BIT <11:06> = "1010 10"
	4640:
	TEST117B2:
		P0,	LOAD-ENUA(ZTARGET412),			;INSTRS-E70 OUTPUT FOR BIT <11:06>="101 010"
			LOAD-ERROR(TEST117B2),			;ERROR DIRECTORY KEY
			DCS-CTR(C6,),				;COMPARE AT TARGET
			BUMP-VERIFY,				;COUNT
		NEXT,	J/GOBUT117B2				;
    (4640)  DCS[1,00,1,0,0,1]  BM[1001,,00,11,,11,00,,001,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,110,001]

	4061:	;(FREE)
	GOBUT117B2:
		SETUP,	RETURN/TEST117B3,			;EXEC SUBR WHICH:
								;(1) D<11:06> => IR<11:06>

		NEXT,	CALL[D11-06]				;(2) BUT(INSTRS) INTO ZTARGET---
    (4061)  DCS[0,00,0,0,0,0]  BM[0100,,00,11,,01,01,,111,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,011,110]



	;CHECK BIT<05:00> = "10 1010"
```

```
    4657:
    TEST117B3:

          P0,     LOAD-ENUA(ZTARGET425),              ;INSTRS-E86 OUTPUT FOR BIT <05:00> = "101 010"
                  LOAD-ERROR(TEST117B3),              ;ERROR DIRECTORY KEY
                  DCS-CTR(C6.),                       ;COMPARE AT TARGET
          NEXT,   J/GOBUT117B3                        ;
 (4657) DCS[1,00,1,0,0,0] BM[1001,,00,11,,11,00,,010,,101,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,110,010]

    4062: ;(FREE)
    GOBUT117B3:
          SETUP,  RETURN/TEST117C1,                   ;EXEC SUBR WHICH:
                                                      ;(1) D<05:00> -> IR<05:00>
          NEXT,   CALL[DOS-00]                        ;(2) BUT (INSTRS) INTO ZTARGET---
 (4062) DCS[0,00,0,0,0,0] BM[0100,,00,11,,01,01,,110,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,100,000]







    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

    ;TESTS 117C 1-2 VERIFIES THAT WITH:
    ;ALU=(A-AND-NOT-B), A=(000000), B=(000000), THEN D=(000000)
    4656:
    TEST117C1:
          P0,     LOAD-ENUA(ZTARGET434),              ;INSTRS FOR IR=(000000)
                  LOAD-ERROR(TEST117C1),              ;ERROR DIRECTORY KEY
                  DCS-CTR(C6.),                       ;COMPARE AT TARGET
          NEXT,   J/ALU117C1                          ;
 (4656) DCS[1,00,1,0,0,0] BM[1001,,00,11,,11,00,,011,,100,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,110,011]

    4063: ;(FREE)
    ALU117C1:
          P2-T,   D_A-AND-NOT-B, D[C]_CINMUX,         ;ALU=(A-AND-NOT-B), D[C]=CINMUX=D[C]=(1)
                  BUS-A_ASPHI(C000000),               ;A=(000000)
                  BUS-B_CSPD(C000000),                ;B=(000000)
          NEXT,   J/GOBUT117C1                        ;
 (4063) DCS[0,00,0,0,0,0] BM[0111,,10,00,,11,01,,100,,000,,,0,1,0,,0,,0,,,0,0100,,,0,,0000,0,,,11,000,,,000,110,100]

    4064: ;(FREE)
    GOBUT117C1:
          SETUP,  RETURN/TEST117C2,                   ;EXEC SUBR WHICH:
                                                      ;(1) D  -> IR
          NEXT,   CALL[DZERO]                         ;(2) BUT(INSTRS) INTO ZTARGET---
 (4064) DCS[0,00,0,0,0,0] BM[0100,,00,11,,01,11,,111,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,100,010]


    ;CHECK THAT D[C] GOT A (1) FROM CINMUX, ABOVE
    4677:
```



```
    TEST117C2:
          P0,     LOAD-ENUA(ZTARGET417),              ;BIT<03> = D[C] = (1)
                  LOAD-ERROR(TEST117C2),              ;ERROR DIRECTORY KEY
                  DCS-CTR(C3.),                       ;COMPARE AT TARGET
          NEXT,   J/GOBUT117C2                        ;
 (4677) DCS[1,00,1,0,0,0] BM[1100,,00,11,,11,00,,001,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,110,101]

    4065: ;(FREE)
    GOBUT117C2:
          SETUP,  RETURN/SCOPE117C,                   ;RETURN TO SCOPE LOOP TEST WORD
          NEXT,   GOTO-PAGE(7),                       ;BUT TABLE
                  J/BUTD[C]C                          ;BIT<03> = D[C]H
 (4065) DCS[0,00,0,0,0,0] BM[0100,,00,00,,01,10,,110,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,001,000]



    4066: ;(FREE)
    SCOPE117C:
          NEXT,   BUTD[SCOPE],                        ;NO ERROR: "TEST120A1" (+1, WORDS)
                  J/TEST120A1                         ; ERROR: "ALU117A1" (-20, WORDS)
 (4066) DCS[0,00,0,1,0,0] BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,110,110,111]


    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

    ;THIS NEXT SET OF 7, TESTS EXERCIZES THE ALU FUNCTION
    ;"A-AND-B", AND THE CARRYOUT OF "CINMUX =(0) INTO D[C]

    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

    ;TESTS 120A 1-3 VERIFIES THAT WITH:
    ;ALU=(A-AND-B), A=(125252), B=(177777), THEN D=(125252)
    4667:
    TEST120A1:
          P0,     LOAD-ENUA(ZTARGET412),              ;BIT<15:12> = "1010"
                  LOAD-ERROR(TEST120A1),              ;ERROR DIRECTORY KEY
                  DCS-CTR(C6.),                       ;COMPARE AT TARGET
          NEXT,   J/ALU120A1                          ;
 (4667) DCS[1,00,1,0,0,0] BM[1001,,00,11,,11,00,,001,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,110,110,100]

    4664:
    ALU120A1:
          P0,     BUMP-VERIFY,                        ;COUNT
          P2-T,   D_A-AND-B, D[C]_ALU15,              ;ALU=(A-AND-B), D[C]=(1)
                  BUS-A_ASPHI(C125252),               ;A=(125252)
                  BUS-B_CSPD(C177777),                ;B=(177777)

          NEXT,   J/GOBUT120A1                        ;D=(125252)
 (4664) DCS[0,00,0,0,0,1] BM[1011,,10,00,,11,01,,110,,100,,,0,1,0,,0,,0,,,0,0101,,,0,,0000,0,,,11,000,,,000,110,111]

    4067: ;(FREE)
    GOBUT120A1:
          SETUP,  RETURN/TEST120A2,                   ;EXEC SUBR WHICH:
                                                      ;(1) D<15:12>  -> IR<15:12>
```

```
                NEXT,   CALL[D15-12]                        ;(2) BUT(IR15-12) INTO ZTARGET---
        (4067)  DCS[0,00,0,0,0,0]  BM[0100,,00,11,,01,01,,101,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,011,100]

        ;CHECK BIT <11:06> = "1010 10"
        4655:
        TEST120A2:
                P0,     LOAD-ENUA(ZTARGET412),              ;INSTRS-E78 OUTPUT FOR BIT <11:06> = "101 010"
                        LOAD-ERROR(TEST120A2),              ;ERROR DIRECTORY KEY
                        DCS-CTR(C6,);                       ;COMPARE AT TARGET
                        BUMP-VERIFY,                        ;COUNT
                NEXT,   J/GOBUT120A2                        ;
        (4655)  DCS[1,00,1,0,0,1]  BM[1001,,00,11,,11,00,,001,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,111,000]

        4070:   ;(FREE)
        GOBUT120A2:
                SETUP,  RETURN/TEST120A3,                   ;EXEC SUBR WHICH:
                                                            ;(1) D<11:06> => IR<11:06>
                NEXT,   CALL[D11-06]                        ;(2) BUT(INSTRS) INTO ZTARGET---
        (4070)  DCS[0,00,0,0,0,0]  BM[0100,,00,11,,01,01,,100,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,011,110]



        ;CHECK BIT<05:00> = "10 1010"
        4654:
        TEST120A3:

                P0,     LOAD-ENUA(ZTARGET425),              ;INSTRS-E88 OUTPUT FOR BIT<05:00>="101 010"
                        LOAD-ERROR(TEST120A3),              ;ERROR DIRECTORY KEY
                        DCS-CTR[C6,],                       ;COMPARE AT TARGET
                NEXT,   J/GOBUT120A3                        ;
        (4654)  DCS[1,00,1,0,0,0]  BM[1001,,00,11,,11,00,,010,,101,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,111,001]

        4071:   ;(FREE)
        GOBUT120A3:
                SETUP,  RETURN/TEST120B1,                   ;EXEC SUBR WHICH:
                                                            ;(1) D<05:00> => IR<05:00>
                NEXT,   CALL[D05-00]                        ;(2) BUT[INSTRS] INTO ZTARGET---
        (4071)  DCS[0,00,0,0,0,0]  BM[0100,,00,11,,01,01,,011,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,100,000]




        ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        ;TESTS 120B 1-4 VERIFIES THAT WITH:
        ;ALU=(A-AND-B), A=(052525), B=(177777), THEN D=(052525)
        4653:
        TEST120B1:
                P0,     LOAD-ENUA(ZTARGET405),              ;BIT<15:12> = "0101"
                        LOAD-ERROR(TEST120B1),              ;ERROR DIRECTORY KEY
                        DCS-CTR(C6,);                       ;COMPARE AT TARGET
```



```
                NEXT,   J/ALU120B1                          ;
        (4653)  DCS[1,00,1,0,0,0]  BM[1001,,00,11,,11,00,,000,,101,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,111,010]

        4072:   ;(FREE)
        ALU120B1:

                P2-T,   D_A-AND-B, D[C]_CINMUX,             ;ALU=(A-AND-B), D[C]=CIN=(0)
                        BUS-A_ASPRI(C052525),               ;A=(052525)
                        BUS-B_CSPD(C177777),                ;B=(177777)
                NEXT,   J/GOBUT120B1                        ;D=(052525)
        (4072)  DCS[0,00,0,0,0,0]  BM[1011,,10,00,,11,01,,111,,000,,,0,1,0,,0,,0,,,0,0101,,,0,,0000,0,,,11,000,,,000,111,011]

        4073:   ;(FREE)
        GOBUT120B1:
                SETUP,  RETURN/TEST120B2,                   ;EXEC SUBR WHICH:
                                                            ;(1) D<15:12> => IR<15:12>
                NEXT,   CALL[D15-12]                        ;(2) BUT(IR15-12) INTO ZTARGET---
        (4073)  DCS[0,00,0,0,0,0]  BM[0100,,00,11,,01,01,,010,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,011,100]

        ;CHECK BIT<11:06> = "0101 01"
        4652:
        TEST120B2:
                P0,     LOAD-ENUA(ZTARGET405),              ;INSTRS-E78 OUTPUT FOR BIT<11:06>="010 101"
                        LOAD-ERROR(TEST120B2),              ;ERROR DIRECTORY KEY
                        DCS-CTR[C6,];                       ;COMPARE AT TARGET
                        BUMP-VERIFY,                        ;COUNT
                NEXT,   J/GOBUT120B2                        ;
        (4652)  DCS[1,00,1,0,0,1]  BM[1001,,00,11,,11,00,,000,,101,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,111,100]

        4074:   ;(FREE)
        GOBUT120B2:
                SETUP,  RETURN/TEST120B3,                   ;EXEC SUBR WHICH:
                                                            ;(1) D<11:06> => IR<11:06>
                NEXT,   CALL(D11-06]                        ;(2) BUT(INSTRS) INTO ZTARGET---
        (4074)  DCS[0,00,0,0,0,0]  BM[0100,,00,11,,01,01,,001,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,011,110]



        ;CHECK BIT<05:00> = "01 0101"
        4651:
        TEST120B3:

                P0,     LOAD-ENUA(ZTARGET432),              ;INSTRS-E88 OUTPUT FOR BIT <05:00>="010 101"
                        LOAD-ERROR(TEST120B3),              ;ERROR DIRECTORY KEY
                        DCS-CTR[C6,],                       ;COMPARE AT TARGET
                NEXT,   J/GOBUT120B3                        ;

        (4651)  DCS[1,00,1,0,0,0]  BM[1001,,00,11,,11,00,,011,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,111,101]

        4075:   ;(FREE)
        GOBUT120B3:
                SETUP,  RETURN/TEST120B4,                   ;EXEC SUBR WHICH:
                                                            ;(1) D<05:00> => IR<05:00>
```

```
                NEXT,   CALL[D05-00]                            ;(2) BUT(INSTRS) INTO ZTARGET---
  (4075)  DCS[0,00,0,0,0,0]  BM[0100,,00,11,,01,01,,000,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,100,000]

        ;CHECK THAT D[C] GOT A (0) FROM "CINMUX" = (0)
        4650:
        TEST120B4:
                P0,     LOAD-ENUA(ZTARGET402),                  ;BIT<00> = D[C] = (0)
                        LOAD-ERROR(TEST120B4),                  ;ERROR DIRECTORY KEY
                        DCS-CTR(C3,),                           ;COMPARE AT TARGET
                        BUMP-VERIFY,                            ;COUNT
                NEXT,   J/GOBUT120B4                            ;
  (4650)  DCS[1,00,1,0,0,1]  BM[1100,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,111,110]

        4076:  ;(FREE)
        GOBUT120B4:
                SETUP,  RETURN/SCOPE120B,                       ;RETURN TO SCOPE LOOP TEST WORD
                NEXT,   GOTO-PAGE(7),                           ;BUT TABLE
                        J/BUTD[C]A                              ;D[C]IN IN BIT <00>
  (4076)  DCS[0,00,0,0,0,0]  BM[0100,,00,00,,01,11,,111,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,100,100]

        4077:  ;(FREE)
        SCOPE120B:
                NEXT,   BUTD[SCOPE],                            ;NO ERROR:  "TEST121A1" (+1. WORDS)
                        J/TEST121A1                             ;   ERROR:  "ALU120A1"  (-15. WORDS)
  (4077)  DCS[0,00,0,1,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,110,110,101]





        ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        ;THIS NEXT SET OF 16, TESTS EXERCIZES THE ALU FUNCTION
        ;"A-XOR-B", AND THE CARRYOUT FUNCTIONS OF "ALU00" AND "ALU07" INTO D[C]

        ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -


        ;TESTS 121A 1-4 VERIFIES THAT WITH:
        ;ALU=(A-XOR-B), A=(000000), B=(052525), THEN D=(052525)
        4665:
        TEST121A1:
                P0,     LOAD-ENUA(ZTARGET408),                  ;BIT<15:12> = "0101"
                        LOAD-ERROR(TEST121A1),                  ;ERROR DIRECTORY KEY
                        DCS-CTR(C6,),                           ;COMPARE AT TARGET
                NEXT,   J/ALU121A1                              ;
  (4665)  DCS[1,00,1,0,0,0]  BM[1001,,00,11,,11,00,,000,,101,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,111,000,110]

        4706:
        ALU121A1:
                P0,     BUMP-VERIFY,                            ;COUNT
                P2-T,   0_A-XOR-B, D[C]_ALU00,                  ;ALU=(A-XOR-B), D[C]=ALU00=(1)
```



```
                        BUS-A_ASPHI(C000000),                   ;A=(000000)
                        BUS-B_CSPD(C052525),                    ;B=(052525)
                NEXT,   J/GOBUT121A1                            ;D=(052525)
  (4706)  DCS[0,00,0,0,0,1]  BM[0110,,10,00,,11,01,,100,,010,,,0,1,0,,0,,0,,,0,0111,,,0,,0000,0,,,11,000,,,001,000,011]

        4103:  ;(FREE)
        GOBUT121A1:
                SETUP,  RETURN/TEST121A2,                       ;EXEC SUBR WHICH:
                                                                ;(1) D<15:12> -> IR<15:12>
                NEXT,   CALL[D15-12]                            ;(2) BUT(IR15-12) INTO ZTARGET---
  (4103)  DCS[0,00,0,0,0,0]  BM[0100,,00,11,,01,11,,110,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,011,100]

        ;CHECK BIT<11:06> = "0101 01"
        4676:
        TEST121A2:
                P0,     LOAD-ENUA(ZTARGET405),                  ;INSTRS-ZTS OUTPUT FOR BIT<11:06>="010 101"
                        LOAD-ERROR(TEST121A2),                  ;ERROR DIRECTORY KEY
                        DCS-CTR(C6,),                           ;COMPARE AT TARGET
                        BUMP-VERIFY,                            ;COUNT
                NEXT,   J/GOBUT121A2                            ;
  (4676)  DCS[1,00,1,0,0,1]  BM[1001,,00,11,,11,00,,000,,101,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,000,100]

        4104:  ;(FREE)
        GOBUT121A2:
                SETUP,  RETURN/TEST121A3,                       ;EXEC SUBR WHICH:
                                                                ;(1) D<11:06> -> IR<11:06>
                NEXT,   CALL[D11-06]                            ;(2) BUT(INSTRS) INTO ZTARGET---
  (4104)  DCS[0,00,0,0,0,0]  BM[0100,,00,11,,01,11,,101,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,011,110]



        ;CHECK BIT<05:00> = "01 0101"
        4675:
        TEST121A3:
                P0,     LOAD-ENUA(ZTARGET432),                  ;INSTRS-ZSS OUTPUT FOR BIT<05:00>="010 101"
                        LOAD-ERROR(TEST121A3),                  ;ERROR DIRECTORY KEY
                        DCS-CTR(C6,),                           ;COMPARE AT TARGET
                NEXT,   J/GOBUT121A3                            ;
  (4675)  DCS[1,00,1,0,0,0]  BM[1001,,00,11,,11,00,,011,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,000,101]

        4105:  ;(FREE)
        GOBUT121A3:
                SETUP,  RETURN/TEST121A4,                       ;EXEC SUBR WHICH:
                                                                ;(1) D<05:00> -> IR<05:00>
                NEXT,   CALL[D05-00]                            ;(2) BUT(INSTRS) INTO ZTARGET---
  (4105)  DCS[0,00,0,0,0,0]  BM[0100,,00,11,,01,11,,100,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,100,000]

        ;CHECK THAT D[C] GOT A (1) FROM "ALU00"
        4674:

        TEST121A4:
                P0,     LOAD-ENUA(ZTARGET403),                  ;BIT<01> = D[C] = (1)
```

```
                LOAD-ERROR(TEST121A4),                  ;ERROR DIRECTORY KEY
                DCS-CTR(C3.),                           ;COMPARE AT TARGET
                BUMP-VERIFY,                            ;COUNT
        NEXT,   J/GOBUT121A4                            ;
  (4674)  DCS[1,00,1,0,0,1]  BM[1100,,00,11,,11,00,,000,,011,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,000,110]

    4106:  ;(FREE)
    GOBUT121A4:
        SETUP,  RETURN/TEST121B1,                       ;RETURN TO START OF NEXT SUBTEST
        NEXT,   GOTO-PAGE(7),                           ;BUT TABLE
                J/BUTD[C]B                              ;D[C]B IN BIT<01>
  (4106)  DCS[0,00,0,0,0,0]  BM[0100,,00,11,,01,10,,011,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,101,000]




    ; - - - - - - - - - - - - - - - - - - - - - - - - - - -

    ;TESTS 121B 1-4 VERIFIES THAT WITH:
    ;ALU=(A-XOR-B), A=(177777), B=(052525), THEN D=(125252)
    4663:
    TEST121B1:
        P0,     LOAD-ENUA(ZTARGET412),                  ;BIT<15:12> = "1010"
                LOAD-ERROR(TEST121B1),                  ;ERROR DIRECTORY KEY
                DCS-CTR(C6.),                           ;COMPARE AT TARGET
        NEXT,   J/ALU121B1                              ;
  (4663)  DCS[1,00,1,0,0,0]  BM[1001,,00,11,,11,00,,001,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,000,111]

    4107:  ;(FREE)
    ALU121B1:
        P0,     BUMP-VERIFY,                            ;COUNT
        P2-T,   D_A-XOR-B, D[C]_ALU00,                  ;ALU=(A-XOR-B), D[C]=ALU00=(0)
                BUS-A_ASPHI(C177777),                   ;A=(177777)
                BUS-B_CSPD(C052525),                    ;B=(052525)
        NEXT,   J/GOBUT121B1                            ;D=(125252)
  (4107)  DCS[0,00,0,0,0,1]  BM[0110,,10,00,,11,01,,101,,010,,,0,1,0,,0,,0,,,0,0111,,,0,,0000,0,,,11,000,,,001,001,000]

    4110:  ;(FREE)
    GOBUT121B1:
        SETUP,  RETURN/TEST121B2,                       ;EXEC SUBR WHICH:
                                                        ;(1) D<15:12> => IR<15:12>
        NEXT,   CALL[D15-12]                            ;(2) BUT(IR15-12) INTO ZTARGET---
  (4110)  DCS[0,00,0,0,0,0]  BM[0100,,00,11,,01,10,,010,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,011,100]

    ;CHECK BIT<11:06> = "1010 10"
    4662:
    TEST121B2:
        P0,     LOAD-ENUA(ZTARGET412),                  ;INSTRS-E78 OUTPUT FOR BIT<11:06>="101 010"
                LOAD-ERROR(TEST121B2),                  ;ERROR DIRECTORY KEY
                DCS-CTR(C6.),                           ;COMPARE AT TARGET
                BUMP-VERIFY,                            ;COUNT
        NEXT,   J/GOBUT121B2                            ;
```



```
  (4662)  DCS[1,00,1,0,0,1]  BM[1001,,00,11,,11,00,,001,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,001,001]

    4111:  ;(FREE)
    GOBUT121B2:
        SETUP,  RETURN/TEST121B3,                       ;EXEC SUBR WHICH:
                                                        ;(1) D<11:06> => IR<11:06>
        NEXT,   CALL[D11-06]                            ;(2) BUT(INSTRS) INTO ZTARGET---
  (4111)  DCS[0,00,0,0,0,0]  BM[0100,,00,11,,01,10,,001,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,011,110]



    ;CHECK BIT<05:00> = "10 1010"
    4661:
    TEST121B3:

        P0,     LOAD-ENUA(ZTARGET425),                  ;INSTRS-E88 OUTPUT FOR BIT<05:00>="101 010"
                LOAD-ERROR(TEST121B3),                  ;ERROR DIRECTORY KEY
                DCS-CTR(C6.),                           ;COMPARE AT TARGET
        NEXT,   J/GOBUT121B3                            ;
  (4661)  DCS[1,00,1,0,0,0]  BM[1001,,00,11,,11,00,,010,,101,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,001,010]

    4112:  ;(FREE)
    GOBUT121B3:
        SETUP,  RETURN/TEST121B4,                       ;EXEC SUBR WHICH:
                                                        ;(1) D<05:00> => IR<05:00>
        NEXT,   CALL[D05-00]                            ;(2) BUT(INSTRS) INTO ZTARGET---
  (4112)  DCS[0,00,0,0,0,0]  BM[0100,,00,11,,01,10,,000,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,100,000]

    ;CHECK THAT D[C] GOT A (0) FROM "ALU00"
    4660:
    TEST121B4:
        P0,     LOAD-ENUA(ZTARGET401),                  ;BIT<01> = D[C] = (0)
                LOAD-ERROR(TEST121B4),                  ;ERROR DIRECTORY KEY
                DCS-CTR(C3.),                           ;COMPARE AT TARGET
                BUMP-VERIFY,                            ;COUNT
        NEXT,   J/GOBUT121B4                            ;
  (4660)  DCS[1,00,1,0,0,1]  BM[1100,,00,11,,11,00,,000,,001,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,001,011]

    4113:  ;(FREE)
    GOBUT121B4:
        SETUP,  RETURN/SCOPE121C,                       ;RETURN TO SCOPE LOOP TEST WORD
        NEXT,   GOTO-PAGE(7),                           ;BUT TABLE
                J/BUTD[C]B                              ;D[C]B IN BIT<01]
  (4113)  DCS[0,00,0,0,0,0]  BM[0100,,00,00,,10,01,,100,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,101,000]



    4114:  ;(FREE)
    SCOPE121C:
        NEXT,   BUTD[SCOPE],                            ;NO ERROR:  "TEST121C1"   (+1. WORDS)
                J/TEST121C1                             ;   ERROR:  "ALU121A1"  (-17. WORDS)
```

```
      (4114)  DCS[0,00,0,1,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,111,000,111]
6979
6980
6981
6982     ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
6983
6984     !TESTS 121C 1-4 VERIFIES THAT WITH:
6985     !ALU=(A-XOR-B), A=(000000), B=(125252), THEN D=(125252)
6986     4707!
6987     TEST121C1:
6988           P0,     LOAD-ENUA(ETARGET412),                  !BIT<15:12> = "1010"
6989                   LOAD-ERROR(TEST121C1),                  !ERROR DIRECTORY KEY
6990                   DCS-CTR(C6,),                           !COMPARE AT TARGET
6991           NEXT,   J/ALU121C1                              !
      (4707)  DCS[1,00,1,0,0,0]  BM[1001,,00,11,,11,00,,001,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,110,111,000]
6992
6993     4670!
6994     ALU121C1:
6995           P0,     BUMP-VERIFY,                            !COUNT
6996           P2=T,   D_A-XOR-B, D[C]_ALU07,                  !ALU=(A-XOR-B), D[C]=ALU07=(1)
6997                   BUS-A_ASPHI(C000000),                   !A=(000000)
6998                   BUS-B_CSPD(C125252),                    !B=(125252)
6999           NEXT,   J/GOBUT121C1                            !D=(125252)
      (4670)  DCS[0,00,0,0,0,1]  BM[0110,,10,00,,11,01,,100,,011,,,0,1,0,,0,,0,,,0,0110,,,0,,0000,0,,,11,000,,,001,001,101]
7000
7001     4115!  !(FREE)
7002     GOBUT121C1:
7003           SETUP,  RETURN/TEST121C2,                       !EXEC SUBR WHICH:
7004                                                           !(1) D<15:12> => IR<15:12>
7005           NEXT,   CALL[D15-12]                            !(2) BUT(IR15-12) INTO ETARGET---
      (4115)  DCS[0,00,0,0,0,0]  BM[0100,,00,11,,00,00,,101,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,011,100]
7006
7007     !CHECK BIT<11:06> = "1010 10"
7008     4605!
7009     TEST121C2:
7010           P0,     LOAD-ENUA(ETARGET413),                  !INSTRS-E78 OUTPUT FOR BIT<11:06>="101 010"
7011                   LOAD-ERROR(TEST121C2),                  !ERROR DIRECTORY KEY
7012                   DCS-CTR(C6,),                           !COMPARE AT TARGET
7013                   BUMP-VERIFY,                            !COUNT
7014           NEXT,   J/GOBUT121C2                            !
      (4605)  DCS[1,00,1,0,0,1]  BM[1001,,00,11,,11,00,,001,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,001,110]
7015
7016     4116!  !(FREE)
7017     GOBUT121C2:
7018           SETUP,  RETURN/TEST121C3,                       !EXEC SUBR WHICH:
7019                                                           !(1) D<11:06> => IR<11:06>
7020           NEXT,   CALL[D11-06]                            !(2) BUT(INSTRS) INTO ETARGET---
      (4116)  DCS[0,00,0,0,0,0]  BM[0100,,00,11,,10,01,,111,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,011,110]
7021
7022
7023     !CHECK BIT<05:00> = "10 1010"
7024     4717!
```



```
7025     TEST121C3:
7026
7027           P0,     LOAD-ENUA(ETARGET425),                  !INSTRS-E88 OUTPUT FOR BIT<05:00>="101 010"
7028                   LOAD-ERROR(TEST121C3),                  !ERROR DIRECTORY KEY
7029                   DCS-CTR(C6,),                           !COMPARE AT TARGET
7030           NEXT,   J/GOBUT121C3                            !
      (4717)  DCS[1,00,1,0,0,0]  BM[1001,,00,11,,11,00,,010,,101,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,001,111]
7031
7032     4117!  !(FREE)
7033     GOBUT121C3:
7034           SETUP,  RETURN/TEST121C4,                       !EXEC SUBR WHICH:
7035                                                           !(1) D<05:00> => IR<05:00>
7036           NEXT,   CALL[D05-00]                            !(2) BUT(INSTRS) INTO ETARGET---
      (4117)  DCS[0,00,0,0,0,0]  BM[0100,,00,11,,10,01,,110,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,100,000]
7037
7038     !CHECK THAT D[C] GOT A (1) FROM "ALU07"
7039     4716!
7040     TEST121C4:
7041           P0,     LOAD-ENUA(ETARGET417),                  !BIT<02> = D[C] = (1)
7042                   LOAD-ERROR(TEST121C4),                  !ERROR DIRECTORY KEY
7043                   DCS-CTR(C3,),                           !COMPARE AT TARGET
7044                   BUMP-VERIFY,                            !COUNT
7045           NEXT,   J/GOBUT121C4                            !
      (4716)  DCS[1,00,1,0,0,1]  BM[1100,,00,11,,11,00,,001,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,010,000]
7046
7047     4120!  !(FREE)
7048     GOBUT121C4:
7049           SETUP,  RETURN/TEST121D1,                       !RETURN TO START OF NEXT SUBTEST
7050           NEXT,   GOTO-PAGE(7),                           !BUT TABLE
7051                   J/BUTD[C]C                              !D[C]H IN BIT<02>
      (4120)  DCS[0,00,0,0,0,0]  BM[0100,,00,11,,10,10,,111,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,001,000]
7052
7053
7054
7055
7056     ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
7057
7058     !TESTS 121D 1-4 VERIFIES THAT WITH:
7059     !ALU=(A-XOR-B), A=(177777), B=(125252), THEN D=(052525)
7060     4727!
7061     TEST121D1:
7062           P0,     LOAD-ENUA(ETARGET405),                  !BIT<15:12> = "0101"
7063                   LOAD-ERROR(TEST121D1),                  !ERROR DIRECTORY KEY
7064                   DCS-CTR(C6,),                           !COMPARE AT TARGET
7065           NEXT,   J/ALU121D1                              !
      (4727)  DCS[1,00,1,0,0,0]  BM[1001,,00,11,,11,00,,000,,101,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,010,001]
7066
7067     4121!  !(FREE)
7068     ALU121D1:
7069           P0,     BUMP-VERIFY,                            !COUNT
7070           P2=T,   D_A-XOR-B, D[C]_ALU07,                  !ALU=(A-XOR-B), D[C]=ALU07=(0)
7071                   BUS-A_ASPHI(C177777),                   !A=(177777)
```

```
                        BUS-B_CSPD(C125252),                    ;B=(125252)
                NEXT,   J/GOBUT121D1                            ;D=(052525)
(4121)  DCS[0,00,0,0,0,1]  BN[0110,,10,00,,11,01,,101,,011,,,0,1,0,,0,,0,,,0,0110,,,0,,0000,0,,,11,000,,,001,010,010]

        4122: ;(FREE)
        GOBUT121D1:
                SETUP,  RETURN/TEST121D2,                       ;EXEC SUBR WHICH:
                                                                ;(1) D<15:12> -> IR<15:12>
                NEXT,   CALL[D15-12]                            ;(2) BUT[IR15-12] INTO ZTARGET---
(4122)  DCS[0,00,0,0,0,0]  BN[0100,,00,11,,10,10,,110,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,011,100]

        ;CHECK BIT<11:06> = "010: 01"
        4726:
        TEST121D2:
                P0,     LOAD-ENUA(ZTARGET405),                  ;INSTRS-ZTS OUTPUT FOR BIT<11:06>="010 101"
                        LOAD-ERROR(TEST121D2),                  ;ERROR DIRECTORY KEY
                        DCS-CTR(C6,),                           ;COMPARE AT TARGET
                        BUMP-VERIFY,                            ;COUNT
                NEXT,   J/GOBUT121D2                            ;
(4726)  DCS[1,00,1,0,0,1]  BN[1001,,00,11,,11,00,,000,,101,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,010,011]

        4123: ;(FREE)
        GOBUT121D2:
                SETUP,  RETURN/TEST121D3,                       ;EXEC SUBR WHICH:
                                                                ;(1) D<11:06> -> IR<11:06>
                NEXT,   CALL[D11-06]                            ;(2) BUT(INSTRS) INTO ZTARGET---
(4123)  DCS[0,00,0,0,0,0]  BN[0100,,00,11,,10,11,,111,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,011,110]



        ;CHECK BIT<05:00> = "01 010:"
        4737:
        TEST121D3:

                P0,     LOAD-ENUA(ZTARGET432),                  ;INSTRS-ZSS OUTPUT FOR BIT<05:00>="010 101"
                        LOAD-ERROR(TEST121D3),                  ;ERROR DIRECTORY KEY
                        DCS-CTR(C6,),                           ;COMPARE AT TARGET
                NEXT,   J/GOBUT121D3                            ;
(4737)  DCS[1,00,1,0,0,0]  BN[1001,,00,11,,11,00,,011,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,010,100]

        4124: ;(FREE)
        GOBUT121D3:
                SETUP,  RETURN/TEST121D4,                       ;EXEC SUBR WHICH:
                                                                ;(1) D<05:00> -> IR<05:00>
                NEXT,   CALL[D05-00]                            ;(2) BUT(INSTRS) INTO ZTARGET---
(4124)  DCS[0,00,0,0,0,0]  BN[0100,,00,11,,10,11,,110,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,100,000]

        ;CHECK THAT D(C) GOT A (0) FROM "ALU07"
        4736:
        TEST121D4:
                P0,     LOAD-ENUA(ZTARGET413),                  ;BIT<02> = D[C] = (0)
```



```
                        LOAD-ERROR(TEST121D4),                  ;ERROR DIRECTORY KEY
                        DCS-CTR(C3,),                           ;COMPARE AT TARGET
                        BUMP-VERIFY,                            ;COUNT
                NEXT,   J/GOBUT121D4                            ;
(4736)  DCS[1,00,1,0,0,1]  BN[1100,,00,11,,11,00,,001,,011,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,010,101]

        4125: ;(FREE)
        GOBUT121D4:
                SETUP,  RETURN/SCOPE121D,                       ;RETURN TO SCOPE LOOP TEST WORD
                NEXT,   GOTO-PAGE(7),                           ;BUT TABLE
                        J/BUTD[C]C                              ;D[C]H IN BIT<02>
(4125)  DCS[0,00,0,0,0,0]  BN[0100,,00,00,,10,10,,110,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,001,000]

        4126: ;(FREE)
        SCOPE121D:
                P0,     BUMP-VERIFY,                            ;COUNT
                NEXT,   BUTO[SCOPE],                            ;NO ERROR: "TEST122A1" (+1, WORDS)
                        J/TEST122A1                             ;   ERROR:  "ALU121C1" (-17, WORDS)
(4126)  DCS[0,00,0,1,0,1]  BN[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,110,111,001]








        ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        ;THIS NEXT SET OF 8, TESTS CHECK THE ALU FUNCTIONS "A"
        ;AND "A-IOR-B", THE FUNCTION D[C]=CINMUX=(1), AND ALSO
        ;THE BUT(D<14:00>=ZERO!D<15>) BUT FOR D=(0)(00000) AND
        ;D=(1)(25252)
        4671:
        TEST122A1:
                P0,     LOAD-ENUA(ZTARGET434),                  ;INSTRS FOR IR=(000000)
                        LOAD-ERROR(TEST122A1),                  ;ERROR DIRECTORY KEY
                        DCS-CTR(C6,),                           ;COMPARE AT TARGET
                NEXT,   J/ALU122A1                              ;
(4671)  DCS[1,00,1,0,0,0]  BN[1001,,00,11,,11,00,,011,,100,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,110,111,010]

        4672:
        ALU122A1:
                P0,     BUMP-VERIFY,                            ;COUNT
                P2-T,   D_A-IOR-B, D[C]_CINMUX,                 ;ALU=(A-IOR-B), D[C]=CINMUX=(1)
                        BUS-A_ASPHI(C000000),                   ;A=(000000)

                        BUS-B_BSPHI(C000000),                   ;B=(000000)
                NEXT,   J/GOBUT122A1                            ;D=(000000)
(4672)  DCS[0,00,0,0,0,1]  BN[1110,,01,11,,11,01,,100,,000,,,0,1,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,010,111]

        4127: ;(FREE)
        GOBUT122A1:
                SETUP,  RETURN/TEST122A2,                       ;EXEC SUBR WHICH:
```

```
                                                            ;(1) D -> IR
         NEXT,   CALL[DZERO]                                ;(2) BUT(INSTRS) INTO STARGET---
  (4127)  DCS[0,00,0,0,0,0]  BM[0100,,00,10,,11,11,,000,,11],,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,100,010]



  ;CHECK THAT D(C) GOT A (1) FROM CINMUX=(1)
  4570:
  TEST122A2:
         P0,     LOAD-ENUA[ZTARGET403],                     ;BIT<00> = D[C] = (1)
                 LOAD-ERROR[TEST122A2],                     ;ERROR DIRECTORY KEY
                 DCS-CTR[C3,],                              ;COMPARE AT TARGET
         NEXT,   J/GOBUT122A2                               ;
  (4570)  DCS[1,00,1,0,0,0]  BM[1100,,00,11,,11,00,,000,,011,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,011,000]

  4130:  ;(FREE)
  GOBUT122A2:
         SETUP,  RETURN/TEST122A3,                          ;RETURN TO START OF NEXT SUBTEST
         NEXT,   GOTO-PAGE(7),                              ;BUT TABLE
                 J/BUTD[C]A                                 ;D[C]R IN BIT <00>
  (4130)  DCS[0,00,0,0,0,0]  BM[0100,,00,11,,10,11,,100,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,100,100]



  ;CHECK THAT D<15>=0, [D<14:00>=ZERO]=1 WHEN D=(000000)
  4734:
  TEST122A3:
         P0,     LOAD-ENUA[ZTARGET402],                     ;SETUP FOR D<15:00>=ZERO
                 LOAD-ERROR[TEST122A3],                     ;ERROR DIRECTORY KEY
                 DCS-CTR[C3,],                              ;COMPARE AT TARGET
         NEXT,   J/GOBUT122A3                               ;
  (4734)  DCS[1,00,1,0,0,0]  BM[1100,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,011,001]

  4131:  ;(FREE)
  GOBUT122A3:
         SETUP,  RETURN/TEST122A4,                          ;RETURN TO START OF NEXT SUBTEST
         NEXT,   GOTO-PAGE(7),                              ;BUT TABLE
                 J/BUTD-IS-ZERO                             ;BIT<1|0> = D<15>$ D<14:00>=ZERO
  (4131)  DCS[0,00,0,0,0,0]  BM[0100,,00,11,,10,11,,101,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,100,001]

  ;CHECK THAT D<15>=(1), [D<14:00>=ZERO]=(0) WHEN D=(125252)
  4735:
  TEST122A4:
         P0,     LOAD-ENUA[ZTARGET401],                     ;SETUP FOR D<15:00>=(125252)
                 LOAD-ERROR[TEST122A4],                     ;ERROR DIRECTORY KEY
                 DCS-CTR[C4,],                              ;COMPARE AT TARGET
                 BUMP-VERIFY,                               ;COUNT
         NEXT,   J/SETD122A4                                ;
  (4735)  DCS[1,00,1,0,0,1]  BM[1011,,00,11,,11,00,,000,,001,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,011,010]

  4132:  ;(FREE)
```



```
  SETD122A4:
         P2-T,   D_CSPD[C125252], D[C]_0,                   ;SETUP D FOR TEST
         NEXT,   J/GOBUT122A4                               ;
  (4132)  DCS[0,00,0,0,0,0]  BM[1010,,10,00,,00,00,,000,,000,,,0,1,0,,0,,0,,,0,0110,,,0,,0000,0,,,11,000,,,001,011,011]

  4133:  ;(FREE)
  GOBUT122A4:
         SETUP,  RETURN/SCOPE122A,                          ;RETURN TO SCOPE LOOP TEST WORD
         NEXT,   GOTO-PAGE(7),                              ;BUT TABLE
                 J/BUTD-IS-ZERO                             ;BIT<1|0> = D15$[D<14:00>=ZERO]
  (4133)  DCS[0,00,0,0,0,0]  BM[0100,,00,00,,10,11,,100,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,100,001]



  4134:  ;(FREE)
  SCOPE122A:
         P0,     BUSDIN_EXIT-[T],                           ;RESET PROC UCON
                 EN-CLK-IR[15-00],                          ;
         NEXT,   BUTD[SCOPE],                               ;NO ERROR:  "TEST130A1"   (+1, WORDS)
                 J/TEST130A1                                ;   ERROR:  "ALU122A1"  (-0, WORDS)
  (4134)  DCS[0,00,0,1,0,0]  BM[0000,,00,00,,00,01,,000,,100,,,0,0,0,,0,,0,,,1,1001,,,0,,0000,0,,,11,000,,,110,111,011]





  ;.PAGE==========================================================================================================

  .TOC   =   TEST130-136:  ALU ARITHMETIC FUNCTION/CARRY LOOKAHEAD TESTS

  ;****************************************************************************************
  ;*
  ;*     TESTS:  130 - 136                                   UWORDS:  127 + 160
  ;*
  ;*     FUNCTIONS:
  ;*
  ;*     ALU ARITHMETIC FUNCTION DECODE, INTERNAL CARRIES, CARRYOUTS, CARRY LOOKAHEAD,
  ;*
  ;****************************************************************************************

  ;
  ;      SUMMARY OF ALU ARITHMETIC / CARRY LOOKAHEAD TESTS:
  ;
  ;      TEST             OPERANDS EMPLOYED:
  ;      NUMB             (A/B)+(B/A)+(CIN)=(COUT)(D)      ALU FUNCTION

  ;      ----             ---------------------------      ------------------
  ;
  ;      130A             (0101)+(0101)+(0)=(0)(1010)      A-PLUS-B-PLUS-0
  ;      130B             (1010)+(1010)+(1)=(1)(0101)      A-PLUS-B-PLUS-1
  ;
  ;      131A             (1010)+(0101)+(0)=(0)(1111)      A-PLUS-B-PLUS-PS[C]
  ;      131B             (0101)+(1010)+(0)=(0)(1111)      DIVIDE/D[C]=0/A-PLUS-B-PLUS-0
```

```
        !
        !       132A            (1010)-(1010)-(0)=(1)(0000)     A-MINUS-B-MINUS-0
        !       132B            (0101)-(0101)-(0)=(1)(0000)     DIVIDE/D[C]=1/A-MINUS-B-MINUS-0
        !
        !       133A            (1000)+(1000)+(0)=(1)(0000)     A-PLUS-B-PLUS-D[C]
        !       133B            (0111)+(0111)+(1)=(0)(1111)     A-PLUS-NOT-B-PLUS-D[C]
        !
        !       134A            (0100)+(1100)+(0)=(1)(0000)     A-PLUS-B-PLUS-D[C]
        !       134B            (1011)+(0011)+(1)=(0)(1111)     A-PLUS-NOT-B-PLUS-D[C]
        !
        !       135A            (1010)+(0110)+(0)=(1)(0000)     A-PLUS-B-PLUS-D[C]
        !       135B            (0101)+(1001)+(1)=(0)(1111)     A-PLUS-NOT-B-PLUS-D[C]
        !
        !       136A            (0101)+(1011)+(0)=(1)(0000)     A-PLUS-B-PLUS-D[C]
        !       136B            (1010)+(0100)+(1)=(0)(1111)     A-PLUS-NOT-B-PLUS-D[C]
        !
        !
        ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        !CHECK INTERNAL ALU CARRIES WITH:     (052525)+(052525)+(0)=(125252)
        !ALSO CHECK ALU FUNCTION "A-PLUS-B-PLUS-0", D[C]_COUT15=(0)
        !
        4673:
        TEST130A1:
                P0,     LOAD-ENUA(ZTARGET434),          !SETUP FOR IR=(000000)/BUTINSTRS TEST
                        LOAD-ERROR(TEST130A1),          !ERROR DIRECTORY KEY
                        DCS-CTR(C8,),                   !COMPARE AT TARGET
                NEXT,   J/ARITH130A1                    !
   (4673)  DCS[1,00,1,0,0,0]  BM[0111,,00,11,,11,00,,011,,100,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,101,100,000]

        4540:
        ARITH130A1:
                P2-T,   D_A-PLUS-B-PLUS-0,              !ALU=(A-PLUS-B), CIN=(0)
                        D[C]_COUT15,                    !GET CARRYOUT
                        BUS-A_ASPHI(C052525),           !A=(052525)
                        BUS-B_BSPHI(C052525),           !B=(052525)
                        SR_A-PLUS-B-PLUS-0,             !D=(125252), COUT15=(0)
                NEXT,   J/COMP130A1                     !
   (4540)  DCS[0,00,0,0,0,0]  BM[1001,,01,11,,11,01,,111,,110,,,0,1,1,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,011,101]

        4135:  !(FREE)
        COMP130A1:
                P0,     BUMP-VERIFY,                    !COUNT
                P2-T,   D_SR-XOR-CSPD(C125252),         !COMPARE RECEIVED: EXPECTED
                        SAVE-D[C],                      !SAVE CARRY
                NEXT,   J/GOBUT130A1                    !
   (4135)  DCS[0,00,0,0,0,1]  BM[0110,,10,00,,00,00,,000,,111,,,0,1,0,,0,,0,,,0,0110,,,0,,0000,0,,,11,000,,,001,011,110]

        4136:  !(FREE)
        GOBUT130A1:
                SETUP,  RETURN/TEST130A2,               !RETURN TO START OF NEXT SUBTEST
                NEXT,   CALL[DINTOIR-B]                 !GO PUT D => IR, BUT(INSTRS)
   (4136)  DCS[0,00,0,0,0,0]  BM[0100,,00,10,,11,01,,000,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,111,011]
```



```



        !CHECK THAT CARRYOUT OF BIT15 (COUT15) WAS CORRECT
        4550:
        TEST130A2:
                P0,     LOAD-ENUA(ZTARGET402),          !BIT<00> CLEAR
                        LOAD-ERROR(TEST130A2),          !ERROR DIRECTORY KEY
                        DCS-CTR(C3,),                   !COMPARE AT TARGET
                NEXT,   J/GOBUT130A2                    !
   (4550)  DCS[1,00,1,0,0,0]  BM[1100,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,011,111]

        4137:  !(FREE)
        GOBUT130A2:
                SETUP,  RETURN/TEST130B1,               !RETURN TO START OF NEXT SUBTEST
                NEXT,   GOTO-PAGE(7),                   !BUT TABLE
                        J/BUTD[C]A                      !D[C]=> COUT15 N IN BIT<00>
   (4137)  DCS[0,00,0,0,0,0]  BM[0100,,00,10,,10,11,,000,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,100,100]



        ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        !CHECK INTERNAL ALU CARRIES WITH: (125252)+(125252)+(1)=(052525)
        !ALSO CHECK ALU FUNCTION "A-PLUS-B-PLUS-1", D[C]_COUT15=(1)
        !
        4530:
        TEST130B1:
                P0,     LOAD-ENUA(ZTARGET434),          !SETUP FOR IR=(000000)/BUTINSTRS TEST
                        LOAD-ERROR(TEST130B1),          !ERROR DIRECTORY KEY
                        DCS-CTR(C8,),                   !COMPARE AT TARGET
                NEXT,   J/ARITH130B1                    !
   (4530)  DCS[1,00,1,0,0,0]  BM[0111,,00,11,,11,00,,011,,100,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,100,000]

        4140:  !(FREE)
        ARITH130B1:
                P2-T,   D_A-PLUS-B-PLUS-1,              !ALU=(A-PLUS-B), CIN=(1)
                        D[C]_COUT15,                    !GET CARRYOUT
                        BUS-A_ASPHI(C125252),           !A=(125252)
                        BUS-B_BSPHI(C125252),           !B=(125252)
                        SR_A-PLUS-B-PLUS-1,             !D=(052525), COUT15=(1)
                NEXT,   J/COMP130B1                     !
   (4140)  DCS[0,00,0,0,0,0]  BM[1100,,01,11,,11,01,,110,,110,,,0,1,1,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,100,001]

        4141:  !(FREE)
        COMP130B1:

                P0,     BUMP-VERIFY,                    !COUNT
                P2-T,   D_SR-XOR-CSPD(C052525),         !COMPARE RECEIVED: EXPECTED
                        SAVE-D[C],                      !SAVE CARRY
                NEXT,   J/GOBUT130B1                    !
   (4141)  DCS[0,00,0,0,0,1]  BM[0110,,10,00,,00,00,,000,,111,,,0,1,0,,0,,0,,,0,0111,,,0,,0000,0,,,11,000,,,001,100,010]

        4142:  !(FREE)
```

```
	GOBUT130B1:
		SETUP,	RETURN/TEST130B2,		;RETURN TO START OF NEXT SUBTEST
		NEXT,	CALL(DINTOIR-6)			;GO PUT D -> IR, BUT(INSTRS)
   (4147)  DCS[0,00,0,0,0,0]  BM[0100,,00,10,,10,11,,010,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,111,011]



	;CHECK THAT CARRYOUT OF BIT15 (COUT15) WAS CORRECT
	4532:
	TEST130B2:
		P0,	LOAD-ENUA(ZTARGET403),		;BIT<00> SET
			LOAD-ERROR(TEST130B2),		;ERROR DIRECTORY KEY
			DCS-CTR(C3,),			;COMPARE AT TARGET
		NEXT,	J/GOBUT130B2			;
   (4532)  DCS[1,00,1,0,0,0]  BM[1100,,00,11,,11,00,,000,,011,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,100,011]

	4143:	;(FREE)
	GOBUT130B2:
		SETUP,	RETURN/SCOPE130B,		;RETURN TO SCOPE LOOP TEST WORD
		NEXT,	GOTO-PAGE(7),			;BUT TABLE
			J/BUTD[C]A			;D[C]H# COUT15 H IN BIT<00>
   (4143)  DCS[0,00,0,0,0,0]  BM[0100,,00,00,,11,00,,100,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,100,100]



	4144:	;(FREE)
	SCOPE130B:
		NEXT,	BUTD[SCOPE],			;NO ERROR:  "TEST131A1"  (+1,WORDS)
			J/TEST131A1			;   ERROR:  "ARITH130A1" (-11,WORDS)
   (4144)  DCS[0,00,0,1,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,101,100,001]



	; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

	;CHECK INTERNAL ALU CARRIES WITH: (125252)+(052525)+(0)=(177777)
	;ALSO CHECK ALU FUNCTION "A-PLUS-B-PLUS-PS[C]", D[C]_COUT15=(0)
	;CIN=PS[C]=(0) FROM INITIALIZATION ROUTINE
	4541:
	TEST131A1:
		P0,	LOAD-ENUA(ZTARGET434),		;SETUP FOR IR=(000000)/BUTINSTRS TEST
			LOAD-ERROR(TEST131A1),		;ERROR DIRECTORY KEY
			DCS-CTR(C8,),			;COMPARE AT TARGET
		NEXT,	J/ARITH131A1			;
   (4541)  DCS[1,00,1,0,0,0]  BM[0111,,00,11,,11,00,,011,,100,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,101,010,000]

	4520:
	ARITH131A1:
		P2-T,	D_A-PLUS-B-PLUS-PS[C],		;ALU=(A-PLUS-B), CIN=PS[C]=(0)
			D[C]_COUT15,			;GET CARRYOUT
			BUS-A_ASPHI(C125252),		;A=(125252)
			BUS-B_CSPD(C052525),		;B=(052525)
```



```
			SR_A-PLUS-B-PLUS-PS[C],		;D=(177777), COUT15=(0)
		NEXT,	J/COMP131A1			;
   (4520)  DCS[0,00,0,0,0,0]  BM[0001,,10,00,,11,01,,110,,110,,,0,1,1,,0,,0,,,0,0111,,,0,,0000,0,,,11,000,,,001,100,101]

	4145:	;(FREE)
	COMP131A1:
		P0,	BUMP-VERIFY,			;COUNT
		P2-T,	D_SR-XOR-CSPD(C177777),		;COMPARE RECEIVED; EXPECTED
			SAVE-D[C],			;SAVE CARRY
		NEXT,	J/GOBUT131A1			;
   (4145)  DCS[0,00,0,0,0,1]  BM[0110,,10,00,,00,00,,000,,111,,,0,1,0,,0,,0,,,0,0101,,,0,,0000,0,,,11,000,,,001,100,110]

	4146:	;(FREE)
	GOBUT131A1:
		SETUP,	RETURN/TEST131A2,		;RETURN TO START OF NEXT SUBTEST
		NEXT,	CALL[DINTOIR-5]			;GO PUT D -> IR, BUT(INSTRS)
   (4146)  DCS[0,00,0,0,0,0]  BM[0100,,00,10,,11,00,,100,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,111,011]



	;CHECK THAT CARRYOUT OF BIT15 (COUT15) WAS CORRECT
	4544:
	TEST131A2:
		P0,	LOAD-ENUA(ZTARGET401),		;BIT<01> CLEAR
			LOAD-ERROR(TEST131A2),		;ERROR DIRECTORY KEY
			DCS-CTR(C3,),			;COMPARE AT TARGET
		NEXT,	J/GOBUT131A2			;
   (4544)  DCS[1,00,1,0,0,0]  BM[1100,,00,11,,11,00,,000,,001,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,100,111]

	4147:	;(FREE)
	GOBUT131A2:
		SETUP,	RETURN/TEST131B1,		;RETURN TO START OF NEXT SUBTEST
		NEXT,	GOTO-PAGE(7),			;BUT TABLE
			J/BUTD[C]B			;D[C]H# COUT15 H IN BIT<01>
   (4147)  DCS[0,00,0,0,0,0]  BM[0100,,00,10,,11,00,,110,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,101,000]



	; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

	;CHECK INTERNAL ALU CARRIES WITH:		(052525)+(125252)+(0)=(177777)
	;ALSO CHECK ALU FUNCTION "DIVIDE-STEP" = "A-PLUS-B" SINCE D[C]H=(0)
	;FROM ABOVE, D[C]_COUT15=(0)
	4546:
	TEST131B1:
		P0,	LOAD-ENUA(ZTARGET434),		;SETUP FOR IR=(000000)/BUTINSTRS TEST

			LOAD-ERROR(TEST131B1),		;ERROR DIRECTORY KEY
			DCS-CTR(C8,),			;COMPARE AT TARGET
		NEXT,	J/ARITH131B1			;
   (4546)  DCS[1,00,1,0,0,0]  BM[0111,,00,11,,11,00,,011,,100,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,101,000]

	4150:	;(FREE)
```

```
7450    ARITH131B1:
7451            P2=T,   D_DIVIDE=STEP,                  ;ALUs(A-PLUS-B), CINs(0)
7452                    D(C)_COUT15,                    ;GET CARRYOUT
7453                    BUS=A_ASPHI(C052525),           ;As(052525)
7454                    BUS=B_CSPD(C125252),            ;Bs(125252)
7455                    SR_DIVIDE=STEP,                 ;Ds(177777), COUT15=(0)
7456            NEXT,   J/COMP13181                     ;
    (4150)  DCS[0,00,0,0,0,0]  BM[1000,,10,00,,11,01,,111,,110...0,1,1,,0,,0,,,0,0110...0,,0000,0,,,11,000,,,001,101,0011
7457
7458    4151:  ;(FREE)
7459    COMP13181:
7460            P0,     BUMP=VERIFY,                    ;COUNT
7461            P2=T,   D_SR=XOR=CSPD(C177777),         ;COMPARE RECEIVED: EXPECTED
7462                    SAVE=D(C),                      ;SAVE CARRY
7463            NEXT,   J/GOBUT13181                    ;
    (4151)  DCS[0,00,0,0,0,1]  BM[0110,,10,00,,00,00,,000,,111...0,1,0,,0,,0,,,0,0101,,,0,,0000,0,,,11,000,,,001,101,0010]
7464
7465    4152:  ;(FREE)
7466    GOBUT13181:
7467            SETUP,  RETURN/TEST13182,               ;RETURN TO START OF NEXT SUBTEST
7468            NEXT,   CALL(DINTOIR=5)                 ;GO PUT D => IR, BUT(INSTRS)
    (4152)  DCS[0,00,0,0,0,0]  BM[0100,,00,11,,11,01,,000,,111...0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,111,0111
7469
7470
7471
7472    ;CHECK THAT CARRYOUT OF BIT15 (COUT15) WAS CORRECT
7473    4750:
7474    TEST13182:
7475            P0,     LOAD=ENUA(ZTARGET413),          ;BIT<02> CLEAR
7476                    LOAD=ERROR(TEST13182),          ;ERROR DIRECTORY KEY
7477                    DCS=CTR(C),);                   ;COMPARE AT TARGET
7478            NEXT,   J/GOBUT13182                    ;
    (4750)  DCS[1,00,1,0,0,0]  BM[1100,,00,11,,11,00,,001,,011...0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,101,0111
7479
7480    4153:  ;(FREE)
7481    GOBUT13182:
7482            SETUP,  RETURN/SCOPE1318,               ;RETURN TO SCOPE LOOP TEST WORD
7483            NEXT,   GOTO=PAGE(7),                   ;BUT TABLE
7484                    J/BUTD(C)C                      ;D(C)#= COUT15 # IN BIT<02>
    (4153)  DCS[0,00,0,0,0,0]  BM[0100,,00,00,,11,01,,100,,111...0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,001,0000]
7485
7486
7487
7488    4154:  ;(FREE)
7489    SCOPE1318:
7490            P3,     CSPD(05)_EMIT, EMIT/170360,     ;CONSTANT FOR USE BELOW
7491            NEXT,   BUTD(SCOPE),                    ;NO $RADR;  "TEST132A1" (+1, WORDS)
7492                    J/TEST132A1                     ;    ERROR:  "ARITH131A1"(+11, WORDS)
    (4154)  DCS[0,00,0,1,0,0]  BM[1111,,10,00,,00,11,,110,,000...0,0,0,,0,,0,,,0,1010,,,1,,0000,0,,,11,000,,,101,010,0011
7493
7494
7495
```



```
7496    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
7497
7498    ;CHECK INTERNAL ALU CARRIES  WITH:              (125252)+(125252)+(0)=(000000)
7499    ;ALSO CHECK ALU FUNCTION "A=MINUS-B", D(C)_COUT15=(1)
7500    ;
7501    4521:
7502    TEST132A1:
7503            P0,     LOAD=ENUA(ZTARGET434),          ;SETUP FOR IR=(0000001/BUTINSTRS TEST
7504                    LOAD=ERROR(TEST132A1),          ;ERROR DIRECTORY KEY
7505                    DCS=CTR(C7,),                   ;COMPARE AT TARGET
7506            NEXT,   J/ARITH132A1                    ;
    (4521)  DCS[1,00,1,0,0,0]  BM[1000,,00,11,,11,00,,011,,100...0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,101,010,0100]
7507
7508    4522:
7509    ARITH132A1:
7510            P3=T,   D_A=MINUS-B,                    ;ALUs(A=MINUS-B=MINUS-1), CINs(1)
7511                    D(C)_COUT15,                    ;GET CARRYOUT
7512                    BUS=A_ASPHI(C125252),           ;As(125252)
7513                    BUS=B_BSPHI(C125252),           ;Bs(125252)
7514                                                    ;Ds(000000), COUT15=(1)
7515            NEXT,   J/GOBUT132A1                    ;
    (4522)  DCS[0,00,0,0,0,0]  BM[1101,,01,11,,11,01,,110,,110...1,1,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,101,1011
7516
7517    4155:  ;(FREE)
7518    GOBUT132A1:
7519            P0,     BUMP=VERIFY,                    ;COUNT
7520            SETUP,  RETURN/TEST132A2,               ;RETURN TO START OF NEXT SUBTEST
7521            NEXT,   CALL(DINTOIR=5)                 ;GO PUT D => IR, BUT(INSTRS)
    (4155)  DCS[0,00,0,0,0,1]  BM[0100,,00,10,,10,01,,110,,111...0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,111,0111
7522
7523
7524
7525    ;CHECK THAT CARRYOUT OF BIT15 (COUT15) WAS CORRECT
7526    4516:
7527    TEST132A2:
7528            P0,     LOAD=ENUA(ZTARGET403),          ;BIT<01> SET
7529                    LOAD=ERROR(TEST132A2),          ;ERROR DIRECTORY KEY
7530                    DCS=CTR(C1,),                   ;COMPARE AT TARGET
7531            NEXT,   J/GOBUT132A2                    ;
    (4516)  DCS[1,00,1,0,0,0]  BM[1100,,00,11,,11,00,,000,,011...0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,101,1101]
7532
7533    4156:  ;(FREE)
7534    GOBUT132A2:
7535            SETUP,  RETURN/TEST132B1,               ;RETURN TO START OF NEXT SUBTEST
7536            NEXT,   GOTO=PAGE(7),                   ;BUT TABLE
7537                    J/BUTD(C)B                      ;D(C)#= COUT15 # IN BIT<01>

    (4156)  DCS[0,00,0,0,0,0]  BM[0100,,00,10,,10,01,,111,,111...0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,101,0000]
7538
7539
7540
7541    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
7542
```

```
	;CHECK INTERNAL ALU CARRIES WITH:		(052525)-(052525)-(0)=(000000)
	;ALSO CHECK ALU FUNCTION "DIVIDE-STEP" = "A-MINUS-B" SINCE D(C)=(1)
	;FROM ABOVE, D(C)_COUT15=(1)
	4517:
	TEST132B1:
		P0,	LOAD-ENUA(ZTARGET434),		;SETUP FOR IR=(000000)/BUTINSTRS TEST
			LOAD-ERROR(TEST132B1),		;ERROR DIRECTORY KEY
			DCS-CTR(C7,),			;COMPARE AT TARGET
		NEXT,	J/ARITH132B1			;
 (4517)	DCS[1,00,1,0,0,0]  BM[1000,,00,11,,11,00,,011,,100,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,101,111]

	4157:	;(FREE)
	ARITH132B1:
		P3-T,	D_DIVIDE-STEP,			;ALU=(A-MINUS-B-MINUS-1), CIN=(1)
			D[C]_COUT15,			;GET CARRYOUT
			BUS-A_ASPHI(C052525),		;A=(052525)
			BUS-B_BSPHI(C052525),		;B=(052525)
							;D=(000000), COUT15=(1)
		NEXT,	J/GOBUT132B1			;
 (4157)	DCS[0,00,0,0,0,0]  BM[1000,,01,11,,11,01,,111,,110,,,1,1,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,110,000]

	4160:	;(FREE)
	GOBUT132B1:
		P0,	BUMP-VERIFY,			;COUNT
		SETUP,	RETURN/TEST132B2,		;RETURN TO START OF NEXT SUBTEST
		NEXT,	CALL[DINTOIR-5]			;GO PUT D => IR, BUT(INSTR5)
 (4160)	DCS[0,00,0,0,0,1]  BM[0100,,00,10,,01,11,,010,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,111,011]



	;CHECK THAT CARRYOUT OF BIT15 (COUT15) WAS CORRECT
	4472:
	TEST132B2:
		P0,	LOAD-ENUA(ZTARGET417),		;BIT<02> SET
			LOAD-ERROR(TEST132B2),		;ERROR DIRECTORY KEY
			DCS-CTR(C3,),			;COMPARE AT TARGET
		NEXT,	J/GOBUT132B2			;
 (4472)	DCS[1,00,1,0,0,0]  BM[1100,,00,11,,11,00,,001,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,110,001]

	4161:	;(FREE)
	GOBUT132B2:
		SETUP,	RETURN/SCOPE132B,		;RETURN TO SCOPE LOOP TEST WORD
		NEXT,	GOTO-PAGE(7),			;BUT TABLE
			J/BUTD[C]C			;D(C)=COUT15 = IN BIT <02>
 (4161)	DCS[0,00,0,0,0,0]  BM[0100,,00,00,,11,10,,010,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,001,000]



	4162:	;(FREE)
	SCOPE132B:
		P3,	CSPD[06]_EMIT, EMIT/007417,	;CONSTANT FOR USE BELOW
		NEXT,	BUTD[SCOPE],			;NO ERROR:  "TEST133A1"   (+1, WORDS)
```



```
			J/TEST133A1			;   ERROR:  "ARITH132A1" (-9, WORDS)
 (4162)	DCS[0,00,0,1,0,0]  BM[0000,,10,11,,11,00,,001,,111,,,0,0,0,,0,,0,,,0,1001,,,1,,0000,0,,,11,000,,,101,010,011]




	; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

	;CHECK CARRY PROPOGATE/GENERATE LOGIC WITH:
	;(103607)-PLUS-(103607)-PLUS-(1)=(007417), COUT15=(1)
	4523:
	TEST133A1:
		P0,	LOAD-ENUA(ZTARGET434),		;SETUP FOR IR=(000000)/BUTINSTRS COMPARE
			LOAD-ERROR(TEST133A1),		;ERROR DIRECTORY KEY
			DCS-CTR(C10,),			;COMPARE AT TARGET
		NEXT,	J/OPB133A1			;
 (4523)	DCS[1,00,1,0,0,0]  BM[0101,,00,11,,11,00,,011,,100,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,101,001,000]

	4510:
	OPB133A1:
		P3,	CSPD[16]_EMIT,			;A,B-SIDE OPERANDS:
			EMIT/103607,			;"1000 0111 1000 0111"
		NEXT,	J/DOPA133A1			;
 (4510)	DCS[0,00,0,0,0,0]  BM[1000,,10,01,,11,10,,000,,111,,,0,0,0,,0,,0,,,0,0001,,,1,,0000,0,,,11,000,,,001,110,011]

	4163:	;(FREE)
	DOPA133A1:
		P2-T,	D_CSPD(D16), D(C)_ALU15,	;OP-A INTO D, D(C)_1
			SR_CSPD(D16),			;OP-A INTO SR TOO
		NEXT,	J/GOTEST133A1			;(OPA=OPB)
 (4163)	DCS[0,00,0,0,0,0]  BM[1010,,10,00,,00,00,,000,,100,,,0,1,1,,0,,0,,,0,0001,,,0,,0000,0,,,11,000,,,001,110,100]

	4164:	;(FREE)
	GOTEST133A1:
		P0,	BUMP-VERIFY,			;COUNT
		SETUP,	RETURN/TEST133A2,		;EXEC SUBR WHICH:
							;(1) D_OPA-PLUS-OPB-PLUS-1
							;(2) D_007417-XOR-D (EQUAL?)
		NEXT,	CALL[ALUCARRY1]			;(3) J/BUTINSTRS/(000000) (CHECK ANSWER)
 (4164)	DCS[0,00,0,0,0,1]  BM[0100,,00,10,,10,00,,000,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,000,000,111]



	;CHECK THAT CARRYOUT COUT15 ABOVE GENERATED CORRECTLY AS A (1)
	4500:
	TEST133A2:

		P0,	LOAD-ENUA(ZTARGET403),		;BIT<00>SET
			LOAD-ERROR(TEST133A2),		;ERROR DIRECTORY KEY
			DCS-CTR(C3,),			;COMPARE AT TARGET
			BUMP-VERIFY,			;COUNT
		NEXT,	J/GOBUT133A2			;
 (4500)	DCS[1,00,1,0,0,1]  BM[1100,,00,11,,11,00,,000,,011,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,110,101]
```

```
4165:  ;(FREE)
GOBUT133A2:
        SETUP,  RETURN/TEST133B1,                           ;RETURN TO START OF NEXT SUBTEST
        NEXT,   GOTO-PAGE(7),                               ;BUT TABLE
                J/BUTD[C]A                                  ;D[C]H = COUT15 IN BIT<00>
 (4165)  DCS[0,00,0,0,0,0]  BM[0100,,00,10,,10,00,,001,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,100,100]




;  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -

;CHECK CARRY PROPOGATE/GENERATE LOGIC WITH COMPLEMENT OF ABOVE:
;(074170)-PLUS-(074170)-PLUS-(0)=(170360), COUT15=(0)
4501:
TEST133B1:
        P0,     LOAD-ENUA(ZTARGET434),                      ;SETUP FOR IR=(000000)/BUTINSTRS COMPARE
                LOAD-ERROR(TEST133B1),                      ;ERROR DIRECTORY KEY
                DCS-CTR(C9,),                               ;COMPARE AT TARGET
        NEXT,   J/OPA133B1                                  ;
 (4501)  DCS[1,00,1,0,0,0]  BM[0110,,00,11,,11,00,,011,,100,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,110,110]

4166:  ;(FREE)
OPA133B1:
        P2-T,   D_NOT-CSPD(D16),                            ;
                D[C]_ALU15,                                 ;OP-A INTO D, D[C]_0
                SR_NOT-CSPD(D16),                           ;OP-A INTO SR TOO
                                                            ;A-SIDE OPERAND:
                                                            ;"0111 1000 0111 1000"
                                                            ;B-SIDE OPERAND WILL BE:
        NEXT,   J/GOTEST133B1                               ;(SAME AS OP-A)
 (4166)  DCS[0,00,0,0,0,0]  BM[0111,,10,00,,11,01,,101,,100,,,0,1,1,,0,,0,,,0,0001,,,0,,0000,0,,,11,000,,,001,110,111]

4167:  ;(FREE)
GOTEST133B1:
        SETUP,  RETURN/TEST133B2,                           ;EXEC SUBR WHICH:
                                                            ;(1) D_(NOT-OPA)-PLUS-(NOT-OPB)
                                                            ;(2) D_170360-XOR-D (EQUAL?)
        NEXT,   CALL[ALUCARRY2]                             ;(3) J/BUTINSTRS/(000000) (CHECK ANSWER)
 (4167)  DCS[0,00,0,0,0,0]  BM[0100,,00,10,,10,00,,010,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,000,010,000]



;CHECK THAT CARRYOUT COUT15 ABOVE GENERATED CORRECTLY AS A (0)
4502:
TEST133B2:
        P0,     LOAD-ENUA(ZTARGET402),                      ;BIT<00> CLEAR
                LOAD-ERROR(TEST133B2),                      ;ERROR DIRECTORY KEY
        NEXT,   J/GOBUT133B2                                ;
 (4502)  DCS[1,00,0,0,0,0]  BM[0000,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,111,000]
```



```
4170:  ;(FREE)
GOBUT133B2:
        SETUP,  RETURN/SCOPE133B,                           ;RETURN TO SCOPE LOOP TEST WORD
        NEXT,   GOTO-PAGE(7),                               ;BUT TABLE
                J/BUTD[C]A                                  ;D[C]H = COUT15 H IN BIT<00>
 (4170)  DCS[0,00,0,0,0,0]  BM[0100,,00,00,,11,11,,001,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,100,100]



4171:  ;(FREE)
SCOPE133B:
        P0,     BUMP-VERIFY,                                ;COUNT
        NEXT,   BUTD[SCOPE],                                ;NO ERROR:  "TEST134A1" (+1, WORDS)
                J/TEST134A1                                 ;   ERROR:  "OPA133A1"  (-11, WORDS)
 (4171)  DCS[0,00,0,1,0,1]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,101,001,001]



;  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -

;CHECK CARRY PROPOGATE/GENERATE LOGIC WITH:
;(045513)-PLUS-(141703)-PLUS-(1)=(007417), COUT15=(1)
4511:
TEST134A1:
        P0,     LOAD-ENUA(ZTARGET434),                      ;SETUP FOR IR=(000000)/BUTINSTRS COMPARE
                LOAD-ERROR(TEST134A1),                      ;ERROR DIRECTORY KEY
                DCS-CTR(C11,),                              ;COMPARE AT TARGET
        NEXT,   J/OPA134A1                                  ;
 (4511)  DCS[1,00,1,0,0,0]  BM[0100,,00,11,,11,00,,011,,100,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,101,001,010]

4512:
OPA134A1:
        P0,     BUMP-VERIFY,                                ;COUNT
        P3,     CSPD[17]_EMIT,                              ;A-SIDE OPERAND:
                EMIT/045513,                                ;"0100 1011 0100 1011"
        NEXT,   J/OPB134A1                                  ;
 (4512)  DCS[0,00,0,0,0,1]  BM[0100,,10,10,,11,01,,001,,011,,,0,0,0,,0,,0,,,0,0000,,,1,,0000,0,,,11,000,,,001,111,010]

4172:  ;(FREE)
OPB134A1:
        P3,     CSPD[16]_EMIT,                              ;B-SIDE OPERAND:
                EMIT/141703,                                ;"1100 0011 1100 0011"
        NEXT,   J/DOPA134A1                                 ;
 (4172)  DCS[0,00,0,0,0,0]  BM[1100,,10,00,,11,11,,000,,011,,,0,0,0,,0,,0,,,0,0001,,,1,,0000,0,,,11,000,,,001,111,011]

4173:  ;(FREE)

DOPA134A1:
        P2-T,   D_CSPD(D17), D[C]_ALU00,                    ;OP-A INTO D, D[C]_1
                SR_CSPD(D17),                               ;OP-A INTO SR TOO
        NEXT,   J/GOTEST134A1                               ;
 (4173)  DCS[0,00,0,0,0,0]  BM[1010,,10,00,,00,00,,000,,010,,,0,1,1,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,111,100]

4174:  ;(FREE)
```

```
	GOTEST134A1:
		P0,	BUMP-VERIFY,			;COUNT
		SETUP,	RETURN/TEST134A2,		;EXEC SUBR WHICH:
							;(1) D_OPA-PLUS-OPB-PLUS-1
							;(2) D_007417-XOR-D (EQUAL?)
		NEXT,	CALL[ALUCARRY1]			;(3) J/BUTINSTRS/(000000) (CHECK ANSWER)
    (4174)  DCS[0,00,0,0,0,1]  BM[0100,,00,10,,10,00,,011,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,000,000,111]



	;CHECK THAT CARRYOUT COUT15 ABOVE GENERATED CORRECTLY AS A (1)
	4503:
	TEST134A2:
		P0,	LOAD-ENUA(ZTARGET403),		;BIT<01> SET
			LOAD-ERROR(TEST134A2),		;ERROR DIRECTORY KEY
			DCS-CTR(C3,),			;COMPARE AT TARGET
			BUMP-VERIFY,			;COUNT
		NEXT,	J/GOBUT134A2			;
    (4503)  DCS[1,00,1,0,0,1]  BM[1100,,00,11,,11,00,,000,,011,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,111,101]

	4175:	;(FREE)
	GOBUT134A2:
		SETUP,	RETURN/TEST134B1,		;RETURN TO START OF NEXT SUBTEST
		NEXT,	GOTO-PAGE(7),			;BUT TABLE
			J/BUTD[C]B			;D[C] = COUT15 IN BIT<01>
    (4175)  DCS[0,00,0,0,0,0]  BM[0100,,00,10,,10,00,,100,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,101,000]





	; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

	;CHECK CARRY PROPOGATE/GENERATE LOGIC WITH COMPLEMENT OF ABOVE:
	;(132264)-PLUS-(036074)-PLUS-(0)=(170360), COUT15=(0)
	4504:
	TEST134B1:
		P0,	LOAD-ENUA(ZTARGET434),		;SETUP FOR IR=(000000)/BUTINSTRS COMPARE
			LOAD-ERROR(TEST134B1),		;ERROR DIRECTORY KEY
			DCS-CTR(C9,),			;COMPARE AT TARGET
		NEXT,	J/OPA134B1			;
    (4504)  DCS[1,00,1,0,0,0]  BM[0110,,00,11,,11,00,,011,,100,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,111,110]

	4176:	;(FREE)
	OPA134B1:
		P2-T,	D_NOT-CSPD(D17),		;
			D[C]_ALU00,			;OP-A INTO D, D[C]_(0)
			SR_NOT-CSPD(D17),		;OP-A INTO SR TOO
							;A-SIDE OPERAND:
							;"1011 0100 1011 0100"
							;B-SIDE OPERAND WILL BE:
		NEXT,	J/GOTEST134B1			;"0011 1100 0011 1100"
```



```
    (4176)  DCS[0,00,0,0,0,0]  BM[0111,,10,00,,11,01,,101,,010,,,0,1,1,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,111,111]

	4177:	;(FREE)
	GOTEST134B1:
		SETUP,	RETURN/TEST134B2,		;EXEC SUBR WHICH:
							;(1) D_(NOT-OPA)-PLUS-(NOT-OPB)
							;(2) D_170360-XOR-D (EQUAL?)
		NEXT,	CALL[ALUCARRY2]			;(3) J/BUTINSTRS/(000000) (CHECK ANSWER)
    (4177)  DCS[0,00,0,0,0,0]  BM[0100,,00,10,,10,00,,101,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,000,010,000]



	;CHECK THAT CARRYOUT COUT15 ABOVE GENERATED CORRECTLY AS A (0)
	4505:
	TEST134B2:
		P0,	LOAD-ENUA(ZTARGET401),		;BIT<01> CLEAR
			LOAD-ERROR(TEST134B2),		;ERROR DIRECTORY KEY
			DCS-CTR(C3,),			;COMPARE AT TARGET
		NEXT,	J/GOBUT134B2			;
    (4505)  DCS[1,00,1,0,0,0]  BM[1100,,00,11,,11,00,,000,,001,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,010,000,000]

	4200:	;(FREE)
	GOBUT134B2:
		SETUP,	RETURN/SCOPE134B,		;RETURN TO SCOPE LOOP TEST WORD
		NEXT,	GOTO-PAGE(7),			;BUT TABLE
			J/BUTD[C]B			;D[C]H = COUT15 H IN BIT<01>
    (4200)  DCS[0,00,0,0,0,0]  BM[0100,,00,01,,00,00,,001,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,101,000]



	4201:	;(FREE)
	SCOPE134B:
		P0,	BUMP-VERIFY,			;COUNT
		NEXT,	BUTD[SCOPE],			;NO ERROR:  "TEST135A1" (+1, WORDS)
			J/TEST135A1			;   ERROR:  "OPA134A1" (-11, WORDS)
    (4201)  DCS[0,00,0,1,0,1]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,101,001,011]



	; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

	;CHECK CARRY PROPOGATE/GENERATE LOGIC WITH:
	;(122645)-PLUS-(064551)-PLUS-(1)=(007417), COUT15=(1)
	4513:
	TEST135A1:
		P0,	LOAD-ENUA(ZTARGET434),		;SETUP FOR IR=(000000)/BUTINSTRS COMPARE

			LOAD-ERROR(TEST135A1),		;ERROR DIRECTORY KEY
			DCS-CTR(C11,),			;COMPARE AT TARGET
		NEXT,	J/OPA135A1			;
    (4513)  DCS[1,00,1,0,0,0]  BM[0100,,00,11,,11,00,,011,,100,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,101,001,100]

	4514:
```

```
OPA135A1:
        P0,     BUMP-VERIFY,                        !COUNT
        P3,     CSPD[17]_EMIT,                      !A-SIDE OPERAND:
                EMIT/122645,                        !"1010 0101 1010 0101"
        NEXT,   J/OPB135A1                          !
(4514)  DCS[0,00,0,0,0,1]  BM[1010,,10,01,,01,10,,100,,101,,,0,0,0,,0,,0,,,0,0000,,,1,,0000,0,,,11,000,,,010,000,010]

    4202: !(FREE)
OPB135A1:
        P3,     CSPD[16]_EMIT,                      !B-SIDE OPERAND:
                EMIT/064551,                        !"0110 1001 0110 1001"
        NEXT,   J/DOPA135A1                         !
(4202)  DCS[0,00,0,0,0,0]  BM[0110,,10,10,,01,01,,101,,001,,,0,0,0,,0,,0,,,0,0001,,,1,,0000,0,,,11,000,,,010,000,011]

    4203: !(FREE)
DOPA135A1:
        P2-T,   D_CSPD(D17), D[C]_ALU15,            !OP-A INTO D, D[C]_1
                SR_CSPD(D17),                       !OP-A INTO SR TOO
        NEXT,   J/GOTEST135A1                       !
(4203)  DCS[0,00,0,0,0,0]  BM[1010,,10,00,,00,00,,000,,100,,,0,1,1,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,010,000,100]

    4204: !(FREE)
GOTEST135A1:
        P0,     BUMP-VERIFY,                        !COUNT
        SETUP,  RETURN/TEST135A2,                   !EXEC SUBR WHICH:
                                                    !(1) D_OPA-PLUS-OPB-PLUS-1
                                                    !(2) D_007417-XOR-D (EQUAL?)
        NEXT,   CALL[ALUCARRY1]                     !(3) J/BUTINSTRS/(000000) (CHECK ANSWER)
(4204)  DCS[0,00,0,0,0,1]  BM[0100,,00,10,,10,00,,110,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,000,000,111]



!CHECK THAT CARRYOUT COUT15 ABOVE GENERATED CORRECTLY AS A (1)
    4506:
TEST135A2:
        P0,     LOAD-ENUA(ZTARGET417),              !BIT<02> SET
                LOAD-ERROR(TEST135A2),              !ERROR DIRECTORY KEY
                DCS-CTR(C3,),                       !COMPARE AT TARGET
                BUMP-VERIFY,                        !COUNT
        NEXT,   J/GOBUT135A2                        !
(4506)  DCS[1,00,1,0,0,1]  BM[1100,,00,11,,11,00,,001,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,010,000,101]

    4205: !(FREE)
GOBUT135A2:
        SETUP,  RETURN/TEST135B1,                   !RETURN TO START OF NEXT SUBTEST
        NEXT,   GOTO-PAGE(7),                       !BUT TABLE
                J/BUTD[C]C                          !D[C]H = COUT15 IN BIT<02>
(4205)  DCS[0,00,0,0,0,0]  BM[0100,,00,10,,10,00,,111,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,001,000]
```



```
! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

!CHECK CARRY PROPOGATE/GENERATE LOGIC WITH COMPLEMENT OF ABOVE:
!(055132)-PLUS-(113226)-PLUS-(0)=(170360), COUT15=(0)
    4507:
TEST135B1:
        P0,     LOAD-ENUA(ZTARGET434),              !SETUP FOR IR=(000000)/BUTINSTRS COMPARE
                LOAD-ERROR(TEST135B1),              !ERROR DIRECTORY KEY
                DCS-CTR(C9,),                       !COMPARE AT TARGET
        NEXT,   J/OPA135B1                          !
(4507)  DCS[1,00,1,0,0,0]  BM[0110,,00,11,,11,00,,011,,100,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,010,000,110]

    4206: !(FREE)
OPA135B1:
        P2-T,   D_NOT-CSPD(D17),                    !
                D[C]_ALU15,                         !OP-A INTO D, D[C]_(0)
                SR_NOT-CSPD(D17),                   !OP-A INTO SR TOO
                                                    !A-SIDE OPERAND:
                                                    !"0101 1010 0101 1010"
                                                    !B-SIDE OPERAND WILL BE:
        NEXT,   J/GOTEST135B1                       !"1001 0110 1001 0110"
(4206)  DCS[0,00,0,0,0,0]  BM[0111,,10,00,,11,01,,101,,100,,,0,1,1,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,010,000,111]

    4207: !(FREE)
GOTEST135B1:
        SETUP,  RETURN/TEST135B2,                   !EXEC SUBR WHICH:
                                                    !(1) D_(NOT-OPA)-PLUS-(NOT-OPB)
                                                    !(2) D_170360-XOR-D (EQUAL?)
        NEXT,   CALL[ALUCARRY2]                     !(3) J/BUTINSTRS/(000000) (CHECK ANSWER)
(4207)  DCS[0,00,0,0,0,0]  BM[0100,,00,10,,01,11,,111,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,000,010,000]



!CHECK THAT CARRYOUT COUT15 ABOVE GENERATED CORRECTLY AS A (0)
    4477:
TEST135B2:
        P0,     LOAD-ENUA(ZTARGET413),              !BIT<02> CLEAR
                LOAD-ERROR(TEST135B2),              !ERROR DIRECTORY KEY
                DCS-CTR(C3,),                       !COMPARE AT TARGET
        NEXT,   J/GOBUT135B2                        !
(4477)  DCS[1,00,1,0,0,0]  BM[1100,,00,11,,11,00,,001,,011,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,010,001,000]

    4210: !(FREE)
GOBUT135B2:
        SETUP,  RETURN/SCOPE135B,                   !RETURN TO SCOPE LOOP TEST WORD

        NEXT,   GOTO-PAGE(7),                       !BUT TABLE
                J/BUTO[C]C                          !D(C)H = COUT15 H IN BIT<02>
(4210)  DCS[0,00,0,0,0,0]  BM[0100,,00,01,,00,01,,001,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,001,000]
```

```
7918    4211:  !(FREE)
7919    SCOPE1358:
7920            P0,     BUMP-VERIFY,                    !COUNT
7921            NEXT,   BUTD[SCOPE],                    !NO ERROR:  "TEST136A1" (+1, WORD8)
7922                    J/TEST136A1                     !   ERROR:  "OPA136A1"  (-1, WORD8)
    (4211)  DCS[0,00,0,1,0,1]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,101,001,101]
7923
7924
7925
7926    ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
7927
7928    !CHECK CARRY PROPOGATE/GENERATE LOGIC WITH:
7929    !(055132)+PLUS+(132264)+PLUS+(1)=(007417), COUT15=(1)
7930    4515:
7931    TEST136A1:
7932            P0,     LOAD-ENUA(ZTARGET434),           !SETUP FOR IR=(000000)/BUTINSTR5 COMPARE
7933                    LOAD-ERROR(TEST136A1),          !ERROR DIRECTORY KEY
7934                    DC5-CTR(C10,),                  !COMPARE AT TARGET
7935            NEXT,   J/OPB136A1                      !
    (4515)  DCS[1,00,1,0,0,0]  BM[0101,,00,11,,11,00,,011,,100,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,101,100,010]
7936
7937    4542:
7938    OPA136A1:
7939            P3,     C5PD[16]_EMIT,                  !B-SIDE OPERAND:
7940                    EMIT/132264,                    !"1011 0100 1011 0100"
7941            NEXT,   J/DOPA136A1                     !
    (4542)  DCS[0,00,0,0,0,0]  BM[1011,,10,01,,00,10,,110,,100,,,0,0,0,,0,,0,,,0,0001,,,1,,0000,0,,,11,000,,,010,001,010]
7942
7943    4212:  !(FREE)
7944    DOPA136A1:
7945            P2=T,   D_NOT=C5PD(D17), D[C]_COUT07,    !DP=A INTO D, D[C]_1
7946                    SP_NOT=C5PD(D17),               !DP=A INTO SR TOO
7947            NEXT,   J/GOTEST136A1                   !"0101 1010 0101 1010"
    (4212)  DCS[0,00,0,0,0,0]  BM[0111,,10,00,,11,01,,101,,101,,,0,1,1,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,010,001,011]
7948
7949    4213:  !(FREE)
7950    GOTEST136A1:
7951            P0,     BUMP-VERIFY,                    !COUNT
7952            SETUP,  RETURN/TEST136A2,               !EXEC SUBR WHICH:
7953                                                    !(1) D_OPA-PLUS-OPB-PLUS-1
7954                                                    !(2) D_007417-XOR-D EQUAL?
7955            NEXT,   CALL[ALUCARRY1]                 !(3) J/BUTINSTR5/(000000) (CHECK ANSWER)
    (4213)  DCS[0,00,0,0,0,1]  BM[0100,,00,10,,01,11,,110,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,000,000,111]
7956
7957
7958
7959    !CHECK THAT CARRYOUT COUT15 ABOVE GENERATED CORRECTLY AS A (1)
7960    4476:
7961    TEST136A2:
7962            P0,     LOAD-ENUA(ZTARGET403),           !BIT<00> SET
7963                    LOAD-ERROR(TEST136A2),          !ERROR DIRECTORY KEY
7964                    DC5-CTR(C3,),                   !COMPARE AT TARGET
7965                    BUMP-VERIFY,                    !COUNT
```



```
7966            NEXT,   J/GOBUT136A2                    !
    (4476)  DCS[1,00,1,0,0,11  BM[1100,,00,11,,11,00,,000,,011,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,010,001,100]
7967
7968    4214:  !(FREE)
7969    GOBUT136A2:
7970            SETUP,  RETURN/TEST136B1,               !RETURN TO START OF NEXT SUBTEST
7971            NEXT,   GOTO-PAGE(7),                   !BUT TABLE
7972                    J/BUTO[C]A                      !D[C]N = COUT15 IN BIT<00>
    (4214)  DCS[0,00,0,0,0,0]  BM[0100,,00,10,,01,11,,101,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,100,100]
7973
7974
7975
7976
7977
7978    ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
7979
7980    !CHECK CARRY PROPOGATE/GENERATE LOGIC WITH COMPLEMENT OF ABOVE:
7981    !(125645)+PLUS+(132264)+PLUS+(0)=(170360), COUT15=(0)
7982    4475:
7983    TEST136B1:
7984            P0,     LOAD-ENUA(ZTARGET434),           !SETUP FOR IR=(000000)/BUTINSTR5 COMPARE
7985                    LOAD-ERROR(TEST136B1),          !ERROR DIRECTORY KEY
7986                    DC5-C7N(C9,),                   !COMPARE AT TARGET
7987            NEXT,   J/OPA136B1                      !
    (4475)  DCS[1,00,1,0,0,0]  BM[0110,,00,11,,11,00,,011,,100,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,010,001,101]
7988
7989    4215:  !(FREE)
7990    OPA136B1:
7991            P2=T,   D_C5PD(D17),                    !
7992                    D[C]_0,                         !DP=A INTO D, D[C]_0
7993                    SR_C5PD(D17),                   !DP=A INTO SR TOO
7994                                                    !A-SIDE OPERAND:
7995                                                    !"1010 0101 1010 0101"
7996                                                    !B-SIDE OPERAND WILL BE:
7997            NEXT,   J/GOTEST136B1                   !"0100 1011 0100 1011"
    (4215)  DCS[0,00,0,0,0,0]  BM[1010,,10,00,,00,00,,000,,000,,,0,1,1,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,010,001,110]
7998
7999    4216:  !(FREE)
8000    GOTEST136B1:
8001            SETUP,  RETURN/TEST136B2,               !EXEC SUBR WHICH:
8002                                                    !(1) D_(NOT-OPA)+PLUS+(NOT-OPB)
8003                                                    !(2) D_170360-XOR-D (EQUAL?)
8004            NEXT,   CALL[ALUCARRY2]                 !(3) J/BUTINSTR5/(000000) (CHECK ANSWER)
    (4216)  DCS[0,00,0,0,0,0]  BM[0100,,00,10,,01,11,,011,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,000,010,000]
8005
8006
8007
8008    !CHECK THAT CARRYOUT COUT15 ABOVE GENERATED CORRECTLY AS A (0)
8009    4473:
8010    TEST136B2:
8011            P0,     LOAD-ENUA(ZTARGET402),           !BIT<00> CLEAR
8012                    LOAD-ERROR(TEST136B2),          !ERROR DIRECTORY KEY
```

```
                  DCS-CTR(C3.),                              ;COMPARE AT TARGET
          NEXT,   J/GOBUT136B2                               ;
   (4473)  DCS[1,00,1,0,0,0]  BM[1100,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,010,001,111]

    4217:  ;(FREE)
    GOBUT136B2:
          SETUP,  RETURN/SCOPE1368,                          ;RETURN TO SCOPE LOOP TEST WORD
          NEXT,   GOTO-PAGE(7);                              ;BUT TABLE
                  J/BUTD[C]A                                 ;D[C]N @ COUT15 H IN BIT<00>
   (4217)  DCS[0,00,0,0,0,0]  BM[0100,,00,01,,00,10,,000,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,100,100]


    4220:  ;(FREE)
    SCOPE1368:
          P0,     BUMP-VERIFY,                               ;COUNT
          NEXT,   BUTD[SCOPE],                               ;NO ERROR:  "TEST320A"  (+1, WORDS)
                  J/TEST320A                                 ;   ERROR:  "OPA1368A1" (-11, WORDS)
   (4220)  DCS[0,00,0,1,0,1]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,101,100,011]


    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

    ;      THE FOLLOWING TWO SUBROUTINES ARE USED IN THE ABOVE CARRY LOOKAHEAD/INTERNAL CARRIES TESTS:

    7007:  ;(FREE)
    ALUCARRY1:
          P3-T,   D_A-PLUS-B-PLUS-D[C],                      ;D <- SUM OF A, B;  D[C] WAS SET PREVIOUSLY
                  D[C]_COUT15,                               ;GET CARRYOUT FOR EXAMINATION LATER
                  SR_A-PLUS-B-PLUS-D(C),                     ;ALSO GET IT INTO THE SR
                  BUS-A_SR,                                  ;A-SIDE OPERAND WAS IN THE SR
                  BUS-B_CSPD(D16),                           ;B-SIDE OPERAND WAS IN CSP(16)
          NEXT,   J/ALUCARRY1A                               ;
   (7007)  DCS[0,00,0,0,0,0]  BM[0100,,10,00,,00,00,,000,,110,,,1,1,1,,0,,0,,,0,0001,,,0,,0000,0,,,11,000,,,000,001,111]

    7017:  ;(FREE)
    ALUCARRY1A:
          P2-T,   D_SR-XOR-CSPD(D06), SAVE-D[C],             ;COMPARE RECEIVED : (007417)
          NEXT,   J/DINTOIRS                                 ;GO PUT D -> IR, BUT(INSTRS) FOR (000000)
   (7017)  DCS[0,00,0,0,0,0]  BM[0110,,10,00,,00,00,,000,,111,,,0,1,0,,0,,0,,,0,1001,,,0,,0000,0,,,11,000,,,010,111,011]


    7020:  ;(FREE)
    ALUCARRY2:
          P3-T,   D_A-PLUS-NOT-B-PLUS-D[C],                  ;D <- DIFF OF A, B;  D[C] WAS SET PREVIOUSLY
                  D[C]_COUT15,                               ;GET CARRYOUT FOR EXAMINATION LATER
```



```
                  SR_A-PLUS-NOT-B-PLUS-D(C),                 ;ALSO GET IT INTO THE SR
                  BUS-A_SR,                                  ;A-SIDE OPERAND WAS IN THE SR
                  BUS-B_CSPD(D16),                           ;B-SIDE OPERAND WAS IN CSP(16)
          NEXT,   J/ALUCARRY2A                               ;
   (7020)  DCS[0,00,0,0,0,0]  BM[0101,,10,00,,00,00,,000,,110,,,1,1,1,,0,,0,,,0,0001,,,0,,0000,0,,,11,000,,,000,010,001]

    7021:  ;(FREE)
    ALUCARRY2A:
          P2-T,   D_SR-XOR-CSPD(D05), SAVE-D(C),             ;COMPARE RECEIVED : (170360)
          NEXT,   J/DINTOIRS                                 ;GO PUT D -> IR, BUT(INSTRS) FOR (000000)
   (7021)  DCS[0,00,0,0,0,0]  BM[0110,,10,00,,00,00,,000,,111,,,0,1,0,,0,,0,,,0,1010,,,0,,0000,0,,,11,000,,,010,111,011]


    ;.PAGE==========================================================================================

    .TOC    *    TEST320:  D[C] SELECTION / COUT07-DOUT07 / D<14:00>=ZERO@BIT<00>

    ;*****************************************************************************
    ;*
    ;*      TESTS:  320 A - F                                    UWORDS:  026 + 014
    ;*
    ;*      FUNCTIONS:
    ;*
    ;*      THESE SIX TESTS CHECK THE D(C) ADDRESS SELECTION LOGIC, COUT07 CARRY
    ;*      AND DOUT07 BUTS, AND THE D<14:00>=ZERO BUT, WHEN ONLY BIT<00>=1 (NOT
    ;*      CHECKED IN TEST410).
    ;*
    ;*****************************************************************************


    ;TEST-320-A SETS ONLY THE D[C] INPUT "ALU00" (CODE=010), AND THEN CHECKS THAT D[C]_ALU00
    ;  RESULTS IN A "1".
    4543:
    TEST320A:
          P0,     LOAD-ENUA(ZTARGET403),                     ;BIT<00> SET
                  LOAD-ERROR(TEST320A),                      ;ERROR DIRECTORY KEY
                  DCS-CTR(C6.),                              ;COMPARE AT TARGET
          NEXT,   J/SETONE320A                               ;
   (4543)  DCS[1,00,1,0,0,0]  BM[1001,,00,11,,11,00,,000,,011,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,111,000,000]

    4700:
    SETONE320A:
          P0,     BUMP-VERIFY,                               ;COUNT
          P3,     CSPD[17]_EMIT, EMIT/000001,                ;A ONE
          NEXT,   J/SETD320A                                 ;
   (4700)  DCS[0,00,0,0,0,1]  BM[0000,,10,00,,00,00,,000,,001,,,0,0,0,,0,,0,,,0,0000,,,1,,0000,0,,,11,000,,,010,010,001]

    4221:  ;(FREE)
```

```
8111    SETD320A:
8112            P2=T,   D_CSPD(D17), D[C]_0,                    ;D GETS (0000001)
8113                    SR_CSPD(D17),                           ;SO DOES SR
8114            P3,     A#BSPRI[11]_D,                          ;STORE THE CONSTANT
8115            NEXT,   J/GETDC320A                             ;
     (4221)  DCS[0,00,0,0,0,0]  BM[1010,,10,11,,00,01,,000,,000,,,0,1,1,,0,,0,,,0,0000,,,0,,1011,0,,,11,000,,,010,010,010]
8116
8117    4222:   ;(FREE)
8118    GETDC320A:
8119            P2=T,   D_A-PLUS-B-PLUS-0, D[C]_ALU00,          ;D[C] CODE (010)
8120                    BUS=A_SR,                               ;A=(000001)
8121                    BUS=B_CSPD(C000000),                    ;B=(000000)
8122            NEXT,   J/GOBUT320A                             ;D=(0000001), ONLY D[C]=ALU00 SET
     (4222)  DCS[0,00,0,0,0,0]  BM[1001,,10,00,,00,00,,000,,010,,,0,1,0,,0,,0,,,0,0100,,,0,,0000,0,,,11,000,,,010,010,011]
8123
8124    4223:   ;(FREE)
8125    GOBUT320A:
8126            P0,     BUMP-VERIFY,                            ;COUNT
8127            SETUP,  RETURN/TEST320B,                        ;RETURN TO START OF NEXT SUBTEST
8128            NEXT,   GOTO-PAGE(7),                           ;BUT TABLE
8129                    J/BUTD[C]A                              ;D[C]H IN BIT<00>
     (4223)  DCS[0,00,0,0,0,1]  BM[0100,,00,10,,11,01,,010,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,100,100]
8130
8131
8132
8133
8134    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
8135
8136    ;TEST-320-B CHECKS THAT D<14:00>=ZERO IS NOT SET WHEN D=(0000001)
8137    4552:
8138    TEST320B:
8139            P0,     LOAD-ENUA(ZTARGET400),                  ;BIT<01> CLEAR
8140                    LOAD-ERROR(TEST320B),                   ;ERROR DIRECTORY KEY
8141                    DCS-CTR(C3,),                           ;COMPARE AT TARGET
8142                    BUMP-VERIFY,                            ;COUNT
8143            NEXT,   J/GOBUT320B                             ;
     (4552)  DCS[1,00,1,0,0,1]  BM[1100,,00,11,,11,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,010,010,100]
8144
8145    4224:   ;(FREE)
8146    GOBUT320B:
8147            SETUP,  RETURN/TEST320C,                        ;RETURN TO START OF NEXT SUBTEST
8148            NEXT,   GOTO-PAGE(7),                           ;BUT TABLE
8149                    J/BUTD-IS-ZERO                          ;CHEK BM IT
     (4224)  DCS[0,00,0,0,0,0]  BM[0100,,00,10,,11,10,,010,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,100,001]
8150
8151
8152
8153    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
8154
8155    ;TEST-320-C CHECKS THAT THE COUTIS/COUTOT SIGNALS DON'T TRACK EACH OTHER
8156    4562:
8157    TEST320C:
```



```
8158            P0,     LOAD-ENUA(ZTARGET413),                  ;BIT<02> CLEAR
8159                    LOAD-ERROR(TEST320C),                   ;ERROR DIRECTORY KEY
8160                    DCS-CTR(C4,),                           ;COMPARE AT TARGET
8161            NEXT,   J/SETONE320C                            ;
     (4562)  DCS[1,00,1,0,0,0]  BM[1001,,00,11,,11,00,,001,,011,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,010,010,101]
8162
8163    4225:   ;(FREE)
8164    SETONE320C:
8165            P3,     CSPD[16]_ENIT, ENIT/100000,             ;A ONE IN B15
8166            NEXT,   J/SETD320C                              ;
     (4225)  DCS[0,00,0,0,0,0]  BM[1000,,10,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0001,,,1,,0000,0,,,11,000,,,010,010,110]
8167
8168    4226:   ;(FREE)
8169    SETD320C:
8170            P2=T,   D_CSPD(D16), D[C]_ALU15,                ;D GETS (100000)
8171                    SR_CSPD(D16),                           ;SO DOES SR
8172            P3,     A#BSPHI[13]_D,                          ;STORE THE CONSTANT
8173            NEXT,   J/GETDC320C                             ;
     (4226)  DCS[0,00,0,0,0,0]  BM[1010,,10,11,,00,01,,001,,100,,,0,1,1,,0,,0,,,0,0001,,,0,,1011,0,,,11,000,,,010,010,111]
8174
8175    4227:   ;(FREE)
8176    GETDC320C:
8177            P0,     BUMP-VERIFY,                            ;COUNT
8178            P2=T,   D_A=PLUS-B-PLUS-0, D[C]_COUT07,         ;D[C] GETS COUT07=0, COUTIS=1
8179                    BUS=A_SR,                               ;A=(100000)
8180                    BUS=B_CSPD(C125252),                    ;B=(125252)
8181            NEXT,   J/GOBUT320C                             ;D=(025252)
     (4227)  DCS[0,00,0,0,0,1]  BM[1001,,10,00,,00,00,,000,,101,,,0,1,0,,0,,0,,,0,0110,,,0,,0000,0,,,11,000,,,010,011,000]
8182
8183    4230:   ;(FREE)
8184    GOBUT320C:
8185            SETUP,  RETURN/TEST320D,                        ;RETURN TO START OF NEXT SUBTEST
8186            NEXT,   GOTO-PAGE(7),                           ;BUT TABLE
8187                    J/BUTD[C]C                              ;D[C]H IN BIT<02>
     (4230)  DCS[0,00,0,0,0,0]  BM[0100,,00,10,,10,11,,111,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,001,000]
8188
8189
8190
8191
8192
8193    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
8194
8195    ;TEST-320-D CHECKS THAT THE BUTR(COUT07#DOUT07) SEES THE (01#) THAT WAS GENERATED
8196    4537:
8197    TEST320D:
8198            P0,     LOAD-ENUA(ZTARGET403),                  ;BIT<2:1> = "01"

8199                    LOAD-ERROR(TEST320D),                   ;ERROR DIRECTORY KEY
8200                    DCS-CTR(C3,),                           ;COMPARE AT TARGET
8201            NEXT,   J/GOBUT320D                             ;
     (4537)  DCS[1,00,1,0,0,0]  BM[1100,,00,11,,11,00,,000,,011,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,010,011,001]
8202
8203    4231:   ;(FREE)
8204    GOBUT320D:
```

```
        SETUP,  RETURN/TEST320E,                        ;RETURN TO START OF NEXT SUBTEST
        NEXT,   GOTO-PAGE(7),                           ;BUT TABLE
                J/BUTCOUT7DOUT7                         ;COUT07#D07 IN BIT<2:1>
(4231)  DCS[0,00,0,0,0,0]  BM[0100,,00,10,,11,01,,100,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,001,001]




;  - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

;TEST-320-E SETS ONLY THE D[C] INPUT "COUT07" (CODE=101), AND THEN CHECKS THAT D[C]_COUT07
;  RESULTS IN A "1".
45541
TEST320E:
        P0,     LOAD-ENUA(ZTARGET403),                  ;BIT<01> SET
                LOAD-ERROR(TEST320E),                   ;ERROR DIRECTORY KEY
                DCS-CTR(C4,),                           ;COMPARE AT TARGET
        NEXT,   J/SETONE320E                            ;
(4554)  DCS[1,00,1,0,0,0]  BM[1001,,00,11,,11,00,,000,,011,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,010,011,010]

4232: ;(FREE)
SETONE320E:
        P0,     BUMP-VERIFY,                            ;COUNT
        P3,     CSPD[15]_EMIT, EMIT/000200,             ;A ONE IN BIT07
        NEXT,   J/SETD320E                              ;
(4232)  DCS[0,00,0,0,0,1]  BM[0000,,10,00,,00,10,,000,,000,,,0,0,0,,0,,0,,,0,0010,,,1,,0000,0,,,11,000,,,010,011,011]

4233: ;(FREE)
SETD320E:
        P2-T,   D_CSPD(D15), D[C]_0,                    ;D GETS (000200)
                SR_CSPD(D15),                           ;SO DOES SR
        P3,     A#BSPHI[15]_D,                          ;STORE THE CONSTANT
        NEXT,   J/GETDC320E                             ;
(4233)  DCS[0,00,0,0,0,0]  BM[1010,,10,11,,00,01,,010,,000,,,0,1,1,,0,,0,,,0,0010,,,0,,1011,0,,,11,000,,,010,011,100]

4234: ;(FREE)
GETDC320E:
        P2-T,   D_A-PLUS-B-PLUS-0, D[C]_COUT07,         ;D[C] CODE (101)
                BUS-A_SR,                               ;A=(000200)
                BUS-B_CSPD(D15),                        ;B=(000200)
        NEXT,   J/GOBUT320E                             ;D=(000400), ONLY D[C]=COUT07 SET
(4234)  DCS[0,00,0,0,0,0]  BM[1001,,10,00,,00,00,,000,,101,,,0,1,0,,0,,0,,,0,0010,,,0,,0000,0,,,11,000,,,010,011,101]

4235: ;(FREE)
GOBUT320E:
        SETUP,  RETURN/TEST320F,                        ;RETURN TO START OF NEXT SUBTEST
        NEXT,   GOTO-PAGE(7),                           ;BUT TABLE
                J/BUTD[C]B                              ;D[C]B IN BIT<01>
(4235)  DCS[0,00,0,0,0,0]  BM[0100,,00,10,,11,01,,110,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,101,000]
```



```

;  - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

;TEST-320-F CHECKS THAT THE BUTR(COUT07#DOUT07) SEES THE (10*) THAT WAS GENERATED
45561
TEST320F:
        P0,     LOAD-ENUA(ZTARGET405),                  ;BIT<2:1> = "10"
                LOAD-ERROR(TEST320F),                   ;ERROR DIRECTORY KEY
                DCS-CTR(C3,),                           ;COMPARE AT TARGET
        NEXT,   J/GOBUT320F                             ;
(4556)  DCS[1,00,1,0,0,0]  BM[1100,,00,11,,11,00,,000,,101,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,010,011,110]

4236: ;(FREE)
GOBUT320F:
        SETUP,  RETURN/SCOPE320,                        ;RETURN TO SCOPE LOOP TEST WORD
        NEXT,   GOTO-PAGE(7),                           ;BUT TABLE
                J/BUTCOUT7DOUT7                         ;COUT07#D07 IN BIT<2:1>
(4236)  DCS[0,00,0,0,0,0]  BM[0100,,00,01,,00,11,,111,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,001,001]


4237: ;(FREE)
SCOPE320:
        P0,     BUMP-VERIFY,                            ;COUNT
                BUSDIN_INIT=[I],                        ;RESET PROC UCONS
                EN-CLK-IR(15-00),                       ;
        NEXT,   BUTD[SCOPE],                            ;NO ERROR:  "TEST350" (+1, WORDS)
                J/TEST350                               ;   ERROR:  "SETONE320A"  (-20, WORDS)
(4237)  DCS[0,00,0,1,0,1]  BN[0000,,00,00,,00,01,,000,,100,,,0,0,0,,0,,0,,,1,1001,,,0,,0000,0,,,11,000,,,111,000,001]







;.PAGE=================================================================================================

.TOC    *   TEST350-352:  ASP/BSP HI/LO ADDRESSING MODES, DATA VALIDITY

;*************************************************************************************
;*
;*      TESTS:  350 - 352                               UWORDS:  075 + 125
;*
;*      FUNCTIONS:
;*

;*      VERIFIES THE ASP/BSP ADDRESSING MODES,
;*
;*************************************************************************************



;TEST 350 A-D USES THE A/B SP "SF" AND "DF" ADDRESS MODES TO WRITE A UNIQUE PATTERN TO
```

```
;LOCATIONS (00:07) OF ASPHI, ASPLO, BSPHI, BSPLO, THRU USE OF A COUNT LOOP.
;THE GENERATED PATTERN IS THEN READ BACK, AND COMPARED TO A REGENERATED ORIGINAL,
;TO CHECK FOR ADDRESSING CORRECTNESS, ABILITY TO READ/WRITE, AND DATAPATH VALIDITY.
;
;FIRST FILL UP THE SCRATCHPADS:
;
;       AT THE END OF THE FILLUP LOOP, A/B SP HI/LO LOOK LIKE THIS:
;
;       LOCTN   ASPHI           BSPHI           ASPLO           BSPLO
;       -----   ------          ------          ------          ------
;
;        00     050700          050700          057077          057077
;        01     051611          051611          056166          056166
;        02     052522          052522          055255          055255
;        03     053433          053433          054344          054344
;        04     054344          054344          053433          053433
;        05     055255          055255          052522          052522
;        06     056166          056166          051611          051611
;        07     057077          057077          050700          050700
;        10     ------          ------          ------          ------
;        11     000001          000001          ------          ------
;        12     000152          000152          ------          ------
;        13     100000          100000          000125          000125
;        14     ------          ------          ------          ------
;        15     000200          000200          ------          ------
;        16     ------          ------          ------          ------
;        17     ------          ------          ------          ------
;
;       NOTE:  CONSTANTS FOR DCS IN ASPHI/BSPHI 01/03/05/07 WERE DESTROYED,
;              AND *MUST* BE RESTORED AFTER LEAVING THIS TEST SEQUENCE.
;
;
; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -


4701:
TEST350:
        P0,     LOAD-ERROR(TEST350),                    ;ERROR DIRECTORY KEY
        P3,     CSPD[15]_EMIT, EMIT/177067,             ;ADDED TO PATTERN "N" TO GET "N+1"
        NEXT,   J/START350                              ;
(4701)  DCS[1,00,0,0,0,0]  BM[1111,,10,11,,10,00,,110,,111,,,0,0,0,,0,,0,,,0,0010,,,1,,0000,0,,,11,000,,,010,100,000]

4240:   ;(FREE)
START350:
        P0,     BUMP-VERIFY,                            ;COUNT
        P3,     CSPD[14]_EMIT, EMIT/057077,             ;THE INITIAL PATTERN
                                                        ;SP=0, DM=7, DF=7
        NEXT,   J/LOADSRD350                            ;
(4240)  DCS[0,00,0,0,0,1]  BM[0101,,10,11,,10,00,,111,,111,,,0,0,0,,0,,0,,,0,0011,,,1,,0000,0,,,11,000,,,010,100,001]

4241:   ;(FREE)
LOADSRD350:
        P2-T,   D_CSPD(D14),                            ;INITIAL PATTERN
                SR_CSPD(D14),                           ;D->IR, SR=TEMP
```



```
        NEXT,   GOTO-PAGE(7),                           ;
                J/DINTOIR350                            ;
(4241)  DCS[0,00,0,0,0,0]  BM[1010,,10,00,,00,00,,000,,111,,,0,1,1,,0,,0,,,0,0011,,,0,,0000,0,,,11,100,,,011,101,011]

;*** LOOP BACK POINT FOR SP FILLUP ***

7353:
DINTOIR350:
        SETUP,  RETURN/WRITEDF350,                      ;CALL SUBR FOR D => IR
        NEXT,   CALL(DINTOIR)                           ;
(7353)  DCS[0,00,0,0,0,0]  BM[0100,,00,00,,00,00,,010,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,110,111]

4002:   ;(FREE)
WRITEDF350:
        P0,     DCS-CTR(C15,),                          ;HOLD UP
        P3,     A&BSPHI[DF]_D,                          ;USE DF TO WRITE A/B HI, W/ WR(AB,H,A)
        NEXT,   GOTO-PAGE(7),                           ;
                J/WRITESF350                            ;
(4002)  DCS[0,00,1,0,0,0]  BM[0000,,00,00,,00,10,,000,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,1011,0,,,11,100,,,000,001,110]

7016:   ;(FREE)
WRITESF350:
        P3,     A&BSPLO[SF]_D,                          ;USE SF TO WRITE A/B LO, W/ WR(AB,L,B)
        NEXT,   J/NEXTPAT350                            ;
(7016)  DCS[0,00,0,0,0,0]  BM[0000,,00,01,,00,11,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0011,0,,,11,000,,,000,010,011]

7023:   ;(FREE)
NEXTPAT350:
        P2-T,   SR&D_SR-PLUS-CSPD(D15), SAVE-D(C),      ;INCREMENT PATTERN
        NEXT,   BUTR(DMO),                              ;IF TRUE: "LOADHI350" (+1, WORDS) EXIT LOOP
                J/DINTOIR350                            ;IF FALSE: "DINTOIR350" (-3, WORDS) KEEP GOING
(7023)  DCS[0,00,0,0,0,0]  BM[1001,,10,00,,00,00,,000,,111,,,0,1,1,,0,,0,,,0,0010,,,0,,0000,0,,,01,001,,,011,101,011]


;COME HERE IF DONE LOOPING
7357:
LOADHI350:
                                                        ;*** IF ANY ASP/BSP HI/LO BIT<03> IN ADDR STUCK ZERO,
                                                        ;       ONE OF THE FOLLOWING TWO WORDS WILL OVERWRITE
                                                        ;       A PREVIOUSLY WRITTEN LOCATION (IE, 02 OR 03) ***
        P2-T,   D_CSPD(D01), SAVE-D(C),                 ;PATTERN OF (000152)
        P3,     A&BSPHI[12]_D,                          ;WRITE SPHI WITH ADDRS BIT<03> SET, DF=0, W/ WR(AB,H,A)
        NEXT,   J/LOADHIA350                            ;  FOR USE IN TEST373
(7357)  DCS[0,00,0,0,0,0]  BM[1010,,10,10,,00,00,,001,,111,,,0,1,0,,0,,0,,,0,1110,,,0,,1011,0,,,11,000,,,000,010,100]

7024:   ;(FREE)

LOADHIA350:
        P2-T,   D_CSPD(D02), SAVE-D(C),                 ;PATTERN OF (000125)
        P3,     A&BSPLO[13]_D,                          ;WRITE SPHI WITH ADDRS BIT<03> SET, DF=0, W/ WR(AB,L,B)
        NEXT,   J/AGAINSRD350                           ;
(7024)  DCS[0,00,0,0,0,0]  BM[1010,,10,11,,00,01,,001,,111,,,0,1,0,,0,,0,,,0,1101,,,0,,0011,0,,,11,000,,,000,010,101]
```

```
8398    7025:  ;(FREE)
8399    AGAIN&RD350:
8400            P2=T,   D_C&PD(D14),                    ;AGAIN RESET D, SR TO INIVIAL PATTERN
8401                    SR_C&PD(D14),                   ;OF (007077)
8402            NEXT,   GOTO-PAGE(4),                   ;
8403                    J/DINTOIRA350                   ;
    (7025)  DC&[0,00,0,0,0,0]  BM[1010,,10,00,,00,00,,000,,100...0,1,1,,0,,0,,,0,0011...0,,0000,0,,,11,100,,,101,110,011]
8404
8405    ;*** LOOP BACK ENTRY POINT FOR TESTS ***
8406
8407    4563:
8408    DINTOIRA350:
8409            SETUP,  RETURN/TEST350A,                ;GO TO SUBR WHICH PUTS SR => IR FOR SF/DF/DM
8410            NEXT,   CALL(&RINTOIR]                  ;
    (4563)  DC&[0,00,0,0,0,0]  BM[0100,,00,11,,11,01,,010,,111...0,0,0,,0,,0,,,0,0000...0,,0000,0,,,11,100,,,010,110,110]
8411
8412
8413    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
8414
8415    ;TEST 350A CHECKS B&P-ADDR&/SF, B&PLC ADDR& FOR ERR0R&
8416    4752:
8417    TE&T350A:
8418            P0,     LOAD-ENUA(%TARGET434),          ;INSTR& FOR IR=(000000)
8419                    LOAD-ERROR(TE&T350A),           ;ERROR DIRECTORY KEY
8420                    DC&-CTR(C7,),                   ;COMPARE AT TARGET
8421            NEXT,   J/COMP350A                      ;
    (4752)  DC&[1,00,1,0,0,0]  BM[1000,,00,11,,11,00,,011,,100...0,0,0,,0,,0,,,0,0000...0,,0000,0,,,11,000,,,111,000,010]
8422
8423    4702:
8424    COMP350A:
8425            P2=T,   D_SR-XOR-B&PLO[SF], SAVE-D[C],  ;COMPARE EXPECTED;RECEIVED, BITWISE
8426            NEXT,   J/GOBUT350A                     ;
    (4702)  DC&[0,00,0,0,0,0]  BM[0110,,00,01,,00,00,,000,,111...0,1,0,,0,,0,,,0,0000...0,,0000,0,,,11,000,,,010,100,011]
8427
8428    4243:  ;(FREE)
8429    GOBUT350A:
8430            P0,     BUMP-VERIFY,                    ;COUNT
8431            &ETUP,  RETURN/RE&ETIR350A,             ;GO TO &UBR WHICH:
8432            NEXT,   CALL[DINTOIP=5]                 ;  D => IR, BUT(IN&TR&)
    (4243)  DC&[0,00,0,0,0,1]  BM[0111,,00,00,,00,10,,010,,111...0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,111,011]
8433
8434    7022:  ;(FREE)
8435    RE&ETIR350A:
8436            &ETUP,  RETURN/TE&T350B,                ;GO TO &UBR WHICH PUTS &R => IR FOR &F/DF/DM
8437            NEXT,   CALL-[&RINTOIR]                 ;
    (7022)  DC&[0,00,0,0,0,0]  BM[0100,,00,11,,11,01,,011,,111...0,0,0,,0,,0,,,0,0000...0,,0000,0,,,11,100,,,010,110,110]
8438
8439
8440    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
8441
8442    ;TEST 350B CHECK& &&P-ADDR&/DF, B&PHI ADDR& FOR ERR0R&
```



```
8443    4753:
8444    TE&T350B:
8445            P0,     LOAD-ENUA(%TARGET434),          ;INSTR& FOR IR=(000000)
8446                    LOAD-ERROR(TE&T350B),           ;ERROR DIRECTORY KEY
8447                    DC&-CTR(C7,),                   ;COMPARE AT TARGET
8448            NEXT,   J/COMP350B                      ;
    (4753)  DC&[1,00,1,0,0,0]  BM[1000,,00,11,,11,00,,011,,100...0,0,0,,0,,0,,,0,0000...0,,0000,0,,,11,000,,,010,100,100]
8449
8450    4244:  ;(FREE)
8451    COMP350B:
8452            P2=T,   D_&P-XOR-B&PHI[DF], &AVE-D[C],  ;COMPARE EXPECTED;RECEIVED, BITWI&E
8453            NEXT,   J/GOBUT350B                     ;
    (4244)  DC&[0,00,0,0,0,0]  BM[0110,,01,00,,00,00,,000,,111...0,1,0,,0,,0,,,0,0000...0,,0000,0,,,11,000,,,010,100,101]
8454
8455    4745:  ;(FREE)
8456    GOBUT350B:
8457            &ETUP,  RETURN/RE&ETIR350B,             ;GO TO &UBR WHICH:
8458            NEXT,   CALL[DINTOIR=5]                 ;  D => IR, BUT(IN&TR&)
    (4245)  DC&[0,00,0,0,0,0]  BM[0111,,00,00,,00,10,,110,,111...0,0,0,,0,,0,,,0,0000...0,,0000,0,,,11,100,,,010,111,011]
8459
8460    7026:  ;(FREE)
8461    RE&ETIR350B:
8462            &ETUP,  RETURN/TE&T350C,                ;GO TO &UBR WHICH PUTS &R => IR FOR &F/DF/DM
8463            NEXT,   CALL[&RINTOIR]                  ;
    (7026)  DC&[0,00,0,0,0,0]  BM[0100,,00,11,,11,00,,010,,111...0,0,0,,0,,0,,,0,0000...0,,0000,0,,,11,100,,,010,110,110]
8464
8465
8466
8467    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
8468
8469    ;TE&T 350C CHECK& A&P-ADDR&/DF, A&PLO ADDR& FOR ERR0R&
8470    4742:
8471    TE&T350C:
8472            P0,     LOAD-ENUA(%TARGET434),          ;INSTR& FOR IR=(000000)
8473                    LOAD-ERROR(TE&T350C),           ;ERROR DIRECTORY KEY
8474                    DC&-CTR(C7,),                   ;COMPARE AT TARGET
8475                    BUMP-VERIFY,                    ;COUNT
8476            NEXT,   J/COMP350C                      ;
    (4742)  DC&[1,00,1,0,0,1]  BM[1000,,00,11,,11,00,,011,,100...0,0,0,,0,,0,,,0,0000...0,,0000,0,,,11,000,,,010,100,110]
8477
8478    4246:  ;(FREE)
8479    COMP350C:
8480            P2=T,   D_A&PLO(DF)-XOR-B&PHI[&P], &AVE-D[C],   ;COMPARE RECEIVED;EXPECTED, BITWI&E
8481            NEXT,   J/GOBUT350C                     ;
    (4246)  DC&[0,00,0,0,0,0]  BM[0110,,01,01,,10,10,,000,,111...0,1,0,,0,,0,,,0,0000...0,,0000,0,,,11,000,,,010,100,111]
8482
8483    4247:  ;(FREE)
8484    GOBUT350C:
8485            &ETUP,  RETURN/RE&ETIR350C,             ;GO TO &UBR WHICH:
8486            NEXT,   CALL[DINTOIR=5]                 ;  D => IR, BUT(IN&TR&)
    (4247)  DC&[0,00,0,0,0,0]  BM[0111,,00,00,,00,10,,111,,111...0,0,0,,0,,0,,,0,0000...0,,0000,0,,,11,100,,,010,111,011]
8487
```

```
	7027:	;(FREE)
	RESETIR350C:
		SETUP,	RETURN/TEST350D,			;GO TO SUBR WHICH PUTS SR => IR FOR SF/DF/DM
		NEXT,	CALL[SRINTOIR]				;
(7027)	DCS[0,00,0,0,0,0]  BM[0100,,00,11,,11,00,,011,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,110,110]



	; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

	;TEST 350D CHECKS ASP-ADDRS/SF, ASPHI ADDRS FOR ERRORS
	4743:
	TEST350D:
		P0,	LOAD-ENUA(ZTARGET434),			;INSTRS FOR IR=(000000)
			LOAD-ERROR(TEST350D),			;ERROR DIRECTORY KEY
			DCS-CTR(C7,),				;COMPARE AT TARGET
		NEXT,	J/COMP350D				;
(4743)	DCS[1,00,1,0,0,0]  BM[1000,,00,11,,11,00,,011,,100,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,010,101,000]

	4250:	;(FREE)
	COMP350D:
		P0,	BUMP-VERIFY,				;COUNT
		P2-T,	D_ASPHI[SF]-XOR-ASPLO[DF], SAVE-D[C],	;COMPARE RECEIVED;EXPECTED, BITWISE
		NEXT,	J/GOBUT350D				;
(4250)	DCS[0,00,0,0,0,1]  BM[0110,,00,00,,11,11,,000,,111,,,0,1,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,010,101,001]

	4251:	;(FREE)
	GOBUT350D:
		P0,	BUMP-VERIFY,				;COUNT
		SETUP,	RETURN/SCOPE350,			;GO TO SUBR WHICH:
		NEXT,	CALL[DINTOIR-5]				;  D => IR, BUT(INSTRS)
(4251)	DCS[0,00,0,0,0,1]  BM[0100,,00,01,,01,01,,010,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,111,011]



	4252:	;(FREE)
	SCOPE350:
		P2-T,	D_SR, SAVE-D[C],			;FOR DISPLAY OF SF/DM/DF
		NEXT,	BUTO[SCOPE],				;NO ERROR:  "RESETIR350" (+1, WORDS)  KEEP ON TESTING
			J/RESETIR350				;   ERROR:  "COMP350A" (-11, WORDS)  HOLD UP PATTERN
(4252)	DCS[0,00,0,1,0,0]  BM[1111,,00,00,,00,00,,000,,111,,,0,1,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,111,000,011]



	; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

	4703:
	RESETIR350:
		P0,	BUMP-VERIFY,				;COUNT
		SETUP,	RETURN/NEXTPATA350,			;PUT OLD PAT FROM SR INTO IR FOR DMO TEST
		NEXT,	CALL[SRINTOIR]				;
(4703)	DCS[0,00,0,0,0,1]  BM[0100,,00,01,,01,01,,011,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,110,110]
```



```
	4253:	;(FREE)
	NEXTPATA350:
		P2-T,	SR&D_SR-PLUS-CSPD(D15), SAVE-D[C],	;GENER NEXT PATTERN INTO D, SR
		NEXT,	BUTR(DMO),				;IF TRUE:  "LOAD05-351" (+1, WORDS) ALL DONE HERE
			J/DINTOIRA350				;IF FALSE: "DINTOIRA" (-14, WORDS)  KEEP ON TESTING
(4253)	DCS[0,00,0,0,0,0]  BM[1001,,10,00,,00,00,,000,,111,,,0,1,1,,0,,0,,,0,0010,,,0,,0000,0,,,01,001,,,101,110,011]



	; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

	;COME HERE IF DONE LOOPING
	;THESE CONSTANTS ARE USED IN THE NEXT TESTS:
	4567:
	LOAD05-351:
		P0,	BUMP-VERIFY,				;COUNT
		P3,	CSPD[05]_EMIT, EMIT/055255,		;SF=2, DF=8 CONSTANT
		NEXT,	J/LOAD06-351				;
(4567)	DCS[0,00,0,0,0,1]  BM[0101,,10,10,,10,10,,101,,101,,,0,0,0,,0,,0,,,0,1010,,,1,,0000,0,,,11,000,,,010,101,100]

	4754:	;(FREE)
	LOAD06-351:
		P3,	CSPD[06]_EMIT, EMIT/054344,		;SF=3, DF=4 CONSTANT
		NEXT,	J/TEST351A				;
(4754)	DCS[0,00,0,0,0,0]  BM[0101,,10,10,,00,11,,100,,100,,,0,0,0,,0,,0,,,0,1001,,,1,,0000,0,,,11,000,,,111,011,010]




	; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

	;TESTS 351 A-D VERIFY THAT THE RIF ADDRESS, WITH ASP HI/LO IMMED0/1 MODES, OPERATES CORRECTLY

	; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

	;TEST 351A CHECKS ASPLO RIF ADDR "001 0"=(02)
	4732:
	TEST351A:
		P0,	LOAD-ENUA(ZTARGET434),			;INSTRS FOR IR=(000000)
			LOAD-ERROR(TEST351A),			;ERROR DIRECTORY KEY
			DCS-CTR(C7,),				;COMPARE AT TARGET
		NEXT,	J/COMP351A				;
(4732)	DCS[1,00,1,0,0,0]  BM[1000,,00,11,,11,00,,011,,100,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,111,000,100]

	4704:

	COMP351A:
		P2-T,	D_CSPD[05]-XOR-ASPLO(R02), SAVE-D[C],	;COMPARE EXPECTED;RECEIVED, BITWISE
		NEXT,	J/GOBUT351A				;CSP[05]=(055255)
(4704)	DCS[0,00,0,0,0,0]  BM[0110,,10,00,,10,00,,101,,111,,,0,1,0,,0,,0,,,0,1010,,,0,,0000,0,,,11,000,,,010,101,110]

	4256:	;(FREE)
	GOBUT351A:
		P0,	BUMP-VERIFY,				;COUNT
```

```
                SETUP,  RETURN/TEST351B,                        ;GO TO SUBR WHICH:
                NEXT,   CALL[DINTOIR-5]                         ;  D => IR, BUT[INSTRS]
     (4256)  DCS[0,00,0,0,0,1]  BM[0100,,00,11,,10,10,,010,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,111,011]



        ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        ;TEST 351B CHECKS ASPLO RIF ADDR "001 1"=(03)
        4722:
        TEST351B:
                P0,     LOAD-ENUA(ZTARGET434),                  ;INSTRS FOR IR=(000000)
                        LOAD-ERROR(TEST351B),                   ;ERROR DIRECTORY KEY
                        DCS-CTR[C7,],                           ;COMPARE AT TARGET
                NEXT,   J/COMP351B                              ;
     (4722)  DCS[1,00,1,0,0,0]  BM[1000,,00,11,,11,00,,011,,100,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,010,101,111]

        4257:  ;(FREE)
        COMP351B:
                P2-T,   D_CSPD[06]-XOR-ASPLO(R03), SAVE-D[C],   ;COMPARE EXPECTED:RECEIVED, BITWISE
                NEXT,   J/GOBUT351B                             ;CSP[06]=(054344)
     (4257)  DCS[0,00,0,0,0,0]  BM[0110,,10,00,,10,01,,101,,111,,,0,1,0,,0,,0,,,0,1001,,,0,,0000,0,,,11,000,,,010,110,000]

        4260:  ;(FREE)
        GOBUT351B:
                SETUP,  RETURN/TEST351C,                        ;GO TO SUBR WHICH:
                NEXT,   CALL[DINTOIR-5]                         ;  D => IR, BUT[INSTRS]
     (4260)  DCS[0,00,0,0,0,0]  BM[0100,,00,11,,10,10,,011,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,111,011]



        ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        ;TEST 351C CHECKS ASPHI RIF ADDR "010 0"=(04)
        4723:
        TEST351C:
                P0,     LOAD-ENUA(ZTARGET434),                  ;INSTRS FOR IR=(000000)
                        LOAD-ERROR(TEST351C),                   ;ERROR DIRECTORY KEY
                        DCS-CTR[C7,],                           ;COMPARE AT TARGET
                        BUMP-VERIFY,                            ;COUNT
                NEXT,   J/COMP351C                              ;
     (4723)  DCS[1,00,1,0,0,1]  BM[1000,,00,11,,11,00,,011,,100,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,010,110,001]

        4261:  ;(FREE)
        COMP351C:
                P2-T,   D_CSPD[06]-XOR-ASPHI(R04), SAVE-D[C],   ;COMPARE EXPECTED:RECEIVED, BITWISE
                NEXT,   J/GOBUT351C                             ;CSP[06]=(054344)
     (4261)  DCS[0,00,0,0,0,0]  BM[0110,,10,00,,11,00,,110,,111,,,0,1,0,,0,,0,,,0,1001,,,0,,0000,0,,,11,000,,,010,110,010]

        4262:  ;(FREE)
        GOBUT351C:
                SETUP,  RETURN/TEST351D,                        ;GO TO SUBR WHICH:
                NEXT,   CALL[DINTOIR-5]                         ;  D => IR, BUT[INSTRS]
```



```
     (4262)  DCS[0,00,0,0,0,0]  BM[0100,,00,11,,10,01,,010,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,111,011]


        ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        ;TEST 351D CHECKS ASPHI RIF ADDR "010 1"=(05)
        4712:
        TEST351D:
                P0,     LOAD-ENUA(ZTARGET434),                  ;INSTRS FOR IR=(000000)
                        LOAD-ERROR(TEST351D),                   ;ERROR DIRECTORY KEY
                        DCS-CTR[C7,],                           ;COMPARE AT TARGET
                NEXT,   J/COMP351D                              ;
     (4712)  DCS[1,00,1,0,0,0]  BM[1000,,00,11,,11,00,,011,,100,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,010,110,011]

        4263:  ;(FREE)
        COMP351D:
                P0,     BUMP-VERIFY,                            ;COUNT
                P2-T,   D_CSPD[05]-XOR-ASPHI(R05), SAVE-D[C],   ;COMPARE EXPECTED:RECEIVED, BITWISE
                NEXT,   J/GOBUT351D                             ;CSP[05]=(055355)
     (4263)  DCS[0,00,0,0,0,1]  BM[0110,,10,00,,11,01,,110,,111,,,0,1,0,,0,,0,,,0,1010,,,0,,0000,0,,,11,000,,,010,110,100]

        4264:  ;(FREE)
        GOBUT351D:
                P0,     BUMP-VERIFY,                            ;COUNT
                SETUP,  RETURN/SCOPE351,                        ;GO TO SUBR WHICH:
                NEXT,   CALL[DINTOIR-5]                         ;  D => IR, BUT[INSTRS]
     (4264)  DCS[0,00,0,0,0,1]  BM[0100,,00,01,,01,10,,101,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,111,011]



        4265:  ;(FREE)
        SCOPE351:
                NEXT,   BUTD[SCOPE],                            ;NO ERROR:  "TEST352A" (+1, WORDS)
                        J/TEST352A                              ;   ERROR:  "COMP351A" (-11, WORDS)
     (4265)  DCS[0,00,0,1,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,111,000,101]




        ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        ;TESTS 352 A-D VERIFY THAT THE RIF ADDRESS, WITH BSP HI/LO IMMED0/1 MODES, OPERATES CORRECTLY

        ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        ;TEST 352A CHECKS BSPLO RIF ADDR "001 0"=(02)

        4705:
        TEST352A:
                P0,     LOAD-ENUA(ZTARGET434),                  ;INSTRS FOR IR=(000000)
                        LOAD-ERROR(TEST352A),                   ;ERROR DIRECTORY KEY
                        DCS-CTR[C7,],                           ;COMPARE AT TARGET
                NEXT,   J/COMP352A                              ;
     (4705)  DCS[1,00,1,0,0,0]  BM[1000,,00,11,,11,00,,011,,100,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,111,001,100]
```

```
8675
8676    4714;
8677    COMP352A:
8678            P2=T,   D_ASPLO[02]-XOR-BSPLO(R02), SAVE=D(C),  ;COMPARE EXPECTED:RECEIVED, BITWISE
8679            NEXT,   J/GOBUT352A                             ;DATA=(052355)
    (4714)  DCS(0,00,0,0,0,0)  BM(0110,,00,10,,10,00,,101,,111...0,1,0,,0,,0...0,0000...0,,0000,0...11,000,,,010,110,110)
8680
8681    4266;  ;(FREE)
8682    GOBUT352A:
8683            P0,     BUMP=VERIFY,                            ;COUNT
8684            SETUP,  RETURN/TEST352B,                        ;GO TO SUBR WHICH:
8685            NEXT,   CALL[DINTOIR=5]                         ;  D => IR, BUT(INSTRS)
    (4266)  DCS(0,00,0,0,0,1)  BM(0100,,00,11,,10,01,,011,,111...0,0,0,,0,,0...0,0000...0,,0000,0...11,100,,,010,111,011)
8686
8687
8688    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
8689
8690    ;TEST 352B CHECKS BSPLO RIF ADDR "001 1"=(03)
8691    4713;
8692    TEST352B:
8693            P0,     LOAD=ENUA(ZTARGET434),                  ;INSTRS FOR IR=(000000)
8694                    LOAD=ERROR(TEST352B),                   ;ERROR DIRECTORY KEY
8695                    DCS=CTR(C7,),                           ;COMPARE AT TARGET
8696            NEXT,   J/COMP352B                              ;
    (4713)  DCS(1,00,1,0,0,0)  BM(1000,,00,11,,11,00,,011,,100...0,0,0,,0,,0...0,0000...0,,0000,0...11,000,,,010,110,111)
8697
8698    4267;  ;(FREE)
8699    COMP352B:
8700            P2=T,   D_ASPLO[03]-XOR-BSPLO(R03), SAVE=D(C),  ;COMPARE EXPECTED:RECEIVED, BITWISE
8701            NEXT,   J/GOBUT352B                             ;DATA=(054144)
    (4267)  DCS(0,00,0,0,0,0)  BM(0110,,00,11,,10,01,,101,,111...0,1,0,,0,,0...0,0000...0,,0000,0...11,000,,,010,111,000)
8702
8703    4270;  ;(FREE)
8704    GOBUT352A:
8705            SETUP,  RETURN/TEST352C,                        ;GO TO SUBR WHICH:
8706            NEXT,   CALL[DINTOIR=5]                         ;  D => IR, BUT(INSTRS)
    (4270)  DCS(0,00,0,0,0,0)  BM(0100,,00,11,,10,10,,100,,111...0,0,0,,0,,0...0,0000...0,,0000,0...11,100,,,010,111,011)
8707
8708
8709    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
8710
8711    ;TEST 352C CHECKS BSPLO RIF ADDR "010 0"=(04)
8712    4724;
8713    TEST352C:
8714            P0,     LOAD=ENUA(ZTARGET434),                  ;INSTRS FOR IR=(000000)
8715                    LOAD=ERROR(TEST352C),                   ;ERROR DIRECTORY KEY
8716                    DCS=CTR(C7,),                           ;COMPARE AT TARGET
8717                    BUMP=VERIFY,                            ;COUNT
8718            NEXT,   J/COMP352C                              ;
    (4724)  DCS(1,00,1,0,0,1)  BM(1000,,00,11,,11,00,,011,,100...0,0,0,,0,,0...0,0000...0,,0000,0...11,000,,,010,111,001)
8719
8720    4271;  ;(FREE)
```



```
8721    COMP352C:
8722            P2=T,   D_ASPLO[04]-XOR-BSPLO(R04), SAVE=D(C),  ;COMPARE EXPECTED:RECEIVED, BITWISE
8723            NEXT,   J/GOBUT352C                             ;DATA=(053433)
    (4271)  DCS(0,00,0,0,0,0)  BM(0110,,00,10,,10,00,,110,,111...0,1,0,,0,,0...0,0000...0,,0000,0...11,000,,,010,111,010)
8724
8725    4272;  ;(FREE)
8726    GOBUT352C:
8727            SETUP,  RETURN/TEST352D,                        ;GO TO SUBR WHICH:
8728            NEXT,   CALL[DINTOIR=5]                         ;  D => IR, BUT(INSTRS)
    (4272)  DCS(0,00,0,0,0,0)  BM(0100,,00,11,,10,10,,101,,111...0,0,0,,0,,0...0,0000...0,,0000,0...11,100,,,010,111,011)
8729
8730
8731    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
8732
8733    ;TEST 352D CHECKS BSPLO RIF ADDR "010 1"=(05)
8734    4725;
8735    TEST352D:
8736            P0,     LOAD=ENUA(ZTARGET434),                  ;INSTRS FOR IR=(000000)
8737                    LOAD=ERROR(TEST352D),                   ;ERROR DIRECTORY KEY
8738                    DCS=CTR(C7,),                           ;COMPARE AT TARGET
8739            NEXT,   J/COMP352D                              ;
    (4725)  DCS(1,00,1,0,0,0)  BM(1000,,00,11,,11,00,,011,,100...0,0,0,,0,,0...0,0000...0,,0000,0...11,000,,,010,111,011)
8740
8741    4273;  ;(FREE)
8742    COMP352D:
8743            P0,     BUMP=VERIFY,                            ;COUNT
8744            P2=T,   D_ASPLO[05]-XOR-BSPLO(R05), SAVE=D(C),  ;COMPARE EXPECTED:RECEIVED, BITWISE
8745            NEXT,   J/GOBUT352D                             ;DATA=(052522)
    (4273)  DCS(0,00,0,0,0,1)  BM(0110,,00,11,,10,01,,110,,111...0,1,0,,0,,0...0,0000...0,,0000,0...11,000,,,010,111,100)
8746
8747    4274;  ;(FREE)
8748    GOBUT352D:
8749            P0,     BUMP=VERIFY,                            ;COUNT
8750            SETUP,  RETURN/SCOPE352,                        ;GO TO SUBR WHICH:
8751            NEXT,   CALL[DINTOIR=5]                         ;  D => IR, BUT(INSTRS)
    (4274)  DCS(0,00,0,0,0,1)  BM(0100,,00,01,,01,11,,101,,111...0,0,0,,0,,0...0,0000...0,,0000,0...11,100,,,010,111,011)
8752
8753
8754
8755    4275;  ;(FREE)
8756    SCOPE352:
8757            P2=T,   D_CSPD(D13), D(C)_D,                    ;RESTORE CONSTANT (000000)
8758            P3,     A=BSPHI[01]_D,                          ;
8759            NEXT,   RUTD[SCOPE],                            ;NO ERROR:  "RESTORE01" (+1, WORDS)
8760                    J/RESTORE01                             ;   ERROR:  "COMP352A"  (-15, WORDS)
    (4275)  DCS(0,00,0,1,0,0)  BM(1010,,10,11,,00,01,,100,,000...0,1,0,,0,,0...0,0100,,,0,,1011,0,,,11,000,,,111,001,101)
8761
8762
8763    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
8764
8765    ;       THIS SECTION OF CODE FINISHES RESTORING THE 4 CONSTANTS IN ASPHI/BSPHI
8766    ;       THAT WERE WIPED OUT IN THE PREVIOUS GROUP OF TESTS.
```

```
47151
RESTORE01:
        P2-T,   D_CSPD(D10),                            ;(052525)
        P3,     A+BSPHI[07]_D,                          ;
        NEXT,   GOTO-PAGE(7),                           ;
                J/RESTORE02                             ;
(4715)  DCS[0,00,0,0,0,0]  BM[1010,,10,11,,00,01,,111,,111,,,0,1,0,,0,,0,,,0,0111,,,0,,1011,0,,,11,100,,,000,011,000]

7030:   ;(FREE)
RESTORE02:
        P2-T,   D_CSPD(D11), D[C]_0,                    ;(125252)
        P3,     A+BSPHI[05]_D,                          ;
        NEXT,   J/RESTORE03                             ;
(7030)  DCS[0,00,0,0,0,0]  BM[1010,,10,11,,00,01,,110,,000,,,0,1,0,,0,,0,,,0,0110,,,0,,1011,0,,,11,000,,,000,011,010]

7032:   ;(FREE)
RESTORE03:
        P2-T,   D_CSPD(D12), D[C]_0,                    ;(177777)
        P3,     A+BSPHI[03]_D,                          ;
        NEXT,   J/RESTORE04                             ;
(7032)  DCS[0,00,0,0,0,0]  BM[1010,,10,11,,00,01,,101,,000,,,0,1,0,,0,,0,,,0,0101,,,0,,1011,0,,,11,000,,,000,011,011]

7033:   ;(FREE)
RESTORE04:
        SETUP,  RETURN/TEST361A,                        ;GO TO SUBR THAT RESTORES CURRENT
        NEXT,   GOTO-PAGE(7),                           ;  DCS MICROCODE VERSION NUMBER,
                J/INSERTREVNO                           ;  B<15:>=(0), INTO B,M, GPR "R5"
(7033)  DCS[0,00,0,0,0,0]  BM[0110,,00,10,,11,11,,011,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,001,110]




;.PAGE;==========================================================================================

.TOC    "       TEST361-372:  TESTING SR, GUARD, RES, AND XMUX

;*******************************************************************************
;*
;*      TESTS:  361 - 372                               UWORDS: 125 + 117
;*
;*      FUNCTIONS:      TESTS THAT SR CAN BE LOADED FROM ALU OUT, READ THRU
;*                      SR AND FLOAT PORTS OF XMUX, SR CAN BE SHIFTED LEFT, RIGHT,
;*                      AND NOP, GUARD-ENABLED, DISABLED, SHIFTED, AND THAT
;*                      ANY "BUTTABLE" BITS CAN BE TESTED.
;*                      PREVIOUSLY TESTED IN TEST 105 A-E WAS THE ABILITY TO
;*                      LOAD/READ THE SR THRU XMUX/SR, AND BUT(SR3=0),
;*                      THESE FUNCTIONS ARE NOT TESTED HERE.
;*
;*******************************************************************************
```



```
; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

;*** TEST 361A ***
;TEST-361A TESTS THAT SR CAN BE LOADED, GUARD CLEARED, WITH PATTERN (052525)(0);
;READ THRU XMUX-FLTPT PORT = (100125)
6573:
TEST361A:
        P0,     LOAD-ENUA(ZTARGET434),                  ;SETUP FOR IR=(000000)/INSTPS TEST
                LOAD-ERROR(TEST361A),                   ;ERROR DIRECTORY KEY
                DCS-CTR(C11,),                          ;COMPARE ENUA:TNUA AT TARGET
        NEXT,   J/SETRES361A                            ;
(6573)  DCS[1,00,1,0,0,0]  BM[0100,,00,11,,11,00,,011,,100,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,101,111,000]

6570:
SETRES361A:
        P1,     CSPD[16]_EMIT,                          ;CSP GETS
        EMITC,  SENDMUX-4567-SEL,                       ;RES REG VALUES
                SR-LOAD, GUARD-EN,                      ;
        NEXT,   J/LOADRES361A                           ;
(6570)  DCS[0,00,0,0,0,0]  BM[0000,,10,10,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0001,,,1,,0000,0,,,11,000,,,000,000,001]

6001:   ;(FREE)
LOADRES361A:
        P0,     BUMP-VERIFY,                            ;COUNT
        P2-T,   RES_CSPB(B16),                          ;STORE RES
                D_ABPHI(C000000), D[C]_1,               ;SET D=ZEROES, D[C]=1
        NEXT,   J/LOADSR361A                            ;
(6001)  DCS[0,00,0,0,0,1]  BM[1111,,11,01,,11,01,,100,,000,,,0,1,0,,0,,0,,,0,0000,,,0,,1000,1,,,11,000,,,000,000,010]

6002:   ;(FREE)
LOADSR361A:
        P2-T,   SR_BSPHI(C052525),                      ;LOAD SR WITH DATA
        NEXT,   J/EXPEC361B                             ;
(6002)  DCS[0,00,0,0,0,0]  BM[1010,,01,11,,00,00,,111,,000,,,0,0,1,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,000,011]

6003:   ;(FREE)
EXPEC361B:
        P0,     BUMP-VERIFY,                            ;COUNT
        P3,     CSPD[16]_EMIT,                          ;EXPECTED VALUE OUT OF XMUX-FLOAT
                EMIT/100125,                            ;(100125)
        NEXT,   J/COMP361B                              ;

(6003)  DCS[0,00,0,0,0,1]  BM[1000,,10,00,,00,01,,010,,101,,,0,0,0,,0,,0,,,0,0001,,,1,,0000,0,,,11,000,,,000,000,100]

6004:   ;(FREE)
COMP361B:
        SETUP,  RETURN/TEST361D,                        ;RETURN TO START OF NEXT SUBTEST
        NEXT,   CALL(CSP16XORFLTTOIR-S)                 ;SUBR: CSP(16).XOR.FLOAT -> IR, BUT(INSTR5)
(6004)  DCS[0,00,0,0,0,0]  BM[0110,,00,11,,11,11,,001,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,000,011,101]
```

```
	; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

	;*** TEST 361D ***
	;TEST 361D CHECKS THAT THE BUT ON BR<1:0> READS THE (0S2825) IN THE BR CORRECTLY
	6771:
	TEST361D:
		P0,	LOAD-ENUA(ZTARGET401),			;EXPECTED VALUE "01"8"1" IN BR<1:0>
			LOAD-ERROR(TEST361D),			;ERROR DIRECTORY KEY
			DCS-CTR(C3,),				;COMPARE ENUA:TNUA AT TARGET
		NEXT,	J/GOBUT361D				;
    (6771)  DCS[1,00,1,0,0,0]  BM[1100,,00,11,,11,00,,000,,011,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,000,101]

	6005:	;(FREE)
	GOBUT361D:
		SETUP,	RETURN/TEST361E,			;RETURN TO START OF NEXT SUBTEST
		NEXT,	GOTO-PAGE(7),				;BUT'S ARE ON PAGE 7
			J/BUTBR1-0				;GO BUT ON BR<1:0>
    (6005)  DCS[0,00,0,0,0,0]  BM[0110,,00,11,,11,10,,111,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,001,101]





	; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

	;*** TEST 361E ***
	;TEST-361E CHECKS THAT GD<3:2> IS CLEARED UNDER BR-LOAD, GUARD-ENABLED
	6767:
	TEST361E:
		P0,	LOAD-ENUA(ZTARGET400),			;EXPECTED VALUE "00" IN GUARD<3:2>
			LOAD-ERROR(TEST361E),			;ERROR DIRECTORY KEY
			DCS-CTR(C3,),				;COMPARE ENUA:TNUA AT TARGET
			BUMP-VERIFY,				;COUNT
		NEXT,	J/GOBUT361E				;
    (6767)  DCS[1,00,1,0,0,1]  BM[1100,,00,11,,11,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,000,110]

	6006:	;(FREE)
	GOBUT361E:
		SETUP,	RETURN/TEST362A,			;RETURN TO START OF NEXT SUBTEST
		NEXT,	GOTO-PAGE(7),				;BUT'S ARE ON PAGE 7
			J/BUTGD3-2				;GO BUT ON GD<3:2>
    (6006)  DCS[0,00,0,0,0,0]  BM[0110,,00,11,,11,10,,101,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,001,100]
```



```
	; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

	;*** TEST 362A ***
	;TEST-362A CHECKS THAT THE BR&GD CAN BE SHIFTED RIGHT 1 POSITION, INSERTING
	;A "1" FROM D<00> INTO BR<15>, SHIFTING THE "1" IN BR<00> INTO GD<3>
	6765:
	TEST362A:
		P0,	LOAD-ENUA(ZTARGET434),			;SETUP FOR IR=(000000)/INSTRS TEST
			LOAD-ERROR(TEST362A),			;ERROR DIRECTORY KEY
			DCS-CTR(C10,),				;COMPARE ENUA:TNUA AT TARGET
		NEXT,	J/SETRES362A				;
    (6765)  DCS[1,00,1,0,0,0]  BM[0101,,00,11,,11,00,,011,,100,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,000,111]

	6007:	;(FREE)
	SETRES362A:
		P0,	BUMP-VERIFY,				;COUNT
		P3,	CSPD[16]_EMIT,				;CSP GETS
		EMITC,	SENDMUX-4567-SEL,			;RES REG VALUES
			SR-RIGHT, GUARD-EN,			;
		NEXT,	J/SETDDC362A				;
    (6007)  DCS[0,00,0,0,0,1]  BM[0010,,10,10,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0001,,,1,,0000,0,,,11,000,,,000,001,000]

	6010:	;(FREE)
	SETDDC362A:
		P2-T,	D_BSPHI(C000001), D(C)_0,		;SETUP D<00>=1 FOR SHIFT
		NEXT,	J/LOADRES362A				;
    (6010)  DCS[0,00,0,0,0,0]  BM[1010,,01,11,,00,00,,000,,000,,,0,1,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,001,001]

	6011:	;(FREE)
	LOADRES362A:
		P2,	RES_CSPB(B16),				;STORE RES
		P3-T,	SR_SR-RIGHT-1,				;SHIFT SR RIGHT, GUARD-ENABLED
		NEXT,	J/COMP362A				;
    (6011)  DCS[0,00,0,0,0,0]  BM[0000,,11,01,,00,00,,000,,000,,,1,0,1,,0,,0,,,0,0000,,,0,,1000,1,,,11,000,,,000,001,010]

	6012:	;(FREE)
	COMP362A:
		P0,	BUMP-VERIFY,				;COUNT
		P2-T,	D_SR-XOR-BSPHI(C125252), SAVE-D(C),	;D _ (125252)+SR, BITWISE
		NEXT,	J/GOBUT362A				;
    (6012)  DCS[0,00,0,0,0,1]  BM[0110,,01,11,,00,00,,110,,111,,,0,1,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,001,011]

	6013:	;(FREE)
	GOBUT362A:
		SETUP,	RETURN/TEST362B,			;RETURN TO START OF NEXT SUBTEST
		NEXT,	CALL(DIRTOIR-5)				;SUBR: D -> IR, BUT(INSTRS)

    (6013)  DCS[0,00,0,0,0,0]  BM[0110,,00,11,,11,10,,011,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,111,011]





	; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
```

```
    !*** TEST 362B ***
    !TEST-362B CHECKS THAT SR-XMUX-FLOAT PORT CAN BE READ, WITH SR=(125252)
    6763:
    TEST362B:
            P0,     LOAD-ENUA(ZTARGET434),                  !SETUP FOR IR=(000000)/INSTRS TEST
                    LOAD-ERROR(TEST362B),                   !ERROR DIRECTORY KEY
                    DCS-CTR(C8,),                           !COMPARE ENUA:TNUA AT TARGET
            NEXT,   J/EXPEC362B                             !
  (6763)  DCS[1,00,1,0,0,0]  BM[0011,,00,11,,11,00,,011,,100,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,001,100]

    6014:  !(FREE)
    EXPEC362B:
            P0,     BUMP-VERIFY,                            !COUNT
            P3,     CSPD(16)_EMIT,                          !EXPECTED VALUE OF XMUX-FLOAT
                    EMIT/000052,                            !(000052)
            NEXT,   J/COMP362B                              !
  (6014)  DCS[0,00,0,0,0,1]  BM[0000,,10,00,,00,00,,101,,010,,,0,0,0,,0,,0,,,0,0001,,,1,,0000,0,,,11,000,,,000,001,101]

    6015:  !(FREE)
    COMP362B:
            SETUP,  RETURN/TEST362D,                        !RETURN TO START OF NEXT SUBTEST
            NEXT,   CALL[CSP16XORFLTTOIR-S]                 !SUBR: CSP(16),XOR,FLOAT => IR, BUT(INSTRS)
  (6015)  DCS[0,00,0,0,0,0]  BM[0110,,00,11,,11,01,,111,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,000,011,101]





    ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

    !*** TEST 362D ***
    !TEST-362D CHECKS THAT THE BUT ON SR<1:0> READS THE (125252) IN THE SR CORRECTLY
    6757:
    TEST362D:
            P0,     LOAD-ENUA(ZTARGET405),                  !EXPECTED VALUE "10"&"1" IN SR<1:0>
                    LOAD-ERROR(TEST362D),                   !ERROR DIRECTORY KEY
                    DCS-CTR(C3,),                           !COMPARE ENUA:TNUA AT TARGET
            NEXT,   J/GOBUT362D                             !
  (6757)  DCS[1,00,1,0,0,0]  BM[1100,,00,11,,11,00,,000,,101,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,001,110]

    6016:  !(FREE)
    GOBUT362D:
            SETUP,  RETURN/TEST362E,                        !RETURN TO START OF NEXT SUBTEST
            NEXT,   GOTO-PAGE(7),                           !BUT'S ARE ON PAGE 7
                    J/BUTSR1-0                              !GO BUT ON SR<1:0>
  (6016)  DCS[0,00,0,0,0,0]  BM[0110,,00,11,,11,01,,101,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,001,101]
```



```
    ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

    !*** TEST 362E ***
    !TEST-362E CHECKS THAT GD<3:2> RECEIVED THE "1" IN SR<00> AFTER THE SHIFT RIGHT
    6755:
    TEST362E:
            P0,     LOAD-ENUA(ZTARGET402),                  !EXPECTED VALUE "10" IN GUARD<3:2>
                    LOAD-ERROR(TEST362E),                   !ERROR DIRECTORY KEY
                    DCS-CTR(C3,),                           !COMPARE ENUA:TNUA AT TARGET
            NEXT,   J/GOBUT362E                             !
  (6755)  DCS[1,00,1,0,0,0]  BM[1100,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,001,111]

    6017:  !(FREE)
    GOBUT362E:
            SETUP,  RETURN/TEST363A,                        !RETURN TO START OF NEXT SUBTEST
            P0,     BUMP-VERIFY,                            !COUNT
            NEXT,   GOTO-PAGE(7),                           !BUT'S ARE ON PAGE 7
                    J/BUTGD3-2                              !GO BUT ON GD<3:2>
  (6017)  DCS[0,00,0,0,0,1]  BM[0110,,00,11,,11,01,,011,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,001,100]





    ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

    !*** TEST 363A ***
    !TEST-363A CHECKS THAT THE SR&GD CAN AGAIN BE SHIFTED RIGHT 1 POSITION, INSERTING
    !A "0" FROM D<00> INTO SR<15>, SHIFTING THE "0" IN SR<00> INTO GD<3>
    6753:
    TEST363A:
            P0,     LOAD-ENUA(ZTARGET434),                  !SETUP FOR IR=(000000)/INSTRS TEST
                    LOAD-ERROR(TEST363A),                   !ERROR DIRECTORY KEY
                    DCS-CTR(C9,),                           !COMPARE ENUA:TNUA AT TARGET
            NEXT,   J/SETODC363A                            !
  (6753)  DCS[1,00,1,0,0,0]  BM[0110,,00,11,,11,00,,011,,100,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,010,000]

    6020:  !(FREE)
    SETODC363A:
            P3-T,   D_NOT-ASPHI(C000001), D[C]_1,           !SETUP D=(177776), D(C)=0 FOR SHIFT RIGHT
            NEXT,   J/SHIFT363A                             !
  (6020)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,11,01,,000,,000,,,1,1,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,010,001]

    6021:  !(FREE)
    SHIFT363A:
            P0,     BUMP-VERIFY,                            !COUNT

            P2-T,   SR_SR-RIGHT-1,                          !SHIFT SR RIGHT, GUARD=ENABLED
            NEXT,   J/COMP363A                              !
  (6021)  DCS[0,00,0,0,0,1]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,1,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,010,010]

    6022:  !(FREE)
    COMP363A:
            P2-T,   D_SR-XOR-ASPHI(C052525), SAVE-D[C],     !D _ (052525)#SR, BITWISE
```

```
                NEXT,   J/GOBUT363A                                     !
     (6027)  DCS[0,00,0,0,0,0]  BN[0110,,01,11,,00,00,,111,,111,,,0,1,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,010,011]

        6023:  !(FREE)
        GOBUT363A:
                SETUP,  RETURN/TEST363B,                                !RETURN TO START OF NEXT SUBTEST
                NEXT,   CALL[DINTOIR-5]                                 !SUBR; D => IR, BUT(INSTRS)
     (6023)  DCS[0,00,0,0,0,0]  BN[0110,,00,11,,11,01,,001,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,111,011]





        ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        !*** TEST 363B ***
        !TEST-363B CHECKS THE GUARD REGISTER WAS SHIFTED RIGHT ALSO
        6751:
        TEST363B:
                P0,     LOAD-ENUA(ZTARGET401),                          !EXPECTED VALUE "01" IN GUARD<3:2>
                        LOAD-ERROR(TEST363B),                           !ERROR DIRECTORY KEY
                        DCS-CTR(C3,),                                   !COMPARE ENUA:TNUA AT TARGET
                NEXT,   J/GOBUT363B                                     !
     (6751)  DCS[1,00,1,0,0,0]  BN[1100,,00,11,,11,00,,000,,001,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,010,100]

        6024:  !(FREE)
        GOBUT363B:
                SETUP,  RETURN/TEST364A,                                !RETURN TO START OF NEXT SUBTEST
                NEXT,   GOTO-PAGE(7),                                   !BUT'S ARE ON PAGE 7
                        J/BUTGD3-2                                      !GO BUT ON GD<3:2>
     (6024)  DCS[0,00,0,0,0,0]  BN[0110,,00,11,,11,00,,111,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,001,100]





        ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        !*** TEST 364A ***
        !TEST-364A/364B CHECK THAT WE ARE ABLE TO FILTER A "1" DOWN TO GD<0>, AND
        !SUBSEQUENTLY ABLE TO RECOVER IT, SO GD<3:0>="0101"
        6747:
        TEST364A:
                P0,     LOAD-ENUA(ZTARGET402),                          !EXPECTED VALUE "10" IN GUARD <3:2>
                        LOAD-ERROR(TEST364A),                           !ERROR DIRECTORY KEY
                        DCS-CTR(C4,),                                   !COMPARE ENUA:TNUA AT TARGET
                        BUMP-VERIFY,                                    !COUNT
                NEXT,   J/SETDDC364A                                    !
     (6747)  DCS[1,00,1,0,0,1]  BN[1011,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,010,101]

        6025:  !(FREE)
        SETDDC364A:
```



```
                P2-T,   D_NOT-ASPHI(C000001), D[C]_1,                   !SETUP D=(177776), D[C]=0 FOR SHIFT RIGHT
                NEXT,   J/GOBUT364A                                     !  WHERE  SR<15> <- (0)
     (6025)  DCS[0,00,0,0,0,0]  BN[0000,,00,00,,11,01,,000,,000,,,0,1,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,010,110]

        6026:  !(FREE)
        GOBUT364A:
                SETUP,  RETURN/TEST364B,                                !RETURN TO START OF NEXT SUBTEST
                P3-T,   SR_SR-RIGHT-1,                                  !SHIFT SR+GD RIGHT
                NEXT,   GOTO-PAGE(7),                                   !BUT'S ARE ON PAGE 7
                        J/BUTGD3-2                                      !GO BUT ON GD<3:2>
     (6026)  DCS[0,00,0,0,0,0]  BN[0110,,00,11,,11,00,,101,,111,,,1,0,1,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,001,100]





        ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        !*** TEST 364B ***
        !TEST-364B CHECKS THAT GD<3:2> READS AS "01"
        6745:
        TEST364B:
                P0,     LOAD-ENUA(ZTARGET401),                          !EXPECTED VALUE "01" IN GUARD<3:2>
                        LOAD-ERROR(TEST364B),                           !ERROR DIRECTORY KEY
                        DCS-CTR(C4,),                                   !COMPARE ENUA:TNUA AT TARGET
                NEXT,   J/SETDDC364B                                    !
     (6745)  DCS[1,00,1,0,0,0]  BN[1011,,00,11,,11,00,,000,,001,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,010,111]

        6027:  !(FREE)
        SETDDC364B:
                P3-T,   D_BSPHI(C000001), D[C]_0,                       !SETUP D, D(C) FOR SR<15>_"1"
                NEXT,   J/GOBUT364B                                     !
     (6027)  DCS[0,00,0,0,0,0]  BN[1010,,01,11,,00,00,,000,,000,,,1,1,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,011,000]

        6030:  !(FREE)
        GOBUT364B:
                SETUP,  RETURN/TEST365A,                                !RETURN TO START OF NEXT SUBTEST
                P2-T,   SR_SR-RIGHT-1,                                  !SHIFT SR+GD RIGHT AGAIN
                NEXT,   GOTO-PAGE(7),                                   !BUT'S ARE ON PAGE 7
                        J/BUTGD3-2                                      !GO BUT ON GD<3:2>
     (6030)  DCS[0,00,0,0,0,0]  BN[0110,,00,11,,11,00,,011,,111,,,0,0,1,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,001,100]





        ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        !*** TEST 365A ***
        !TEST-365A CHECKS THAT CAN SHIFT SR-RIGHT, W/GUARD-DISABLED
        6743:
        TEST365A:
```

```
            P0,     LOAD-ENUA(ZTARGET434),                  |SETUP FOR IR=(000000)/INSTRS TEST
                    LOAD-ERROR(TEST365A),                   |ERROR DIRECTORY KEY
                    DCS-CTR(C10,),                          |COMPARE ENUA:TNUA AT TARGET
            NEXT,   J/SETRES365A                            |
   (6743)  DCS[1,00,1,0,0,0]  BM[0101,,00,11,,11,00,,011,,100,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,011,001]

    6031:  |(FREE)
    SETRES365A:
            P0,     BUMP-VERIFY,                            |COUNT
            P3,     CSPD[16]_EMIT,                          |CSP GETS
            EMITC,  SENDMUX-4567-SEL,                       |RES REG VALUES
                    SR-RIGHT, GUARD-DIS,                    |
            NEXT,   J/LOADRES365A                           |
   (6031)  DCS[0,00,0,0,0,1]  BM[0010,,10,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0001,,,1,,0000,0,,,11,000,,,000,011,010]

    6032:  |(FREE)
    LOADRES365A:
            P2,     RES_CSPB(B16),                          |STORE RES
            P3-T,   SR_SR-RIGHT-1,                          |SHIFT SR RIGHT, GUARD-DISABLED
            NEXT,   J/EXPEC365A                             |
   (6032)  DCS[0,00,0,0,0,0]  BM[0000,,11,01,,00,00,,000,,000,,,1,0,1,,0,,0,,,0,0000,,,0,,1000,1,,,11,000,,,000,011,011]

    6033:  |(FREE)
    EXPEC365A:
            P0,     BUMP-VERIFY,                            |COUNT
            P3,     CSPD[16]_EMIT,                          |EXPECTED VALUE OUT OF XMUX-SR
                    EMIT/145252,                            |(145252)
            NEXT,   J/COMP365A                              |
   (6033)  DCS[0,00,0,0,0,1]  BM[1100,,10,10,,10,10,,101,,010,,,0,0,0,,0,,0,,,0,0001,,,1,,0000,0,,,11,000,,,000,011,100]

    6034:  |(FREE)
    COMP365A:
            SETUP,  RETURN/TEST365B,                        |RETURN TO START OF NEXT SUBTEST
            NEXT,   CALL[CSP16XORSRYDIR-5]                  |SUBR; CSP(16),XOR,SR => IR, BUT(INSTRS)
   (6034)  DCS[0,00,0,0,0,0]  BM[0110,,00,11,,11,00,,001,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,000,011,001]





    | - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

    |*** TEST 365B ***
    |TEST-365B CHECKS THAT THE GUARD WASN'T ALTERED ON SHIFT RIGHT/DISABLED
    6741:
    TEST365B:
            P0,     LOAD-ENUA(ZTARGET401),                  |EXPECTED VALUE "01" IN GUARD<3:2>
                    LOAD-ERROR(TEST365B),                   |ERROR DIRECTORY KEY
                    DCS-CTR(C3,),                           |COMPARE ENUA:TNUA AT TARGET
            NEXT,   J/GOBUT365B                             |
   (6741)  DCS[1,00,1,0,0,0]  BM[1100,,00,11,,11,00,,000,,001,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,011,101]

    6035:  |(FREE)
```



```
    GOBUT365B:
            SETUP,  RETURN/TEST366A,                        |RETURN TO START OF NEXT SUBTEST
            NEXT,   GOTO-PAGE(7),                           |BUT'S ARE ON PAGE 7
                    J/BUTGD3-2                              |GO BUT ON GD<3:2>
   (6035)  DCS[0,00,0,0,0,0]  BM[0110,,00,11,,10,11,,111,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,001,100]





    | - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

    |*** TEST 366A ***
    |TEST-366A CHECKS THAT SR CAN SHIFT LEFT, GUARD-DISABLED & NOT ALTERED, "0" IN
    |D(C) SHIFTED INTO SR<00>
    6737:
    TEST366A:
            P0,     LOAD-ENUA(ZTARGET434),                  |SETUP FOR IR=(000000)/INSTRS TEST
                    LOAD-ERROR(TEST366A),                   |ERROR DIRECTORY KEY
                    DCS-CTR(C11,),                          |COMPARE ENUA:TNUA AT TARGET
            NEXT,   J/SETRES366A                            |
   (6737)  DCS[1,00,1,0,0,0]  BM[0100,,00,11,,11,00,,011,,100,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,011,110]

    6036:  |(FREE)
    SETRES366A:
            P0,     BUMP-VERIFY,                            |COUNT
            P3,     CSPD[16]_EMIT,                          |CSP GETS
            EMITC,  SENDMUX-4567-SEL,                       |RES REG VALUES
                    SR-LEFT, GUARD-DIS,                     |
            NEXT,   J/LOADRES366A                           |
   (6036)  DCS[0,00,0,0,0,1]  BM[0001,,10,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0001,,,1,,0000,0,,,11,000,,,000,011,111]

    6037:  |(FREE)
    LOADRES366A:
            P2-T,   RES_CSPB(B16),                          |STORE RES
                    D_ZERO, D[C]_ALU07,                     |AND SET D, D[C]
            NEXT,   J/SHIFT366A                             |
   (6037)  DCS[0,00,0,0,0,0]  BM[0011,,11,01,,00,00,,000,,011,,,0,1,0,,0,,0,,,0,0000,,,0,,1000,1,,,11,000,,,000,100,000]

    6040:  |(FREE)
    SHIFT366A:
            P0,     BUMP-VERIFY,                            |COUNT
            P3-T,   SR_SR-LEFT-1,                           |SHIFT SR LEFT 1, SR<00>_D[C]="0"
            NEXT,   J/EXPEC366A                             |
   (6040)  DCS[0,00,0,0,0,1]  BM[0000,,00,00,,00,00,,000,,000,,,1,0,1,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,100,001]

    6041:  |(FREE)
    EXPEC366A:
            P3,     CSPD[16]_EMIT,                          |EXPECTED RESULT AFTER SHIFT
                    EMIT/112524,                            |(112524)
            NEXT,   J/COMP366A                              |
   (6041)  DCS[0,00,0,0,0,0]  BM[1001,,10,01,,01,01,,010,,100,,,0,0,0,,0,,0,,,0,0001,,,1,,0000,0,,,11,000,,,000,100,010]
```

```
	6042:	!(FREE)
	COMP366A:
		SETUP,	RETURN/TEST366B,			!RETURN TO START OF NEXT SUBTEST
		NEXT,	CALL[CSP16XORSRTDIR-5]			!SUBR: CSP(16),XOR,SR => IR, BUT(INSTRS)
 (6042)	DCS[0,00,0,0,0,0]  BM[0110,,00,11,,10,11,,101,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,000,011,001]






	! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

	!*** TEST 366B ***
	!TEST-366B CHECKS THAT GUARD WASN'T ALTERED ON SHIFT LEFT/GUARD-DISABLED
	6735:
	TEST366B:
		P0,	LOAD-ENUA(ZTARGET401),			!GUARD SHOULD STILL BE "01"*"01"
			LOAD-ERROR(TEST366B),			!ERROR DIRECTORY KEY
			DCS-CTR(C3,),				!COMPARE ENUA:TNUA AT TARGET
		NEXT,	J/GOBUT366B
 (6735)	DCS[1,00,1,0,0,0]  BM[1100,,00,11,,11,00,,000,,001,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,100,011]

	6043:	!(FREE)
	GOBUT366B:
		SETUP,	RETURN/TEST366C,			!RETURN TO START OF NEXT SUBTEST
		NEXT,	GOTO-PAGE(7),				!BUT'S ARE ON PAGE 7
			J/BUTGD3-2				!GO BUT ON GUARD <3:2>
 (6043)	DCS[0,00,0,0,0,0]  BM[0110,,00,11,,10,11,,011,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,001,100]






	! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

	!*** TEST 366C ***
	!TEST-366C CHECKS THAT SR CAN AGAIN SHIFT LEFT, GUARD-DISABLED, SR<00>_D[C]="1"
	6733:
	TEST366C:
		P0,	LOAD-ENUA(ZTARGET434),			!SETUP FOR IR=(000000)/INSTRS TEST
			LOAD-ERROR(TEST366C),			!ERROR DIRECTORY KEY
			DCS-CTR(C10,),				!COMPARE ENUA:TNUA AT TARGET
		NEXT,	J/SETDDC366C
 (6733)	DCS[1,00,1,0,0,0]  BM[0101,,00,11,,11,00,,011,,100,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,100,100]

	6044:	!(FREE)
	SETDDC366C:
		P2-T,	D_ASRHI(C177777), D[C]_1,		!SETUP D, D[C]
		NEXT,	J/SHIFT366C				!
 (6044)	DCS[0,00,0,0,0,0]  BM[1111,,00,00,,11,01,,101,,000,,,0,1,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,100,101]
```



```

	6045:	!(FREE)
	SHIFT366C:
		P2-T,	SR_SR-LEFT-1,				!SHIFT SR LEFT 1, SR<00>_D[C]="1"
		NEXT,	J/EXPEC366C
 (6045)	DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,1,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,100,110]

	6046:	!(FREE)
	EXPEC366C:
		P0,	BUMP-VERIFY,				!COUNT
		P3,	CSPD[16]_EMIT,				!EXPECTED VALUE AFTER SHIFT
			EMIT/025251,				!(025251)
		NEXT,	J/COMP366C				!
 (6046)	DCS[0,00,0,0,0,1]  BM[0010,,10,10,,10,10,,101,,001,,,0,0,0,,0,,0,,,0,0001,,,1,,0000,0,,,11,000,,,000,100,111]

	6047:	!(FREE)
	COMP366C:
		SETUP,	RETURN/TEST367A,			!RETURN TO START OF NEXT SUBTEST
		NEXT,	CALL[CSP16XORSRTDIR-5]			!SUBR: CSP(16),XOR,SR => IR, BUT(INSTRS)
 (6047)	DCS[0,00,0,0,0,0]  BM[0110,,00,11,,10,11,,001,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,000,011,001]






	! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

	!*** TEST 367A ***
	!TEST-367A CHECKS THAT LOAD/GUARD-DISABLED LOADS SR, GUARD NOT ALTERED
	!CHECK THAT BUTA(CLEAR-FLAGS) CLEARS RES TO SR-LOAD/GUARD-DIS
	6731:
	TEST367A:
		P0,	LOAD-ENUA(ZTARGET401),			!GUARD SHOULD STILL BE "01"*"01" AFTER LOAD
			LOAD-ERROR(TEST367A),			!ERROR DIRECTORY KEY
			DCS-CTR(C6,),				!COMPARE ENUA:TNUA AT TARGET
		NEXT,	J/SETRES367A				!
 (6731)	DCS[1,00,1,0,0,0]  BM[1001,,00,11,,11,00,,000,,001,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,101,000]

	6050:	!(FREE)
	SETRES367A:
		P0,	BUMP-VERIFY,				!COUNT
		P3,	CSPD[16]_EMIT,				!CSP GETS
		EMITC,	SFNDMUX-4567-SEL,			!RES VALUES
			SR-NOP, GUARD-EN,			!FIRST LOAD COMPLEMENT OF THOSE AFTER BUTA(CLEAR-FLAGS)
		NEXT,	J/LOADRES367A				!

 (6050)	DCS[0,00,0,0,0,1]  BM[0011,,10,10,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0001,,,1,,0000,0,,,11,000,,,000,101,001]

	6051:	!(FREE)
	LOADRES367A:
		P0,	BUMP-VERIFY,				!COUNT
		P2-T,	RES_CSPB(B16),				!STORE RES
			D_ZERO, D[C]_ALU07,			!SET D, D[C]
```

```
            P1,     BUTA(CLP-FLAG-RES-UCON),                 ;RESET RES TO SR-LOAD, GUARD-DIS
            NEXT,   J/LOADSR367A                             ;
 (6051)  DCS[0,00,0,0,0,1]  BM[0011,,11,01,,00,00,,000,,011,,,0,1,0,,0,,0,,,0,0000,,,0,,1000,1,,,11,010,,,000,101,010]

    6052:  ;(FREE)
    LOADSR367A:
            P2-T,   SR_BSPHI(C052525),                       ;SR_(052525)
            NEXT,   J/GOBUT367B                              ;
 (6052)  DCS[0,00,0,0,0,0]  BM[1010,,01,11,,00,00,,111,,000,,,0,0,1,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,101,011]

    6053:  ;(FREE)
    GOBUT367B:
            SETUP,  RETURN/TEST370A,                         ;RETURN TO START OF NEXT SUBTEST
            NEXT,   GOTO-PAGE(7),                            ;BUT'S ARE ON PAGE 7
                    J/BUTGD3-2                               ;GO BUT ON GUARD <3:2>
 (6053)  DCS[0,00,0,0,0,0]  BM[0110,,00,11,,10,10,,101,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,001,100]

    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

    ;*** TEST 370A ***
    ;TEST-370A CHECKS THAT SHIFT LEFT/GUARD-ENABLED SHIFTS GD<3>="0" INTO SR<00>
    6725:
    TEST370A:
            P0,     LOAD-ENUA(ZTARGET434),                   ;SETUP FOR IR=(000000)/INSTRS TEST
                    LOAD-ERROR(TEST370A),                    ;ERROR DIRECTORY KEY
                    DCS-CTR(C10,),                           ;COMPARE ENUA:TNUA AT TARGET
            NEXT,   J/SETRES370A                             ;
 (6725)  DCS[1,00,1,0,0,0]  BM[0101,,00,11,,11,00,,011,,100,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,101,100]

    6054:  ;(FREE)
    SETRES370A:
            P0,     BUMP-VERIFY,                             ;COUNT
            P1,     CBPD[16]_EMIT,                           ;CBP GETS
            EMITC,  SENDMUX-4567-SEL,                        ;RES VALUES
                    SR-LEFT, GUARD-EN,                       ;
            NEXT,   J/LOADRES370A                            ;
 (6054)  DCS[0,00,0,0,0,1]  BM[0001,,10,10,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0001,,,1,,0000,0,,,11,000,,,000,101,101]

    6055:  ;(FREE)
    LOADRES370A:
            P2-T,   RES_CBPD(B16),                           ;STORE RES
                    D_ASPHI(C000000), D[C]_1,                ;SETUP D, D[C]
            NEXT,   J/SHIFT370A                              ;
 (6055)  DCS[0,00,0,0,0,0]  BM[1111,,11,01,,11,01,,100,,000,,,0,1,0,,0,,0,,,0,0000,,,0,,1000,1,,,11,000,,,000,101,110]

    6056:  ;(FREE)
    SHIFT370A:
            P2-T,   SR_SR-LEFT-1,                            ;SHIFT SR LEFT 1, SR<00>_GD<3>="0"
```



```
            NEXT,   J/COMP370A                               ;
 (6056)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,1,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,101,111]

    6057:  ;(FREE)
    COMP370A:
            P2-T,   D_SR-XOR-BSPHI(C125252), SAVE-D[C],      ;D _ (125252)-SR,  BITWISE
            NEXT,   J/GOBUT370A
 (6057)  DCS[0,00,0,0,0,0]  BM[0110,,01,11,,00,00,,110,,111,,,0,1,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,110,000]

    6060:  ;(FREE)
    GOBUT370A:
            SETUP,  RETURN/TEST370B,                         ;RETURN TO START OF NEXT SUBTEST
            NEXT,   CALL(DINTOIR-5]                          ;SUBR: D => IR, BUT(INSTRS)
 (6060)  DCS[0,00,0,0,0,0]  BM[0110,,00,11,,10,10,,011,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,111,011]

    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

    ;*** TEST 370B ***
    ;TEST-370B CHECKS THAT THE GUARD, NON ENABLED, WAS ALSO SHIFTED LEFT
    6723:
    TEST370B:
            P0,     LOAD-ENUA(ZTARGET402),                   ;GUARD IS NOW "10"="10"
                    LOAD-ERROR(TEST370B),                    ;ERROR DIRECTORY KEY
                    DCS-CTR(C3,),                            ;COMPARE ENUA:TNUA AT TARGET
            NEXT,   J/GOBUT370B                              ;
 (6723)  DCS[1,00,1,0,0,0]  BM[1100,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,110,001]

    6061:  ;(FREE)
    GOBUT370B:
            SETUP,  RETURN/TEST370C,                         ;RETURN TO START OF NEXT SUBTEST
            P0,     BUMP-VERIFY,                             ;COUNT
            NEXT,   GOTO-PAGE(7),                            ;BUT'S ARE ON PAGE 7
                    J/BUTGD3-2                               ;GO BUT ON GUARD <3:2>
 (6061)  DCS[0,00,0,0,0,1]  BM[0110,,00,11,,10,10,,001,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,001,100]

    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

    ;*** TEST 370C ***

    ;TEST-370C CHECKS THAT SHIFT LEFT/GUARD-ENABLED SHIFTS GD<3>="1" INTO SR<00>
    6721:
    TEST370C:
```

```
             P0,      LOAD-ENUA(ZTARGET434),                     ;SETUP FOR IR=(000000)/INSTRS TEST
                      LOAD-ERROR(TEST370C),                      ;ERROR DIRECTORY KEY
                      DCS-CTR(C9,),                              ;COMPARE ENUA:THUA AT TARGET
             NEXT,    J/SETDDC370C                               ;
   (6721)  DCS[1,00,1,0,0,0]  BM[0110,,00,11,,11,00,,011,,100,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,110,010]

       6062:  ;(FREE)
       SETDDC370C:
             P0,      BUMP-VERIFY,                               ;COUNT
             P2-T,    D_ZERO, D[C]_ALU07,                        ;SET D, D[C]
             NEXT,    J/SHIFT370C                                ;
   (6062)  DCS[0,00,0,0,0,1]  BM[0011,,00,00,,00,00,,000,,011,,,0,1,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,110,011]

       6063:  ;(FREE)
       SHIFT370C:
             P3-T,    SP_SR-LEFT-1,                              ;SHIFT SR LEFT, SR<00>_GD<3>="1"
             NEXT,    J/COMP370C                                 ;
   (6063)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,1,0,1,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,110,100]

       6064:  ;(FREE)
       COMP370C:
             P2-T,    D_SR+XOR-BSPHI(C052525), SAVE-D[C],        ;D _ (052525)+SR, BITWISE
             NEXT,    J/GOBUT370C                                ;
   (6064)  DCS[0,00,0,0,0,0]  BM[0110,,01,11,,00,00,,111,,111,,,0,1,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,110,101]

       6065:  ;(FREE)
       GOBUT370C:
             SETUP,   RETURN/TEST370D,                           ;RETURN TO START OF NEXT SUBTEST
             NEXT,    CALL[DINTOIR-5]                            ;SUBR: D -> IR, BUT(INSTRS)
   (6065)  DCS[0,00,0,0,0,0]  BM[0110,,00,11,,10,01,,111,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,111,011]







       ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

       ;*** TEST 370D ***
       ;TEST-370D CHECKS THAT THE "1" THAT GOT PUT IN GD<0> CAN BE SHIFTED BACK
       6717:
       TEST370D:
             P0,      LOAD-ENUA(ZTARGET401),                     ;GUARD IS NOW "01"?"00"
                      LOAD-ERROR(TEST370D),                      ;ERROR DIRECTORY KEY
                      DCS-CTR(C3,),                              ;COMPARE ENUA:THUA AT TARGET
             NEXT,    J/GOBUT370D                                ;
   (6717)  DCS[1,00,1,0,0,0]  BM[1100,,00,11,,11,00,,000,,001,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,110,110]

       6066:  ;(FREE)
       GOBUT370D:
             SETUP,   RETURN/TEST371A,                           ;RETURN TO START OF NEXT SUBTEST
             NEXT,    GOTO-PAGE(7),                              ;BUT'S ARE ON PAGE 7
```



```
                      J/BUTGD3-2                                 ;GO BUT ON GUARD <3:2>
   (6066)  DCS[0,00,0,0,0,0]  BM[0110,,00,11,,10,01,,101,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,001,100]







       ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

       ;*** TEST 371A ***
       ;TEST-371A CHECKS THAT SR-NOP FUNCTION DOES NOTHING
       6715:
       TEST371A:
             P0,      LOAD-ENUA(ZTARGET434),                     ;SETUP FOR IR=(000000)/INSTRS TEST
                      LOAD-ERROR(TEST371A),                      ;ERROR DIRECTORY KEY
                      DCS-CTR(C10,),                             ;COMPARE ENUA:THUA AT TARGET
             NEXT,    J/SETRES371A                               ;
   (6715)  DCS[1,00,1,0,0,0]  BM[0101,,00,11,,11,00,,011,,100,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,110,111]

       6067:  ;(FREE)
       SETRES371A:
             P0,      BUMP-VERIFY,                               ;COUNT
             P3,      CSPD[16]_EMIT,                             ;CSP GETS
             EMITC,   SENDMUX-4567-SEL,                          ;REG VALUES
                      SR-NOP, GUARD-EN,                          ;
             NEXT,    J/LOADRES371A                              ;
   (6067)  DCS[0,00,0,0,0,1]  BM[0011,,10,10,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0001,,,1,,0000,0,,,11,000,,,000,111,000]

       6070:  ;(FREE)
       LOADRES371A:
             P2-T,    REG_CSPB(B16),                             ;STORE REG
                      D_BSPHI(C000000), D[C]_1,                  ;SETUP D, D[C]
             NEXT,    J/SHIFT371A                                ;
   (6070)  DCS[0,00,0,0,0,0]  BM[1111,,11,01,,11,01,,100,,000,,,0,1,0,,0,,0,,,0,0000,,,0,,1000,1,,,11,000,,,000,111,001]

       6071:  ;(FREE)
       SHIFT371A:
             P0,      BUMP-VERIFY,                               ;COUNT
             P2-T,    CLK-SR,                                    ;DO AN SR-NOP
             NEXT,    J/COMP371A                                 ;
   (6071)  DCS[0,00,0,0,0,1]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,1,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,111,010]

       6072:  ;(FREE)
       COMP371A:

             P2-T,    D_SR+XOR-BSPHI(C052525), SAVE-D[C],        ;D _ (052525)+SR, BITWISE
             NEXT,    J/GOBUT371A                                ;
   (6072)  DCS[0,00,0,0,0,0]  BM[0110,,01,11,,00,00,,111,,111,,,0,1,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,111,011]

       6073:  ;(FREE)
       GOBUT371A:
             SETUP,   RETURN/TEST371B,                           ;RETURN TO START OF NEXT SUBTEST
```

```
        P0,     BUMP-VERIFY,                            ;COUNT
        NEXT,   CALL(DINTOIR-8)                         ;SUBR: D => IR, BUT(INSTR5)
(6073)  DCS[0,00,0,0,0,1]  BM[0110,,00,11,,10,01,,011,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,111,011]




;- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

;*** TEST 371B ***
;TEST-371B CHECKS THAT THE GUARD WASN'T ALTERED EITHER
6713:
TEST371B:
        P0,     LOAD-ENUA(ETARGET401),                  ;GUARD IS STILL "01"*"00"
                LOAD-ERROR(TEST371B),                   ;ERROR DIRECTORY KEY
                DCS-CTR(C3,),                           ;COMPARE ENUA:THUA AT TARGET
        NEXT,   J/GOBUT371B                             ;
(6713)  DCS[1,00,1,0,0,0]  BM[1100,,00,11,,11,00,,000,,001,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,111,100]

6074:   ;(FREE)
GOBUT371B:
        SETUP,  RETURN/SCOPE371,                        ;RETURN TO SCOPE LOOP TEST WORD
        NEXT,   GOTO-PAGE(7),                           ;BUT'S ARE ON PAGE 7
                J/BUTGD3-2                              ;GO BUT ON GUARD <3:2>
(6074)  DCS[0,00,0,0,0,0]  BM[0110,,00,00,,01,11,,101,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,001,100]


;SCOPE LOOP TEST FOR SR, GUARD, XNUX, RES AREA CODE
6075:   ;(FREE)
SCOPE371:
        P0,     BUMP-VERIFY,                            ;COUNT
        P2,     RES_CSPD(C000000),                      ;RESET RES TO SR-LOAD/GUARD-DIS
        NEXT,   PUTD[SCOPE],                            ;NO ERROR:  "TEST372A" [+3, WORDS]
                J/TEST372A                              ;   ERROR:  "SETRES361A" [-$S, WORDS]
(6075)  DCS[0,00,0,1,0,1]  BM[0000,,10,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0100,,,0,,1000,1,,,11,000,,,101,111,001]




;  - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

;       THE LLOWING TWO SUBROUTINES ARE ALSO USED IN THE ABOVE TESTS:


7031:   ;(FREE)
CSP16XORSRTOIR5:
        P2-T,   D_SR-XOR-CSPB(816), SAVE-D[C],          ;COMPARE SR-XNUX:EXPECTED VALUE, BITWISE
        NEXT,   J/DINTOIR5                              ;  AND PUT IN IR TO DO INSTR5 TEST
(7031)  DCS[0,00,0,0,0,0]  BM[0110,,11,01,,00,00,,000,,111,,,0,1,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,010,111,011]
```



```
7035:   ;(FREE)
CSP16XORFLTTOIR5:
        P2-T,   D_FLTPT-XOR-CSPB(816), SAVE-D[C],       ;COMPARE FLTPT-XMUX:EXPECTED VALUE, BITWISE
        NEXT,   J/DINTOIR5                              ;  AND PUT IN IR TO DO INSTR5 TEST
(7035)  DCS[0,00,0,0,0,0]  BM[0110,,11,01,,00,01,,000,,111,,,0,1,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,010,111,011]




;.PAGE==============================================================================================

.TOC    *    TEST372A-372B:  TESTING CUA, PROCESSOR MUX, AND BUTA(SUBR A)

;*************************************************************************************
;*
;*      TESTS:  372A - 372B                             UWORDS: 015 + 002
;*
;*      FUNCTIONS:      TESTS THAT CUA IS LOADED, AND CAN BE READ THRU PROCESSOR MUX,
;*                      ALSO TESTS THAT CAN BE PUT INTO D, AND BUT(SUBR A) LOADS
;*                      D<14:03> INTO RETURN,
;*
;*************************************************************************************




;- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

;*** TEST 372A ***
;TEST-372A CHECKS THE CUA => D => SUBR A => RETURN PATH WITH PATTERN [6222]
6571:
TEST372A:
        P0,     LOAD-ENUA(CUA372A),                     ;WHERE WE BUT [RETURN] TO
                LOAD-ERROR(TEST372A),                   ;ERROR DIRECTORY KEY
                DCS-CTR(C7,),                           ;IN 7, MICROWORDS
        NEXT,   J/LOOP372A                              ;
(6571)  DCS[1,00,1,0,0,0]  BM[1000,,00,11,,00,10,,010,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,101,100,000]

6540:
LOOP372A:
        SELECT, UCON-PROC,                              ;SELECT PROCESSOR UCON CONTROL

        ENABLE, BUSDIN_CUA[14-03],                      ;  PUT 0#CUA[11-00]#EXFLAG<2:1>#FOVP ON BUSDIN
        P0,     SET-UCON-CONTROL,                       ;LOAD UCON REGISTER AT P0
                BUMP-VERIFY,                            ;COUNT
        P3,     BUTA(CUA-TRACK),                        ;RESET TRACKING OF CUA
        NEXT,   J/SETDC372A                             ;
(6540)  DCS[0,00,0,0,0,1]  BM[0000,,00,00,,01,01,,000,,000,,,0,0,0,,0,,0,,,1,1001,,,0,,0000,0,,,11,001,,,000,111,110]

6076:   ;(FREE)
```

```
	SETDC372A:
		P2-T,	D_ASPHI(C000000), D[C]_1,		;SET D[C] FLAG = (1) FOR FIRST LOOP
		NEXT,	J/CUA372A				;
(6076)	DCS[0,00,0,0,0,0]  BM[1111,,00,00,,11,01,,100,,000,,,0,1,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,010,010,010]

	6222:
	CUA372A:
		P3,	CSPD[16]_BUSDIN, RETURN/BUTERRORS,	;COPY CUA [WHICH IS ADDRESS OF THIS UWORD] INTO CSP
		NEXT,	BUTP(D[C]-B),				;IF D[C] SET [PASS 1], GO TO "LOAD372A"
			J/TEST372B				;IF D[C] RESET [PASS 2], GO TO "TEST372B"
(6222)	DCS[0,00,0,0,0,0]  BM[0110,,10,01,,11,11,,110,,000,,,0,0,0,,0,,0,,,0,0001,,,1,,0000,0,,,10,011,,,111,000,001]

	6703:
	LOAD372A:
		P0,	BUMP-VERIFY,				;COUNT
		P2-T,	D_CSPB(B16), D[C]_0,			;PUT CUA FROM CSP INTO D, RESET D[C]
		NEXT,	J/SUBRA372A				;
(6703)	DCS[0,00,0,0,0,1]  BM[1010,,11,01,,00,00,,000,,000,,,0,1,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,111,111]

	6077:	;(FREE)
	SUBRA372A:
		SETUP,	RETURN/BUTERRORS,			;IF BUTA(SUBR-B) USED INSTEAD
		NEXT,	PAGE(7),				;SUBROUTINE IS ON PAGE 7
			BUTA(SUBR-A),				;LOAD PAGE, LOAD RETURN FROM D
			J/ZTARGET555				;THE SUBROUTINE
(6077)	DCS[0,00,0,0,0,0]  BM[0110,,00,01,,11,11,,110,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,101,,,101,101,101]

	;NEXT WORD COMES FROM "ZTARGET555", WHICH DOES ONLY A BUTA(RETURN),
	;WHICH SHOULD RETURN TO "CUA372A" [-3, WORDS]






	; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

	;*** TEST 372B ***
	;TEST-372B CHECKS THE CUA -> D -> SUBR A -> RETURN PATH WITH PATTERN [5555]
	6701:
	TEST372B:
		P0,	LOAD-ENUA(CUA372B),			;WHERE WE BUT[RETURN] TO
			LOAD-ERROR(TEST372B),			;ERROR DIRECTORY KEY
			DCS-CTR(C7.),				;IN 7, MICROWORDS
		NEXT,	J/PAGE372B				;
(6701)	DCS[1,00,1,0,0,0]  BM[1000,,00,10,,11,01,,101,,101,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,000,000]

	6100:	;(FREE)
	PAGE372B:
		SETUP,	RETURN/BUTERRORS,			;JUNK IN RETURN REGISTER
		NEXT,	GOTO-PAGE(5),				;XFER TO PAGE 5
			J/SETDC372B				;
```



```
(6100)	DCS[0,00,0,0,0,0]  BM[0101,,00,01,,11,11,,110,,101,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,000,000,000]

	5000:	;(FREE)
	SETDC372B:
		P0,	BUMP-VERIFY,				;COUNT
		P2-T,	D_ASPHI(C000000), D[C]_1,		;SET D[C] FLAG = (1) FOR FIRST LOOP
		P1,	BUTA(CUA-TRACK),			;RESET TRACKING OF CUA
		NEXT,	J/CUA372B				;
(5000)	DCS[0,00,0,0,0,1]  BM[1111,,00,00,,11,01,,100,,000,,,0,1,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,001,,,101,101,101]

	5555:
	CUA372B:
		P3,	CSPD[16]_BUSDIN, RETURN/BUTERRORS,	;COPY CUA [WHICH IS ADDRESS OF THIS WORD] INTO CSP
		NEXT,	BUTR(D[C]-B),				;IF D[C] SET [PASS 1], GO TO "LOAD372B"
			J/SCOPE372B				;IF D[C] RESET [PASS2], GO TO "SCOPE372B"
(5555)	DCS[0,00,0,0,0,0]  BM[0101,,10,01,,11,11,,110,,000,,,0,0,0,,0,,0,,,0,0001,,,1,,0000,0,,,10,011,,,101,000,001]

	5503:
	LOAD372B:
		P2-T,	D_CSPB(B16), D[C]_0,			;PUT CUA FROM CSP INTO D, RESET D[C]
		NEXT,	J/SUBRA372B				;
(5503)	DCS[0,00,0,0,0,0]  BM[1010,,11,01,,00,00,,000,,000,,,0,1,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,010,010,011]

	5223:	;(FREE)
	SUBRA372B:
		P0,	BUMP-VERIFY,				;COUNT
		SETUP,	RETURN/BUTERRORS,			;IF BUTA(SUBR-B) USED INSTEAD
		NEXT,	PAGE(7),				;SUBROUTINE IS ON PAGE 7
			BUTA(SUBR-A),				;LOAD PAGE, LOAD RETURN FROM D
			J/RESETUCONP				;THE SUBR:  RESETS PROC UCON:  BUSDIN_EMIT, EN-CLK-IR
(5223)	DCS[0,00,0,0,0,1]  BM[0101,,00,01,,11,11,,110,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,101,,,010,111,001]

	;NEXT WORD COMES FROM "RESETUCONP", WHICH DOES ONLY A BUTA(RETURN),
	;WHICH SHOULD RETURN TO "CUA372B" [-3,WORDS]

	5501:
	SCOPE372B:
		NEXT,	GOTO-PAGE(6),				;XFER
			BUTD[SCOPE],				;NO ERROR:  "TEST373A" [+1,WORDS]
			J/TEST373A				;   ERROR:  "LOOP372A" [-12,WORDS]
(5501)	DCS[0,00,0,1,0,0]  BM[0000,,00,00,,00,00,,000,,110,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,101,100,001]





	;.PAGE==========================================================================================


	.TOC	*	TEST373:  CHECK JANUPP W/ BUTA(DIAGNOSE), BM EXT BIT FLPADR
```

```
9715    !*************************************************************************************
9716    !*
9717    !*      TEST8:  373 A = B                               UWORD8:  007 + 014
9718    !*
9719    !*      FUNCTIONS:
9720    !*
9721    !*      THE FOLLOWING SET OF TWO TESTS PERFORMS SEVERAL FUNCTIONS:
9722    !*
9723    !*              TEST-373-A CHECKS THAT CONTROL CAN BE PASSED TO THE BASE MACHINE, VIA
9724    !*              BUTA(DIAGNOSE), AND SEVERAL B.M. WORDS EXECUTED.  A BUTA(DIAGNOSE) IN
9725    !*              THE B.M. SHOULD THEN BE ENCOUNTERED, RETURNING CONTROL TO THE DCS VIA
9726    !*              A JAMUPP FORCE.
9727    !*
9728    !*              TEST-373-B THEN CHECKS THAT THE B.M. MACHINE CODE CORRECTLY
9729    !*              ASSERTED THE "FLPADR-L" EXTENSION BIT, FORCING A READ, VIA THE ASP/DF
9730    !*              FIELD MODE, OF A SCRATCHPAD.
9731    !*
9732    !*************************************************************************************
9733
9734
9735
9736    !THIS TEST GOES TO THE B.M. VIA BUTA(DIAGNOSE)
9737    6541:
9738    TEST373A:
9739            P0,     LOAD-ENUA(4777),                        !JAMUPP8 GO HERE IN DCS
9740                    LOAD-ERROR(TEST373A),                   !ERROR DIRECTORY KEY
9741                    DCS-CTR(C7,),                           !COMPARE AT JAMUPP
9742            NEXT,   J/RETURN373A                            !
     (6541)  DCS[1,00,1,0,0,0]  BM[1000,,00,10,,01,11,,111,,111...0,0,0,,0,,0,,0...0,0000...0,,0000,0,,,11,000,,,101,110,010]
9743
9744    6562:
9745    RETURN373A:
9746            P3,     CSPD[00]_EMIT, RETURN/TEST373B,         !RETURN AFTER PROCESSING, TO NEXT TEST
9747            NEXT,   GOTO-PAGE(7),                           !JAMUPP ROUTINE EXPECTS RETURN ADDRESS IN CSP(00)
9748                    J/SETIN373A                             !
     (6562)  DCS[0,00,0,0,0,0]  BM[0110,,10,10,,11,11,,010,,111...0,0,0,,0,,0,,0...0,1111...1,,0000,0,,,11,100,,,000,011,100]
9749
9750    7034:  ;(FREE)
9751    SETIN373A:
9752            P2=0,   IR_EMIT, EMIT/000002,                   !SET DF8(2) IN IR
9753            P3,     BUTA(DIAGNOSE),                         !BEGIN THE TRANSFER SEQUENCE
9754            NEXT,   J/SETSR373A                             !
     (7034)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,010...0,0,0,,0,,0,,0...1,1010...0,,0000,0,,,11,011,,,000,011,111]
9755
9756    7037:  ;(FREE)
9757    SETSR373A:
9758            P2=T,   SR_CSPD(C052525),                       !SET SR<00>=(1) FOR JAMUPP EXPECTED (BY JAMUPP SERVICER)
9759            NEXT,   GOTO-PAGE(3),                           !SET PAGE TO BE ACTUAL (3)
9760                    J/MED25                                 !POINT UPP TO B.M. DISP, ON PAGE
     (7037)  DCS[0,00,0,0,0,0]  BM[1010,,10,00,,00,00,,000,,011...0,0,1,,0,,0,,0...0,0111...0,,0000,0,,,11,100,,,000,010,000]
9761
9762    !THE SEQUENCE OF CONTROL SHOULD NOW BE COMING FROM THE B.M.!
```



```
9763
9764    !3070!
9765    !MED25!
9766    !       NEXT,   GOTO-PAGE(2),                           !XFER
9767    !               J/MED25A                                !
9768    !
9769    !2071!
9770    !MED25A!
9771    !       P2=T,   D_ASPHI(DF)-TOP, SAVE-D(C),             !READ FROM ASPHI/DF, WITH FLPADR ASSERTED
9772    !                                                       !SINCE DF=(2), THE SPADDR8=(12) WILL BE FORCED
9773    !       P3,     BUTA(DIAGNOSE),                         !ALSO START TO RETURN XFER TO DCS, VIA JAMUPP
9774    !       NEXT,   J/MED19A                                !
9775    !
9776    !2064!
9777    !MED19A!
9778    !       NEXT,   GOTO-PAGE(3),                           !THIS WORD ALSO GETS EXECUTED
9779    !               J/MED19                                 !BUT NOW JAMUPP GETS FORCED, AND WE GO NO FURTHER
9780    !
9781    !*** CONTROL NOW COMES BACK TO DCS JAMUPP POINT ***
9782    !
9783    !4777!
9784    !JAMUPP00!:
9785    !       P3=T,   SR_0,                                   !SAVE OLD D IN SR
9786    !       NEXT,   BUTR(SR00),                             !TEST SR<00>:
9787    !                                                       !IF=(0), NOT-EXPECTED JAM, GOTO(JAMUPP003), (SEE JAMUPP ROUTINE)
9788    !               J/JAMUPP003                             !IF=(1), EXPECTED JAM, GOTO(JAMUPP002B)
9789    !
9790    !* COME HERE FOR EXPECTED JAM *
9791    !4757!
9792    !JAMUPP002B:
9793    !       P2=T,   D_CSPD(000), SAVE-D(C),                 !GET RETURN ADDRESS, STORED IN CSP(00)
9794    !       NEXT,   J/JAMUPP002C                            !
9795    !
9796    !=[4000!4777]
9797    !JAMUPP002C:
9798    !       P0,     RETURN_D[14-03], PAGE(7),               !PUT RETURN ADDRESS INTO RETURN REGISTER
9799    !       P2=T,   D_SR, SAVE-D(C),                        !RESTORE OLD D
9800    !       NEXT,   J/JAMUPP002D                            !
9801    !
9802    !=[7000!7377]
9803    !JAMUPP002D:
9804    !       P2=T,   SR_ZERO,                                !ZERO OUT SR<00>, FURTHER JAMUPP8 NOW ILLEGAL
9805    !       NEXT,   BUTA(RETURN),                           !AND RETURN TO FLOW8
9806    !               J/BUTERROR?                             !ONLY IF ERROR
9807    !
9808

9809    !AT THIS POINT, CONTROL SHOULD NOW RETURN TO THIS POINT; THE NEXT TEST
9810    !THIS TEST NOW CHECKS TO SEE THAT THE B.M. FUNCTION WAS EXECUTED CORRECTLY
9811    !NOTE THAT IN TEST-350 ( A WHILE BACK ), ASB6PHI[12] WERE LOADED WITH (000152) DATA
9812
9813    4572:
9814    TEST373B:
9815            P0,     LOAD-ENUA(ETARGET425),                  !FOR INSTR8-E78-(425) DECODE
9816                    LOAD-ERROR(TEST373B),                   !ERROR DIRECTORY KEY
```

```
9817                DC8-CTR(C6,),                          ;COMPARE AT TARGET
9818          NEXT,  J/GOBUT373B                           !
     (6572)  DC8(1,00,1,0,0,0)  BM(1001,,00,11,,11,00,,010,,101,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,000,001)
9819
9820     6101:  ;(FREE)
9821     GOBUT373B:
9822          SETUP,  RETURN/SCOPE373,                     ;RETURN TO SCOPE LOOP TEST WORD
9823          NEXT,   CALL(DINTCIR=5)                      ;GO DO D -> DBUF -> IR, BUT(INSTRS)
     (6101)  DC8(0,00,0,0,0,0)  BM(0110,,00,00,,10,00,,010,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,111,011)
9824
9825
9826
9827     6102:  ;(FREE)
9828     SCOPE373:
9829          P0,     PUMP-VERIFY,                         ;COUNT
9830          NEXT,   BUTD(SCOPE),                         ;NO ERROR:  "TEST374"  (+1, WORDS)
9831                  J/TEST374                            ;   ERROR:  "RETURN373A"  (-12, WORDS)
     (6102)  DC8(0,00,0,1,0,1)  BM(0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,101,110,011)
9832
9833
9834
9835
9836
9837     1,PAGE*************************************************************************************************************
9838
9839
9840     .TOC    *    TEST374:  A/B BP REWRITE MODES VERIFICATION
9841
9842
9843     !*************************************************************************************
9844     !*
9845     !*      TEST8:  374 A - F                                UWORDS:  078 + 336
9846     !*
9847     !*      FUNCTIONS:
9848     !*
9849     !*      THE FOLLOWING GROUP OF TWELVE TESTS PERFORMS A VERIFICATION OF THE A/B BP REWRITE
9850     !*      FUNCTIONS:
9851     !*
9852     !*              WR(A,L,A)       WR(A,L,B)       WR(A,H,A)       WR(A,H,B)
9853     !*              WR(B,L,A)       WR(B,L,B)       WR(B,H,A)       WR(B,H,B)
9854     !*              WR(AB,L,A)      WR(AB,L,B)      WR(AB,H,A)      WR(AB,H,B)
9855     !*
9856     !*      EACH FUNCTION IS INVOKED, AND THE RESULTANT SCRATCHPADS ARE CHECKED TO INSURE
9857     !*      THAT (1) ONLY THE RIGHT SCRATCHPADS WERE WRITTEN, AND (2) THE CORRECT ADDRESS
9858     !*      WAS USED FOR THE REWRITE.
9859     !*
9860     !*************************************************************************************
9861
9862     !
9863     !       SUMMARY OF ASP/BSP HI/LO REWRITE FUNCTIONALITY TESTS:
9864     !
9865     !                                                --DF---  --SF---
9866     !       TEST  REWRITE-->FUNCTION   ASP   BSP    ASP BSP  ASP BSP  SR IN
```



```
9867     !       NUMB  A/B HI/LO A/B-ADR   ADDR  ADDR   H-L H-L  H-L H-L  OCTAL
9868     !       ----  --- ----- -------   ----  ----   - - - -  - - - -  ------
9869     !
9870     !        A1    A    LO   A=ADDR   SF/4  DF/2   0 0 0 0  0 1 0 0  002100
9871     !        A2    "    ""   B=ADDR    "     "     0 1 0 0  0 0 0 0
9872     !
9873     !        B1    A    HI   A=ADDR   DF/2  SF/4   1 0 0 0  0 0 0 0  100010
9874     !        B2    "    ""   B=ADDR    "     "     0 0 0 0  1 0 0 0
9875     !
9876     !        C1    B    LO   A=ADDR   DF/4  SF/2   0 0 0 1  0 0 0 0  010001
9877     !        C2    "    ""   B=ADDR    "     "     0 0 0 0  0 0 0 1
9878     !
9879     !        D1    B    HI   A=ADDR   SF/2  DF/4   0 0 0 0  0 0 1 0  001040
9880     !        D2    "    ""   B=ADDR    "     "     0 0 1 0  0 0 0 0
9881     !
9882     !        E1    AB   LO   A=ADDR   SF/4  DF/2   0 0 0 0  0 1 0 1  002520
9883     !        E2    ""   ""   B=ADDR    "     "     0 1 0 1  0 0 0 0
9884     !
9885     !        F1    AB   HI   A=ADDR   DF/4  SF/2   1 0 1 0  0 0 0 0  120012
9886     !        F2    ""   ""   B=ADDR    "     "     0 0 0 0  1 0 1 0
9887     !
9888
9889
9890
9891
9892
9893     ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
9894
9895     !*** TEST 374A ***
9896
9897     65631
9898     TEST374A:
9899          P0,     BUMP-VERIFY,                         ;COUNT
9900                  LOAD-ERROR(TEST374),                 ;ERROR DIRECTORY KEY
9901          P3,     CSPD[01]_EMIT,                       ;CONSTANT FOR:
9902          EMITC,  SP-LEFT, GUARD-DIS,                  ;SETUP REG FOR SR FUNCTION
9903                  EMIT07/1,                            ;FLAG BIT FOR TESTING
9904          NEXT,   J/LOADRES3T4                         !
     (6563)  DC8(1,00,0,0,0,1)  BM(0001,,10,00,,00,10,,000,,000,,,0,0,0,,0,,0,,,0,1110,,,1,,0000,0,,,11,000,,,001,000,011)
9905
9906     6103:  ;(FREE)
9907     LOADRES374:
9908          P0,     DC8-CTR(C15,),                       ;HOLD UP ERROR COUNTER
9909          P2,     RES_CSPD(001),                       ;SP NOW SETUP
9910          NEXT,   GOTO-PAGE(5),                        ;XFR
9911                  J/EXPEC374A1                         !
     (6103)  DC8(0,00,1,0,0,0)  BM(0000,,10,00,,00,00,,000,,101,,,0,0,0,,0,,0,,,0,1110,,,0,,1000,1,,,11,100,,,010,010,010)
9912
9913     5222:  ;(FREE)
9914     EXPEC374A1:
9915          P3,     CSPD[02]_EMIT, EMIT/002100,          ;EXPECTED "SERIAL" REPRESENTATION OF RESULT
9916          NEXT,   J/ZERO374A1                          !
     (5222)  DC8(0,00,0,0,0,0)  BM(0000,,10,01,,00,01,,000,,000,,,0,0,0,,0,,0,,,0,1101,,,1,,0000,0,,,11,000,,,101,011,110)
```

```
 9917
 9918    5535:
 9919    ZERO374A1:
 9920            P0,     BUMP-VERIFY,                    ;COUNT
 9921            SETUP,  RETURN/DOWRITE374A1,            ;EXEC SUBR WHICH:
 9922                                                    ;(1) (000402) -> IR,
 9923            NEXT,   CALL[ZERO&F04DF02]              ;(2)  WRITES ZEROES TO A/B SP MI/LO SF/DF
    (5536)  DC&[0,00,0,0,0,1]  BM[0101,,00,01,,00,10,,101,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,000,100,001]
 9924
 9925    5225:  ;(FREE)
 9926    DOWRITE374A1:
 9927            P2=T,   D_CSPD(D01), D[C]=0,            ;DATA WITH BIT<07> SET
 9928                    ASP=ADDR&=R[SF],                ;ADDRESS ASP WITH SF MODE
 9929                    BSP=ADDR&=R[DF],                ;ADDRESS BSP WITH DF MODE
 9930            P3,     WR(A,LO,A=ADDR),                ;SELECT THE PARTICULAR FUNCTION TO TEST,
 9931            NEXT,   J/GETTEM374A1                   ;  USING A=ADDR FOR REWRITE
    (5225)  DC&[0,00,0,0,0,0]  BM[1010,,10,00,,00,11,,000,,000,,,0,1,0,,0,,0,,,0,1110,,,0,,0001,0,,,11,000,,,010,010,110]
 9932
 9933    5226:  ;(FREE)
 9934    GETTEM374A1:
 9935            P3,     CSPD[00]_EMIT, RETURN/ZERO374A2,        ;(SEE DESCRIP OF SUBR FOR FUNCTION)
 9936            NEXT,   CALL[SFDFTDBR]                  ;
    (5226)  DC&[0,00,0,0,0,0]  BM[0101,,10,01,,00,10,,111,,101,,,0,0,0,,0,,0,,,0,1111,,,1,,0000,0,,,11,100,,,111,100,110]
 9937
 9938
 9939
 9940    5227:  ;(FREE)
 9941    ZERO374A2:
 9942            SETUP,  RETURN/DOWRITE374A2,            ;AGAIN GO WRITE ZEROES TO A/B-SP-MI/LO
 9943            NEXT,   CALL[ZERO&F04DF02]              ;
    (5227)  DC&[0,00,0,0,0,0]  BM[0101,,00,01,,00,11,,000,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,000,100,001]
 9944
 9945    5230:  ;(FREE)
 9946    DOWRITE374A2:
 9947            P2=T,   D_CSPD(D01), D[C]=0,            ;DATA WITH BIT<07> SET
 9948                    ASP=ADDR&=R[SF],                ;ADDRESS ASP WITH SF MODE
 9949                    BSP=ADDR&=R[DF],                ;ADDRESS BSP WITH DF MODE
 9950            P3,     WR(A,LO,B=ADDR),                ;USE SAME FUNCTION AS ABOVE,
 9951            NEXT,   J/GETTE%374A2                   ;  ONLY USE B=ADDR FOR REWRITE THIS TIME
    (5230)  DC&[0,00,0,0,0,0]  BM[1010,,10,00,,00,11,,000,,000,,,0,1,0,,0,,0,,,0,1110,,,0,,0101,0,,,11,000,,,010,011,001]
 9952
 9953    5231:  ;(FREE)
 9954    GETTEM374A2:
 9955            P3,     CSPD[00]_EMIT, RETURN/TEST374A2,        ;(SEE DESCRIP OF SUBR FOR FUNCTION)
 9956            NEXT,   CALL[SFDFTDBR]                  ;
    (5231)  DC&[0,00,0,0,0,0]  BM[0101,,10,11,,00,00,,100,,101,,,0,0,0,,0,,0,,,0,1111,,,1,,0000,0,,,11,100,,,111,100,110]
 9957
 9958    56044
 9959    TEST374A2:
 9960            P0,     LOAD-TMUA(TTARGET434),          ;NOW SETUP FOR IR=ZERO COMPARE
 9961                    LOAD-ERROR(TEST374A2),          ;ERROR DIRECTORY KEY
 9962                    DC&-CTR(C&,),                   ;COMPARE AT TARGET
```



```
 9963                    BUMP-VERIFY,                    ;COUNT
 9964            NEXT,   J/GOTEST374A2                   ;
    (5604)  DC&[1,00,1,0,0,1]  BM[1001,,00,11,,11,00,,011,,100,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,010,011,010]
 9965
 9966    5232:  ;(FREE)
 9967    GOTEST374A2:
 9968            SETUP,  RETURN/SCOPE374A,               ;GO EXEC SUBR THAT:
 9969            NEXT,   CALL[DI%TDIR=5]                 ;  PUTS 0 -> IR, SUT(INSTR%) TO TEST FOR ZERO
    (5232)  DC&[0,00,0,0,0,0]  BM[0101,,00,01,,00,11,,011,,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,111,011]
 9970
 9971
 9972
 9973
 9974    5233:  ;(FREE)
 9975    SCOPE374A1:
 9976            NEXT,   BUTD[SCOPE],                    ;NO ERROR:  "EXPEC374B1" (+1, WORD&)
 9977                    J/EXPEC374B1                    ;   ERROR:  "ZERO374A1" (-8, WORD&)
    (5233)  DC&[0,00,0,1,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,101,011,111]
 9978
 9979
 9980
 9981
 9982    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
 9983
 9984    ;*** TEST 374B ***
 9985
 9986    5537:
 9987    EXPEC374B1:
 9988            P3,     CSPD[02]_EMIT, EMIT/100010,     ;EXPECTED "SERIAL" REPRESENTATION OF RESULT
 9989            NEXT,   J/ZERO374B1                     ;
    (5537)  DC&[0,00,0,0,0,0]  BM[1000,,10,00,,00,00,,001,,000,,,0,0,0,,0,,0,,,0,1101,,,1,,0000,0,,,11,000,,,101,101,110]
 9990
 9991    5556:
 9992    ZERO374B1:
 9993            P0,     BUMP-VERIFY,                    ;COUNT
 9994            SETUP,  RETURN/DOWRITE374B1,            ;EXEC SUBR WHICH:
 9995                                                    ;(1) (000402) -> IR,
 9996            NEXT,   CALL[ZERO&F04DF02]              ;(2)  WRITES ZEROES TO A/B SP MI/LO SF/DF
    (5556)  DC&[0,00,0,0,0,1]  BM[0101,,00,01,,00,11,,100,,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,000,100,001]
 9997
 9998    5234:  ;(FREE)
 9999    DOWRITE374B1:
10000            P2=T,   D_CSPD(D01), D[C]=0,            ;DATA WITH BIT<07> SET
10001                    ASP=ADDR&=R[DF],                ;ADDRESS ASP WITH DF MODE
10002                    BSP=ADDR&=R[SF],                ;ADDRESS BSP WITH SF MODE
10003            P3,     WR(A,MI,A-ADDR),                ;SELECT THE PARTICULAR FUNCTION TO TEST,

10004            NEXT,   J/GETTEM374B1                   ;  USING A-ADDR FOR REWRITE
    (5234)  DC&[0,00,0,0,0,0]  BM[1010,,10,01,,00,10,,000,,000,,,0,1,0,,0,,0,,,0,1110,,,0,,1001,0,,,11,000,,,010,011,101]
10005
10006    5235:  ;(FREE)
10007    GETTEM374B1:
10008            P3,     CSPD[00]_EMIT, RETURN/ZERO374B2,        ;(SEE DESCRIP OF SUBR FOR FUNCTION)
```

```
                NEXT,   CALL[SFDFTOSR]                          ;
      (5235)  DCS[0,00,0,0,0,0]  BM[0101,,10,01,,00,11,,110,,101,,,0,0,0,,0,,0,,,0,1111,,,1,,0000,0,,,11,100,,,111,100,110]



        5236:  ;(FREE)
        ZERO374B2:
                SETUP,  RETURN/DOWRITE374B2,            ;AGAIN GO WRITE ZEROES TO A/B-SP-HI/LO
                NEXT,   CALL[ZEROSF040F02]              ;
      (5236)  DCS[0,00,0,0,0,0]  BM[0101,,00,01,,00,11,,111,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,000,100,001]

        5237:  ;(FREE)
        DOWRITE374B2:
                P2-T,   D_CSPD(D01), D[C]_0,            ;DATA WITH BIT<07> SET
                        ASP-ADDRS-R[DF],                ;ADDRESS ASP WITH DF MODE
                        BSP-ADDRS-R[SF],                ;ADDRESS BSP WITH SF MODE
                P3,     WR(A,HI,B-ADDR),                ;USE SAME FUNCTION AS ABOVE,
                NEXT,   J/GETTEM374B2                   ;  ONLY USE B-ADDR FOR REWRITE THIS TIME
      (5237)  DCS[0,00,0,0,0,0]  BM[1010,,10,01,,00,10,,000,,000,,,0,1,0,,0,,0,,,0,1110,,,0,,1101,0,,,11,000,,,010,100,000]

        5240:  ;(FREE)
        GETTEM374B2:
                P3,     CSPD[00]_EMIT, RETURN/TEST374B2,        ;(SEE DESCRIP OF SUBR FOR FUNCTION)
                NEXT,   CALL[SFDFTOSR]                  ;
      (5240)  DCS[0,00,0,0,0,0]  BM[0101,,10,11,,00,00,,110,,101,,,0,0,0,,0,,0,,,0,1111,,,1,,0000,0,,,11,100,,,111,100,110]

        5605:
        TEST374B2:
                P0,     LOAD-ENUA(ZTARGET434),          ;NOW SETUP FOR IR=ZERO COMPARE
                        LOAD-ERROR(TEST374B2),          ;ERROR DIRECTORY KEY
                        DCS-CTR[C6,],                   ;COMPARE AT TARGET
                        BUMP-VERIFY,                    ;COUNT
                NEXT,   J/GOTEST374B2                   ;
      (5605)  DCS[1,00,1,0,0,1]  BM[1001,,00,11,,11,00,,011,,100,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,010,100,001]

        5241:  ;(FREE)
        GOTEST374B2:
                SETUP,  RETURN/SCOPE374B,               ;GO EXEC SUBR THAT:
                NEXT,   CALL[DINTOIR-5]                 ;  PUTS D => IR, BUT(INSTRS) TO TEST FOR ZERO
      (5241)  DCS[0,00,0,0,0,0]  BM[0101,,00,01,,01,00,,010,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,111,011]




        5242:  ;(FREE)
        SCOPE374B:
                NEXT,   BUTO[SCOPE],                    ;NO ERROR:  "EXPEC374C1" (+1, WORDS)
                        J/EXPEC374C1                    ;  ERROR:  "ZERO374B1" (+0, WORDS)
      (5242)  DCS[0,00,0,1,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,101,101,111]
```



```

        ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        ;*** TEST 374C ***

        5557:
        EXPEC374C1:
                P3,     CSPD[02]_EMIT, EMIT/010001,     ;EXPECTED "SERIAL" REPRESENTATION OF RESULT
                NEXT,   J/ZERO374C1                     ;
      (5557)  DCS[0,00,0,0,0,0]  BM[0001,,10,00,,00,00,,000,,001,,,0,0,0,,0,,0,,,0,1101,,,1,,0000,0,,,11,000,,,101,110,110]

        5566:
        ZERO374C1:
                P0,     BUMP-VERIFY,                    ;COUNT
                SETUP,  RETURN/DOWRITE374C1,            ;EXEC SUBR WHICH:
                                                        ;(1) (000204) => IR,
                NEXT,   CALL[ZEROSF020F04]              ;(2)  WRITES ZEROES TO A/B SP HI/LO SF/DF
      (5566)  DCS[0,00,0,0,0,1]  BM[0101,,00,01,,01,00,,011,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,000,011,110]

        5243:  ;(FREE)
        DOWRITE374C1:
                P2-T,   D_CSPD(D01), D[C]_0,            ;DATA WITH BIT<07> SET
                        ASP-ADDRS-R[DF],                ;ADDRESS ASP WITH DF MODE
                        BSP-ADDRS-R[SF],                ;ADDRESS BSP WITH SF MODE
                P3,     WR(B,LO,A-ADDR),                ;SELECT THE PARTICULAR FUNCTION TO TEST.
                NEXT,   J/GETTEM374C1                   ;  USING A-ADDR FOR REWRITE
      (5243)  DCS[0,00,0,0,0,0]  BM[1010,,10,01,,00,10,,000,,000,,,0,1,0,,0,,0,,,0,1110,,,0,,0010,0,,,11,000,,,010,100,100]

        5244:  ;(FREE)
        GETTEM374C1:
                P3,     CSPD[00]_EMIT, RETURN/ZERO374C2,        ;(SEE DESCRIP OF SUBR FOR FUNCTION)
                NEXT,   CALL[SFDFTOSR]                  ;
      (5244)  DCS[0,00,0,0,0,0]  BM[0101,,10,01,,01,00,,110,,101,,,0,0,0,,0,,0,,,0,1111,,,1,,0000,0,,,11,100,,,111,100,110]



        5246:  ;(FREE)
        ZERO374C2:
                SETUP,  RETURN/DOWRITE374C2,            ;AGAIN GO WRITE ZEROES TO A/B-SP-HI/LO
                NEXT,   CALL[ZEROSF020F04]              ;
      (5246)  DCS[0,00,0,0,0,0]  BM[0101,,00,01,,01,00,,111,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,000,011,110]

        5247:  ;(FREE)
        DOWRITE374C2:
                P2-T,   D_CSPD(D01), D[C]_0,            ;DATA WITH BIT<07> SET
                        ASP-ADDRS-R[DF],                ;ADDRESS ASP WITH DF MODE

                        BSP-ADDRS-R[SF],                ;ADDRESS BSP WITH SF MODE
                P3,     WR(B,LO,B-ADDR),                ;USE SAME FUNCTION AS ABOVE,
                NEXT,   J/GETTEM374C2                   ;  ONLY USE B-ADDR FOR REWRITE THIS TIME
      (5247)  DCS[0,00,0,0,0,0]  BM[1010,,10,01,,00,10,,000,,000,,,0,1,0,,0,,0,,,0,1110,,,0,,0110,0,,,11,000,,,010,101,000]

        5250:  ;(FREE)
```

```
10100    GETTEM374C2:
10101            P3,     CSPD[00]_EMIT, RETURN/TEST374C2,        ;(SEE DESCRIP OF SUBR FOR FUNCTION)
10102            NEXT,   CALL[SPDFTO&R]                          ;
     (5250)  DCS[0,00,0,0,0,0]  BM[0101,,10,10,,10,10,,100,,101,,,0,0,0,,0,,0,,,0,1111,,,1,,0000,0,,,11,100,,,111,100,110]
10103
10104    5524:
10105    TEST374C2:
10106            P0,     LOAD-ENUA(ETARGET434),                  ;NOW SETUP FOR IR=ZERO COMPARE
10107                    LOAD-ERROR(TEST374C2),                  ;ERROR DIRECTORY KEY
10108                    DCS-CTR(C6,),                           ;COMPARE AT TARGET
10109                    BUMP-VERIFY,                            ;COUNT
10110            NEXT,   J/GOTEST374C2                           ;
     (5524)  DCS[1,00,1,0,0,1]  BM[1001,,00,11,,11,00,,011,,100,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,010,101,001]
10111
10112    5251:  ;(FREE)
10113    GOTEST374C2:
10114            SETUP,  RETURN/SCOPE374C,                       ;GO EXEC SUBR THAT:
10115            NEXT,   CALL[DINTOIR=5]                         ;  PUTS D -> IR, BUT(INSTRS) TO TEST FOR ZERO
     (5251)  DCS[0,00,0,0,0,0]  BM[0101,,00,01,,01,01,,010,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,111,011]
10116
10117
10118
10119
10120    5252:  ;(FREE)
10121    SCOPE374C:
10122            NEXT,   BUTD[SCOPE],                            ;NO ERROR:  "EXPEC374D1" (+1, WORDS)
10123                    J/EXPEC374D1                            ;   ERROR:  "ZERO374C1" (-1, WORDS)
     (5252)  DCS[0,00,0,1,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,101,110,111]
10124
10125
10126
10127
10128    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
10129
10130    ;*** TEST 374D ***
10131
10132    5567:
10133    EXPEC374D1:
10134            P3,     CSPD[02]_EMIT, EMIT/001040,             ;EXPECTED "SERIAL" REPRESENTATION OF RESULT
10135            NEXT,   J/ZERO374D1                             ;
     (5567)  DCS[0,00,0,0,0,0]  BM[0000,,10,00,,10,00,,100,,000,,,0,0,0,,0,,0,,,0,1101,,,1,,0000,0,,,11,000,,,101,001,100]
10136
10137    5514:
10138    ZERO374D1:
10139            P0,     BUMP-VERIFY,                            ;COUNT
10140            SETUP,  RETURN/DOWRITE374D1,                    ;EXEC SUBR WHICH:
10141                                                            ;(1) (0002)04) -> IR,
10142            NEXT,   CALL[ZEROSF02DF04]                      ;(2)  WRITES ZEROES TO A/B SP HI/LO SF/DF
     (5514)  DCS[0,00,0,0,0,1]  BM[0101,,00,01,,01,01,,011,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,000,011,110]
10143
10144    5253:  ;(FREE)
10145    DOWRITE374D1:
```



```
10146            P2=Y,   D_CSPD(D01), D[C]_0,                    ;DATA WITH BIT<07> SET
10147                    ASP=ADDRS-R[SF],                        ;ADDRESS ASP WITH SF MODE
10148                    BSP=ADDRS-R[DF],                        ;ADDRESS BSP WITH DF MODE
10149            P3,     WR(B,HI,A-ADDR),                        ;SELECT ONE PARTICULAR FUNCTION TO TEST,
10150            NEXT,   J/GETTEM374D1                           ;  USING A-ADDR FOR REWRITE
     (5253)  DCS[0,00,0,0,0,0]  BM[1010,,10,00,,00,11,,000,,000,,,0,1,0,,0,,0,,,0,1110,,,0,,1010,0,,,11,000,,,010,101,100]
10151
10152    5254:  ;(FREE)
10153    GETTEM374D1:
10154            P3,     CSPD[00]_EMIT, RETURN/ZERO374D2,        ;(SEE DESCRIP OF SUBR FOR FUNCTION)
10155            NEXT,   CALL[SPDFTO&R]                          ;
     (5254)  DCS[0,00,0,0,0,0]  BM[0101,,10,01,,01,01,,101,,101,,,0,0,0,,0,,0,,,0,1111,,,1,,0000,0,,,11,100,,,111,100,110]
10156
10157
10158
10159    5255:  ;(FREE)
10160    ZERO374D2:
10161            SETUP,  RETURN/DOWRITE374D2,                    ;AGAIN GO WRITE ZEROES TO A/B-SP-HI/LO
10162            NEXT,   CALL[ZEROSF02DF04]                      ;
     (5255)  DCS[0,00,0,0,0,0]  BM[0101,,00,01,,01,01,,110,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,000,011,110]
10163
10164    5256:  ;(FREE)
10165    DOWRITE374D2:
10166            P2=Y,   D_CSPD(D01), D[C]_0,                    ;DATA WITH BIT<07> SET
10167                    ASP=ADDRS-R[SF],                        ;ADDRESS ASP WITH SF MODE
10168                    BSP=ADDRS-R[DF],                        ;ADDRESS BSP WITH DF MODE
10169            P3,     WR(B,HI,B-ADDR),                        ;USE SAME FUNCTION AS ABOVE,
10170            NEXT,   J/GETTEM374D2                           ;  ONLY USE B-ADDR FOR REWRITE THIS TIME
     (5256)  DCS[0,00,0,0,0,0]  BM[1010,,10,00,,00,11,,000,,000,,,0,1,0,,0,,0,,,0,1110,,,0,,1110,0,,,11,000,,,010,101,111]
10171
10172    5257:  ;(FREE)
10173    GETTEM374D2:
10174            P3,     CSPD[00]_EMIT, RETURN/TEST374D2,        ;(SEE DESCRIP OF SUBR FOR FUNCTION)
10175            NEXT,   CALL[SPDFTO&R]                          ;
     (5257)  DCS[0,00,0,0,0,0]  BM[0101,,10,10,,10,00,,000,,101,,,0,0,0,,0,,0,,,0,1111,,,1,,0000,0,,,11,100,,,111,100,110]
10176
10177    5500:
10178    TEST374D2:
10179            P0,     LOAD-ENUA(ETARGET434),                  ;NOW SETUP FOR IR=ZERO COMPARE
10180                    LOAD-ERROR(TEST374D2),                  ;ERROR DIRECTORY KEY
10181                    DCS-CTR(C6,),                           ;COMPARE AT TARGET
10182                    BUMP-VERIFY,                            ;COUNT
10183            NEXT,   J/GOTEST374D2                           ;
     (5500)  DCS[1,00,1,0,0,1]  BM[1001,,00,11,,11,00,,011,,100,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,010,110,000]
10184
10185    5260:  ;(FREE)

10186    GOTEST374D2:
10187            SETUP,  RETURN/SCOPE374D,                       ;GO EXEC SUBR THAT:
10188            NEXT,   CALL[DINTOIR=5]                         ;  PUTS D -> IR, BUT(INSTRS) TO TEST FOR ZERO

     (5260)  DCS[0,00,0,0,0,0]  BM[0101,,00,01,,01,10,,001,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,111,011]
10189
10190
```

```
10191
10192
10193    5261:  1(FREE)
10194    SCOPE374D:
10195            NEXT,   BUTD[SCOPE],                    ;NO ERROR:  "EXPEC374E1" (+1, WORDS)
10196                    J/EXPEC374E1                    ;   ERROR:  "ZERO374D1" (+8, WORDS)
     (5261)  DCS[0,00,0,1,0,0]  BM[0000,,00,00,,00,08,,000,,000...0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,101,001,101]
10197
10198
10199
10200
10201    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
10202
10203    ;*** TEST 374E ***
10204
10205    5515:
10206    EXPEC374E1:
10207            P3,     CSPD[02]_EMIT, EMIT/002520,     ;EXPECTED "SERIAL" REPRESENTATION OF RESULT
10208            NEXT,   J/ZERO374E1                     ;
     (5515)  DCS[0,00,0,0,0,0]  BM[0000,,10,01,,01,01,,010,,000...0,0,0,,0,,0,,,0,1101,,,1,,0000,0,,,11,000,,,101,010,110]
10209
10210    5526:
10211    ZERO374E1:
10212            P0,     BUMP=VERIFY,                    ;COUNT
10213            SETUP,  RETURN/DDWRITE374E1,            ;EXEC SUBR WHICH:
10214                                                    ;(1) (000402) -> IR,
10215            NEXT,   CALL[ZERO&F04DF02]              ;(2)  WRITES ZEROES TO A/B SP HI/LO SF/DF
     (5526)  DCS[0,00,0,0,0,1]  BM[0101,,00,01,,01,10,,010,,111...0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,000,100,001]
10216
10217    5262:  1(FREE)
10218    DDWRITE374E1:
10219            P2=T,   D_CSPD[001], D[C]=0,            ;DATA WITH BIT<07> SET
10220                    ASP=ADDR&=R[SF],                ;ADDRESS ASP WITH SF MODE
10221                    BSP=ADDR&=R[DF],                ;ADDRESS BSP WITH DF MODE
10222            P3,     WR(AB,LO,A=ADDR),               ;SELECT THE PARTICULAR FUNCTION TO TEST,
10223            NEXT,   J/GETTEM374E1                   ;  USING A=ADDR FOR REWRITE
     (5262)  DCS[0,00,0,0,0,0]  BM[1010,,10,00,,00,11,,000,,000...0,1,0,,0,,0,,,0,1110,,,0,,0011,0,,,11,000,,,010,110,011]
10224
10225    5263:  1(FREE)
10226    GETTEM374E1:
10227            P3,     CSPD[00]_EMIT, RETURN/ZERO374E2,        ;(SEE DESCRIP OF SUBR FOR FUNCTION)
10228            NEXT,   CALL[&FDFTO&F]                  ;
     (5263)  DCS[0,00,0,0,0,0]  BM[0101,,10,01,,01,10,,100,,101...0,0,0,,0,,0,,,0,1111,,,1,,0000,0,,,11,100,,,111,100,110]
10229
10230
10231
10232    5264:  1(FREE)
10233    ZERO374E2:
10234            SETUP,  RETURN/DDWRITE374E2,            ;AGAIN GO WRITE ZEROES TO A/B=SP=HI/LO
10235            NEXT,   CALL[ZERO&F04DF02]              ;
     (5264)  DCS[0,00,0,0,0,0]  BM[0101,,00,01,,01,10,,101,,111...0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,000,100,001]
10236
```



```
10237    5265:  1(FREE)
10238    DDWRITE374F2:
10239            P2=T,   D_CSPD[001], D[C]=0,            ;DATA WITH BIT<07> SET
10240                    ASP=ADDR&=R[SF],                ;ADDRESS ASP WITH SF MODE
10241                    BSP=ADDR&=R[DF],                ;ADDRESS BSP WITH DF MODE
10242            P3,     WR(AB,LO,B=ADDR),               ;USE SAME FUNCTION AS ABOVE,
10243            NEXT,   J/GETTEM374E2                   ;  ONLY USE B=ADDR FOR REWRITE THIS TIME
     (5265)  DCS[0,00,0,0,0,0]  BM[1010,,10,00,,00,11,,000,,000...0,1,0,,0,,0,,,0,1110,,,0,,0111,0,,,11,000,,,010,110,110]
10244
10245    5266:  1(FREE)
10246    GETTEM374F2:
10247            P3,     CSPD[00]_EMIT, RETURN/TEST374E2,        ;(SEE DESCRIP OF SUBR FOR FUNCTION)
10248            NEXT,   CALL[&FDFTO&F]                  ;
     (5266)  DCS[0,00,0,0,0,0]  BM[0101,,10,10,,10,00,,010,,101...0,0,0,,0,,0,,,0,1111,,,1,,0000,0,,,11,100,,,111,100,110]
10249
10250    5502:
10251    TEST374F2:
10252            P0,     LOAD=FNUA(&TARGET434),          ;NOW SETUP FOR IR=ZERO COMPARE
10253                    LOAD=ERROR(TEST374E2),          ;ERROR DIRECTORY KEY
10254                    DCS=CTR(C6,),                   ;COMPARE AT TARGET
10255                    BUMP=VERIFY,                    ;COUNT
10256            NEXT,   J/GOTEST374E2                   ;
     (5502)  DCS[1,00,1,0,0,1]  BM[1001,,00,11,,11,00,,011,,100...0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,010,110,111]
10257
10258    5267:  1(FREE)
10259    GOTEST374F2:
10260            SETUP,  RETURN/SCOPE374E,               ;GO EXEC SUBR THAT:
10261            NEXT,   CALL[DINTOIR=5]                 ;  PUTS D => IR, BUT(INSTR5) TO TEST FOR ZERO
     (5267)  DCS[0,00,0,0,0,0]  BM[0101,,00,01,,01,11,,000,,111...0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,111,011]
10262
10263
10264
10265
10266    5270:  1(FREE)
10267    SCOPE374E:
10268            NEXT,   BUTD[SCOPE],                    ;NO ERROR:  "EXPEC374F1" (+1, WORDS)
10269                    J/EXPEC374F1                    ;   ERROR:  "ZERO374E1" (+8, WORDS)
     (5270)  DCS[0,00,0,1,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,101,010,111]
10270
10271
10272
10273
10274    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
10275
10276    ;*** TEST 374F ***
10277

10278    5527:
10279    EXPEC374F1:
10280            P3,     CSPD[02]_EMIT, EMIT/120012,     ;EXPECTED "SERIAL" REPRESENTATION OF RESULT
10281            NEXT,   J/ZERO374F1                     ;
     (5527)  DCS[0,00,0,0,0,0]  BM[1010,,10,00,,00,00,,001,,010...0,0,0,,0,,0,,,0,1101,,,1,,0000,0,,,11,000,,,101,001,110]
10282
10283    5516:
```

```
10284    ZERO374F1:
10285            P0,     BUMP=VERIFY,                            ;COUNT
10286            SETUP,  RETURN/DOWRITE374F1,                    ;EXEC SUBR WHICH:
10287                                                            ;(1) (000204) -> IR,
10288            NEXT,   CALL[ZEROSF02DF04]                      ;(2)  WRITES ZEROES TO A/B SP HI/LO SF/DF
     (5516)  DCS[0,00,0,0,0,1]  BM[0101,,00,01,,01,11,,001,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,000,011,110]
10289
10290    5271:  ;(FREE)
10291    DOWRITE374F1:
10292            P2=T,   D_CSPD(D01), D[C]_0,                    ;DATA WITH BIT<07> SET
10293                    ASP=ADDR8+R[DF],                        ;ADDRESS ASP WITH DF MODE
10294                    BSP=ADDR8+R[SF],                        ;ADDRESS BSP WITH SF MODE
10295            P3,     WR(AB,NI,A+ADDR),                       ;SELECT THE PARTICULAR FUNCTION TO TEST,
10296            NEXT,   J/GETTEM374F1                           ;  USING A=ADDR FOR REWRITE
     (5271)  DCS[0,00,0,0,0,0]  BM[1010,,10,01,,00,10,,000,,000,,,0,1,0,,0,,0,,,0,1110,,,0,,1011,0,,,11,000,,,010,111,010]
10297
10298    5272:  ;(FREE)
10299    GETTEM374F1:
10300            P3,     CSPD[00]_EMIT, RETURN/ZERO374F2,        ;(SEE DESCRIP OF SUBR FOR FUNCTION)
10301            NEXT,   CALL[SFDFTO8R]                          ;
     (5272)  DCS[0,00,0,0,0,0]  BM[0101,,10,01,,01,11,,011,,101,,,0,0,0,,0,,0,,,0,1111,,,1,,0000,0,,,11,100,,,111,100,110]
10302
10303
10304
10305    5273:  ;(FREE)
10306    ZERO374F2:
10307            SETUP,  RETURN/DOWRITE374F2,                    ;AGAIN GO WRITE ZEROES TO A/B=SP=HI/LO
10308            NEXT,   CALL[ZEROSF02DF04]                      ;
     (5273)  DCS[0,00,0,0,0,0]  BM[0101,,00,01,,01,11,,100,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,000,011,110]
10309
10310    5274:  ;(FREE)
10311    DOWRITE374F2:
10312            P2=T,   D_CSPD(D01), D[C]_0,                    ;DATA WITH BIT<07> SET
10313                    ASP=ADDR8+R[DF],                        ;ADDRESS ASP WITH DF MODE
10314                    BSP=ADDR8+R[SF],                        ;ADDRESS BSP WITH SF MODE
10315            P3,     WR(AB,NI,B+ADDR),                       ;USE SAME FUNCTION AS ABOVE,
10316            NEXT,   J/GETTEM374F2                           ;  ONLY USE B=ADDR FOR REWRITE THIS TIME
     (5274)  DCS[0,00,0,0,0,0]  BM[1010,,10,01,,00,10,,000,,000,,,0,1,0,,0,,0,,,0,1110,,,0,,1111,0,,,11,000,,,010,111,101]
10317
10318    5275:  ;(FREE)
10319    GETTEM374F2:
10320            P3,     CSPD[00]_EMIT, RETURN/TEST374F2,        ;(SEE DESCRIP OF SUBR FOR FUNCTION)
10321            NEXT,   CALL[SFDFTO8R]                          ;
     (5275)  DCS[0,00,0,0,0,0]  BM[0101,,10,10,,10,00,,100,,101,,,0,0,0,,0,,0,,,0,1111,,,1,,0000,0,,,11,100,,,111,100,110]
10322
10323    5504:
10324    TEST374F2:
10325            P0,     LOAD=ENUA(ZTARGET434),                  ;NOW SETUP FOR IR=ZERO COMPARE
10326                    LOAD=ERROR(TEST374F2);                  ;ERROR DIRECTORY KEY
10327                    DCS-CTR(C6,),                           ;COMPARE AT TARGET
10328                    BUMP=VERIFY,                            ;COUNT
10329            NEXT,   J/GOTEST374F2                           ;
```



```
     (5504)  DCS[1,00,1,0,0,1]  BM[1001,,00,11,,11,00,,011,,100,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,010,111,110]
10330
10331    5276:  ;(FREE)
10332    GOTEST374F2:
10333            SETUP,  RETURN/SCOPE374F,                       ;GO EXEC SUBR THAT;
10334            NEXT,   CALL[DISTOIP-5]                         ;  PUTS D -> IR, BUT(INSTRS) TO TEST FOR ZERO
     (5276)  DCS[0,00,0,0,0,0]  BM[0101,,00,01,,01,11,,111,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,111,011]
10335
10336
10337
10338
10339    5277:  ;(FREE)
10340    SCOPE374F:
10341            P2,     RES_CSPD(C000000),                      ;RESET RES TO SR=LOAD/GUARD=DISABLED
10342            NEXT,   PUTD[SCOPE],                            ;NO ERROR:  "TEST375A" (+17, WORDS)
10343                    J/TEST375A                              ;   ERROR:  "ZERO374F1" (-8, WORDS)
     (5277)  DCS[0,00,0,1,0,0]  BM[0000,,10,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0100,,,0,,1000,1,,,11,000,,,101,001,111]
10344
10345
10346
10347
10348    ;  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -
10349
10350    ;       THE FOLLOWING SUBROUTINES ARE USED IN THE ABOVE TESTS (374 A1-F2)!
10351    ;
10352    ;       THIS FIRST SUBROUTINE LOADS THE IR WITH THE APPROPRIATELY
10353    ;               SELECTED SF AND DF FIELD VALUES, THEN PROCEEDS TO
10354    ;               ZERO OUT THE ASP AND BSP LOCATIONS CORRESPONDING TO
10355    ;               THESE VALUES, IN BOTH ASP/BSP HI/LO SP'S.
10356    ;
10357    ;               NOTE:  THIS SUBR MUST BE ENTERED WITH BUSDIN_EMIT AND
10358    ;                       EN CLK IN PROC *CONS ENABLED
10359    ;
10360    ;               ALSO:  THE WRITE FUNCTION USED IS:  WR(AB,W,A) OR WR(AB,L,A)
10361    ;                       WITH A-ADDR = SF OR DF, RESPECTIVELY
10362    ;
10363
10364    7036:  ;(FREE)
10365    ZERO8F02DF04:
10366            P2=U,   IR_EMIT, EMIT/000204,                   ;SF=2, DF=4
10367            NEXT,   J/ZERO8FDF                              ;GO ZERO
     (7036)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,10,,000,,100,,,0,0,0,,0,,0,,,1,1010,,,0,,0000,0,,,11,000,,,000,100,010]
10368
10369    7041:  ;(FREE)
10370    ZERO8F04DF02:
10371            P2=U,   IR_EMIT, EMIT/000402,                   ;SF=4, DF=2
10372            NEXT,   J/ZERO8FDF                              ;GO ZERO
     (7041)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,01,00,,000,,010,,,0,0,0,,0,,0,,,1,1010,,,0,,0000,0,,,11,000,,,000,100,010]
10373
10374    7042:  ;(FREE)
10375    ZERO8FDF:
10376            P2=T,   D_ZERO, SAVE=D[C],                      ;ZEROES
10377            P3,     ASRSPHI[DF]_D,                          ;
```

```
                NEXT,   J/ZEROSFDFA                             ;
     (7042)  DCS[0,00,0,0,0,0]  BM[0011,,00,00,,00,10,,000,,111,,,0,1,0,,0,,0,,,0,0000,,,0,,1011,0,,,11,000,,,000,100,011]

        7043:  ;(FREE)
        ZEROSFDFA:
                P3,     A*BSPLO[DF]_D,                          ;
                NEXT,   J/ZEROSF                                ;
     (7043)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,10,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0011,0,,,11,000,,,000,100,100]

        7044:  ;(FREE)
        ZEROSF:
                P2-T,   D_ZERO, SAVE=D[C],                      ;ENTER HERE FOR SF ONLY
                P3,     A*BSPHI[SF]_D,                          ;
                NEXT,   J/ZEROSFA                               ;
     (7044)  DCS[0,00,0,0,0,0]  BM[0011,,00,01,,00,11,,000,,111,,,0,1,0,,0,,0,,,0,0000,,,0,,1011,0,,,11,000,,,000,100,101]

        7045:  ;(FREE)
        ZEROSFA:
                P3,     A*BSPLO[SF]_D,                          ;
                NEXT,   BUTA(RETURN),                           ;AND RETURN
                        J/BUTERROR7                             ;
     (7045)  DCS[0,00,0,0,0,0]  BM[0000,,00,01,,00,11,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0011,0,,,11,111,,,011,111,110]



        ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        ;       THIS SECOND SUBROUTINE NOW READS BACK ALL THE REGISTERS IN ASP/BSP,
        ;               HI/LO, SF/DF THAT THE WRITE COULD HAVE REFERENCED.
        ;               IF LOCATION WAS WRITTEN, BIT<07> SHOULD BE SET,
        ;               ELSE IT SHOULD REMAIN CLEAR (FROM ZEROSFXXDFXX ROUTINE)
        ;               THE BITS ARE SHIFTED SERIALLY INTO THE SR, AND AFTER
        ;               TWO WRITES (WHICH GIVES 16. BITS, THE SR JUST FILLED),
        ;               THE COMPARE IS MADE BETWEEN EXPECTED:RECEIVED ANSWERS.
        ;
        ;               NOTE:  THIS SUBR REQUIRES SR-LEFT/GUARD-DISABLED SETUP,
        ;                       AND CSP(02) CONTAINS EXPECTED ANSWER,
        ;                       CSP(00) CONTAINS RETURN ADDRESS, IN BITS<14:03>.
        ;

        5746:
        SFDFTOSR:
                P0,     LOAD-ENUA(ZTARGET777),                  ;COMPARE POINT IS AT EXIT
                        LOAD-ERROR(SFDFTOSR),                   ;THEORETICALLY THIS CODE SHOULD NEVER BE USED
                        DCS-CTR(C11,1),                         ;COMPARE AT TARGET
                NEXT,   GOTO-PAGE(7),                           ;SUBR IS HERE, NOTE OVERLAP W/ BIT<2:0> OF ENUA
                        J/SFDFTOSRAA                            ;
     (5746)  DCS[1,00,1,0,0,0]  BM[0100,,00,11,,11,11,,111,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,000,100,000]

        7040:  ;(FREE)
        SFDFTOSRAA:
                P2-T,   D_ASPHI[DF], D[C]_ALU07,                ;SR<15,07> = ASPHI/DF
                NEXT,   J/SFDFTOSRA                             ;
```



```
     (7040)  DCS[0,00,0,0,0,0]  BM[1111,,00,00,,11,10,,000,,011,,,0,1,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,100,111]

        7047:  ;(FREE)
        SFDFTOSRA:
                P2-T,   SR_SR-LEFT=1,                           ;STORE PAST
                        D_ASPLO[DF], D[C]_ALU07,                ;SR<14,06> = ASPLO/DF
                NEXT,   J/SFDFTOSRB                             ;
     (7047)  DCS[0,00,0,0,0,0]  BM[1111,,00,00,,10,10,,000,,011,,,0,1,1,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,101,000]

        7050:  ;(FREE)
        SFDFTOSRB:
                P2-T,   SR_SR-LEFT=1,                           ;STORE PAST
                        D_BSPHI[DF], D[C]_ALU07,                ;SR<13,05> = BSPHI/DF
                NEXT,   J/SFDFTOSRC                             ;
     (7050)  DCS[0,00,0,0,0,0]  BM[1010,,01,00,,00,00,,000,,011,,,0,1,1,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,101,001]

        7051:  ;(FREE)
        SFDFTOSRC:
                P2-T,   SR_SR-LEFT=1,                           ;STORE PAST
                        D_BSPLO[DF], D[C]_ALU07,                ;SR<12,04> = BSPLO/DF
                NEXT,   J/SFDFTOSRD                             ;
     (7051)  DCS[0,00,0,0,0,0]  BM[1010,,00,00,,00,00,,000,,011,,,0,1,1,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,101,010]

        7052:  ;(FREE)
        SFDFTOSRD:
                P2-T,   SR_SR-LEFT=1,                           ;STORE PAST
                        D_ASPHI[SF], D[C]_ALU07,                ;SR<11,03> = ASPHI/SF
                NEXT,   J/SFDFTOSRE                             ;
     (7052)  DCS[0,00,0,0,0,0]  BM[1111,,00,00,,11,11,,000,,011,,,0,1,1,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,101,011]

        7053:  ;(FREE)
        SFDFTOSRE:
                P2-T,   SR_SR-LEFT=1,                           ;STORE PAST
                        D_ASPLO[SF], D[C]_ALU07,                ;SR<10,02> = ASPLO/SF
                NEXT,   J/SFDFTOSRF                             ;
     (7053)  DCS[0,00,0,0,0,0]  BM[1111,,00,00,,10,11,,000,,011,,,0,1,1,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,101,100]

        7054:  ;(FREE)
        SFDFTOSRF:
                P2-T,   SR_SR-LEFT=1,                           ;STORE PAST
                        D_BSPHI[SF], D[C]_ALU07,                ;SR<09,01> = BSPHI/SF
                NEXT,   J/SFDFTOSRG                             ;
     (7054)  DCS[0,00,0,0,0,0]  BM[1010,,01,01,,00,00,,000,,011,,,0,1,1,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,101,101]

        7055:  ;(FREE)
        SFDFTOSRG:

                P2-T,   SR_SR-LEFT=1,                           ;STORE PAST
                        D_BSPLO[SF], D[C]_ALU07,                ;SR<08,00> = BSPLO/SF
                NEXT,   J/SFDFTOSRH                             ;
     (7055)  DCS[0,00,0,0,0,0]  BM[1010,,00,01,,00,00,,000,,011,,,0,1,1,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,000,101,110]

        7056:  ;(FREE)
        SFDFTOSRH:
                P2-T,   SR_SR-LEFT=1,                           ;STORE PAST
```

```
10472                   D_CSPD(D00), D(C)_8,                    ;RETRIEVE RETURN ADDRESS
10473             NEXT, J/SPDFTOSRI                             ;
     (70056)  DCS[0,00,0,0,0,0]  BM[1010..10,00..00,00..000..000...0,1,1...0..0..0...0,1111...0..0000,0...11,000...000,101,111]
10474
10475     7057:  ;(FREE)
10476     SPDFTOSRI:
10477             P0,     RETURN_D[14-03], PAGE(7),               ;PUT RETURN ADDRESS INTO RETURN REGISTER,
10478             P2=T,   D_SP=XOR=CSPD(D02), SAVE=D(C),          ;COMPARE RECEIVED:EXPECTED
10479             NEXT,   J/ZTARGET777                            ;GO DO A BUVA(RETURN)
     (70057)  DCS[0,00,0,0,0,0]  BM[0110..10,00..00,00..000..111...0,1,0...0..0...0,1101...0..0000,0...11,101...111,111,111]
10480
10481
10482
10483
10484
10485     ;.PAGE*****************************************************************************************************
10486
10487
10488     .TOC    *    TEST375-376:  BYTE WRITE TO ASP/BSP LO, SP ADDRS R=OR=1/FLIPT=INHIBIT
10489
10490
10491     ;*****************************************************************************************
10492     ;*
10493     ;*      TESTS:  375A - 376                              UWORDS:  020 + 017
10494     ;*
10495     ;*      FUNCTIONS:
10496     ;*
10497     ;*      THE FOLLOWING SET OF THREE CHECKS TESTS THE FOLLOWING FEATURES OF THE ASP/BSP:
10498     ;*
10499     ;*      TEST 375 A/B TEST THAT BYTE WRITES CAN BE PERFORMED TO ASP/BSP LO, USING THE
10500     ;*      "DAD" EXTENSION BIT COMBINATION.
10501     ;*
10502     ;*      TEST 376 VERIFIES THAT A FLOATING POINT INSTRUCTION IN THE IR FORCES SP=ADDRS
10503     ;*      BIT<09> TO A (0), AND THAT BUTA(R=OR=1) IN THE REFERENCING MICROWORD FORCES
10504     ;*      BIT<00> TO A (1).
10505     ;*
10506     ;*****************************************************************************************
10507
10508
10509
10510     ;TEST 375 A CHECKS THAT THE DAD BIT COMBINATION /11 DOES A BYTE WRITE (IE, LO BYTE ONLY)
10511     55171
10512     TEST375A:
10513             P0,     LOAD=ENUA(ZTARGET432),                  ;SETUP FOR INSTRS IR=(000125)/E78/432
10514                     LOAD=ERROR(TEST375A),                   ;ERROR DIRECTORY KEY
10515                     DCS=CTR(C11,),                          ;COMPARE AT TARGET
10516             NEXT,   J/SETSP378A                             ;
     (55171)  DCS[1,00,1,0,0,0]  BM[0100..00,11..11,00..011..010...0,0,0...0..0...0,0000...0..0000,0...11,000...101,000,110]
10517
10518     55061
10519     SETSP375A:
10520             P2=T,   D_CSPD(C125252),                        ;FIRST SETUP SP'S WITH JUNK
10521             P3,     A:BSPLO[06]_D=(A),                      ;
10522             NEXT,   GOTO=PAGE(7),                           ;XFER FOR DCS=DAD BITS
```



```
10523                   J/FIRST375A                             ;
     (55006)  DCS[0,00,0,0,0,0]  BM[1010..10,00..00,00..111..111...0,1,0...0..0...0...0110...0..0011,0...11,100...000,100,110]
10524
10525     7046:  ;(FREE)
10526     FIRST375A:
10527             SETUP,  FIRST=1=OR=2,                           ;DAD/01, SHOULDN'T CAUSE BYTE WRITE
10528             P2=T,   D_ZERO,                                 ;SHOULD WRITE ALL ZEROES
10529             P3,     ASPLO[06]_D,                            ;
10530             NEXT,   J/SECOND375A                            ;
     (70046)  DCS[0,01,0,0,0,0]  BM[0011..00,00..00,00..111..000...0,1,0...0..0...0...0,0000...0..0001,0...11,000...000,110,001]
10531
10532     7061:  ;(FREE)
10533     SECOND375A:
10534             SETUP,  SECOND=1=OR=2,                          ;DAD/10, SHOULDN'T CAUSE BYTE WRITE
10535             P2=T,   D_ZERO,                                 ;SHOULD WRITE ALL ZEROES
10536             P3,     BSPLO[06]_D,                            ;
10537             NEXT,   J/BYTE375A                              ;
     (70061)  DCS[0,10,0,0,0,0]  BM[0011..00,10..00,00..111..000...0,1,0...0..0...0...0,0000...0..0110,0...11,000...000,110,010]
10538
10539     7062:  ;(FREE)
10540     BYTE375A:
10541             SETUP,  BYTE=WRITE,                             ;DAD/11, SHOULD CAUSE A BYTE WRITE
10542             P2=T,   D_ASPHI(C052525),                       ;WRITE THE (125) IN THE LOW BYTE
10543             P3,     A:BSPLO[06]_D=(B),                      ;UPPER BYTE SHOULD BE (000) FROM ABOVE
10544             NEXT,   J/CHECK375A                             ;
     (70062)  DCS[0,11,0,0,0,0]  BM[1111..00,10..11,01..111..000...0,1,0...0..0...0...0,0000...0..0111,0...11,000...000,110,011]
10545
10546     7063:  ;(FREE)
10547     CHECK375A:
10548             P2=T,   D_ASPLO(R06),                           ;GET THE A SIDE SP
10549             NEXT,   J/GOBUT375A                             ;
     (70063)  DCS[0,00,0,0,0,0]  BM[1111..00,00..10,00..111..000...0,1,0...0..0...0...0,0000...0..0000,0...11,000...000,110,100]
10550
10551     7064:  ;(FREE)
10552     GOBUT375A:
10553             SETUP,  RETURN/TEST375B,                        ;RETURN TO START OF NEXT SUBTEST
10554             NFXT,   CALL[D(INTOIR=5]                        ;GO CHECK (000125) OBTAINED
     (70064)  DCS[0,00,0,0,0,0]  BM[0101..00,11..00,10..101..111...0,0,0...0..0...0...0,0000...0..0000,0...11,100...010,111,011]
10555
10556
10557
10558
10559
10560
10561     ;TEST 375 B NOW CHECKS THE SAME THING (000125) IS ON THE B SIDE
10562     5625:
10563
10564     TEST375B:
10565             P0,     LOAD=ENUA(ZTARGET432),                  ;SETUP FOR INSTRS IR=(000125)/E78/432
10566                     LOAD=ERROR(TEST375B),                   ;ERROR DIRECTORY KEY
10567                     DCS=CTR(C7,),                           ;COMPARE AT TARGET
10568             NEXT,   J/CHECK375B                             ;
     (55625)  DCS[1,00,1,0,0,0]  BM[1000..00,11..11,00..011..010...0,0,0...0..0...0...0,0000...0..0000,0...11,000...011,000,000]
10569
```

```
	5300:	!(FREE)
	CHECK375B:
		P2-T,	D_BSPLO(R06),				!GET THE B SIDE SP
		NEXT,	J/GOBUT375B				!
 (5300)	DCS[0,00,0,0,0,0]  BM[1010,,00,10,,00,00,,111,,000,,,0,1,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,011,000,001]

	5301:	!(FREE)
	GOBUT375B:
		SETUP,	RETURN/TEST376A,			!RETURN TO START OF NEXT SUBTEST
		NEXT,	CALL[DINTOIR-8]				!GO CHECK (000125) OBTAINED
 (5301)	DCS[0,00,0,0,0,0]  BM[0101,,00,10,,10,01,,011,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,111,011]





	!	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-

	!TEST 376 A NOW DOES THE SP ADDRESS MODE, W/ FLTPT-INHIBIT AND BUTA(R-OR-1) ACTIVE
	!NOTE:
	!	THE ASPHI CONTAINS THE FOLLOWING VALUES IN THESE LOCATIONS:
	!
	!	ASPHI(02)	ASPHI(03)	ASPHI(06)	ASPHI(07)
	!	(052522)	(177777)	(056166)	(052525)
	!

	5513:
	TEST376A:
		P0,	LOAD-ENUA(DOIT376A),			!MAKE SURE BUTA(R-OR-1) DOESN'T CAUSE A BRANCH
			LOAD-ERROR(TEST376A),			!ERROR DIRECTORY KEY
			DCS-CTR(C7,),				!COMPARE AT TARGET
		NEXT,	J/SETIR376A				!
 (5513)	DCS[1,00,1,0,0,0]  BM[1101,,00,10,,11,00,,100,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,011,000,010]

	5302:	!(FREE)
	SETIR376A:
		SETUP,	BUTA(R-OR-1),				!SETUP ACTIVE BUT MODIFICATION OF SP BIT<00> ADDRESS
		P2-U,	IR_EMIT, EMIT/170600,			!IR=FLTPT INSTR, SF=(6)
		NEXT,	J/DOIT376A				!
 (5302)	DCS[0,00,0,0,0,0]  BM[1111,,00,00,,01,10,,000,,000,,,0,0,0,,0,,0,,,1,1010,,,0,,0000,0,,,10,010,,,100,100,000]

	5440:
	DOIT376A:
		P2-T,	D_A-XOR-B,				!COMPARE SF/OBTAINED;REGISTER EXPECTED
			BUS-A_ASPHI(SF),			!SF ON A GOES FROM (6) -> (3) FROM FLTPT,
								!AND FROM (2) -> (3) FOR (R-OR-1)
			BUS-B_BSPHI(R03),			!THIS WE EXPECT
		NEXT,	J/TEST376A1				!
 (5440)	DCS[0,00,0,0,0,0]  BM[0110,,01,11,,11,11,,101,,000,,,0,1,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,110,001,100]

	!	-	-	-	-	-	-	-	-	-	-	-	-
```



```
	!NOW CHECK THE RIGHT RESULT WAS OBTAINED
	5614:
	TEST376A1:
		P0,	LOAD-ENUA(ZTARGET434),			!SETUP FOR IR=(000000)/INSTRS COMPARE
			LOAD-ERROR(TEST376A1),			!ERROR DIRECTORY KEY
			DCS-CTR(C6,),				!COMPARE AT TARGET
		NEXT,	J/GOBUT376A				!
 (5614)	DCS[1,00,1,0,0,0]  BM[1001,,00,11,,11,00,,011,,100,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,011,000,011]

	5303:	!(FREE)
	GOBUT376A:
		SETUP,	RETURN/SCOPE376,			!RETURN TO SCOPE LOOP TEST WORD
		NEXT,	CALL[DINTOIR-5]				!CHECK FOR (000000)
 (5303)	DCS[0,00,0,0,0,0]  BM[0101,,00,01,,10,00,,100,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,111,011]



	5304:	!(FREE)
	SCOPE376:
		P0,	BUMP-VERIFY,				;COUNT
		NEXT,	BUTO[SCOPE],				!NO ERROR:  "TEST410"  (+1, WORDS)
			J/TEST410				!   ERROR:  "SETUP375A"  (-12, WORDS)
 (5304)	DCS[0,00,0,1,0,1]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,101,000,111]





	!.PAGE===================================================================================================


	.TOC	*	TEST410:  BYTE/BYTE CONSTANT/D=ZERO


	!**************************************************************************************
	!*
	!*	TESTS:  410 A - E				UWORDS:  044 + 020
	!*
	!*	FUNCTIONS:
	!*
	!*	THE FOLLOWING TESTS RUN A COUNT PATTERN THRU THE IR, MAINTAING TOTALS OF
	!*	THE NUMBER OF TIMES:
	!*
	!*	BYTE=H=LOW, BYTE/1-OR-2-FIRST=HIGH, BYTE/1-OR-2-SECOND=HIGH,
	!*	(D=ZERO)-H=HIGH, AND (D=ZERO)-H=LOW.
	!*

	!*	AT THE END, THE TESTS COMPARE THE EXPECTED COUNTS TO THE RECEIVED COUNTS.
	!*
	!**************************************************************************************



	5507:
	TEST410:
```

```
10667            P0,     LOAD=ERROR(TEST410),                    ;ERROR DIRECTORY KEY
10668                    DC6=CTR(C6,),                           ;COMPARE BELOW
10669            NEXT,   GOTO=PAGE(6),                           ;XFER
10670                    J/SETBYTEB410                           ;
    (5507)  DC6[1,00,1,0,0,0]  BM[1001,,00,00,,00,00,,000,,110...0,0,0,0,,0,,0...0,0000...0,,0000,0,,,11,100,,,101,110,000]
10671
10672    6560:
10673    SETBYTEB410:
10674            P2=T,   D_ZERO,                                 ;ZEROES INTO:
10675            P3,     A:BSPHI[10]_D=(A),                      ;ASPHI[10] = BYTE-FIRST
10676            NEXT,   GOTO=PAGE(7),                           ;
10677                    J/SETBYTEC410                           ;BSPHI[10] = WORD (=-BYTE)
    (6560)  DC6[0,00,0,0,0,0]  BM[0011,,00,00,,00,00,,000,,111...0,1,0,,0,,0...0,0000,,,0,,1011,0,,,11,100,,,000,110,000]
10678
10679    7060:  ;(FREE)
10680    SETBYTEC410:
10681            P3,     A:BSPLO[10]_D=(B),                      ;ASPLO[10] = BYTE-SECOND
10682            NEXT,   J/SETBYTED410                           ;BSPLO[10] = IR-DATA
    (7060)  DC6[0,00,0,0,0,0]  BM[0000,,00,10,,00,00,,000,,000...0,0,0,,0,,0...0,0000,,,0,,0111,0,,,11,000,,,000,110,110]
10683
10684    7066:  ;(FREE)
10685    SETBYTED410:
10686            P3,     A:BSPLO[11]_D=(B),                      ;ASPLO[11] = D-NONZERO
10687            NEXT,   GOTO=PAGE(6),                           ;BSPLO[11] = D-ZERO
10688                    J/SETBYTEE410                           ;
    (7066)  DC6[0,00,0,0,0,0]  BM[0000,,00,11,,00,00,,000,,110...0,0,0,,0,,0...0,0000,,,0,,0111,0,,,11,100,,,000,000,000]
10689
10690    6000:  ;(FREE)
10691    SETBYTEE410:
10692            P0,     BUMP=VERIFY,                            ;COUNT
10693            P3,     CSP[17]_EMIT, EMIT/000001,              ;A (1) IN BYTE-CONSTANT
10694            NEXT,   J/SETBYTEG410                           ;  LOCATION IN CSP
    (6000)  DC6[0,00,0,0,0,1]  BM[0000,,10,00,,00,00,,000,,001...0,0,0,,0,,0...0,0000,,,1,,0000,0,,,11,000,,,001,000,101]
10695
10696    6105:  ;(FREE)
10697    SETBYTEG410:
10698            P2=U,   IR_DBUF=(I),                            ;SETUP DCONS FOR D ==> IR
10699            P3,     DBUF_D=(I),                             ;
10700            NEXT,   J/TESTD410                              ;
    (6105)  DC6[0,00,0,0,0,0]  BM[0100,,00,00,,00,01,,000,,100...0,0,0,,0,,0...1,1011,,,0,,0000,0,,,11,000,,,111,111,011]
10701
10702
10703
10704    ;*** LOOP BACK ENTRY POINT ***
10705
10706    6773:
10707    TESTD410:
10708            P0,     LOAD=ENUA(GOFOR410),                    ;COMPARE AT END OF LOOP
10709                    LOAD=ERROR(TESTD410),                   ;ERROR DIRECTORY KEY
10710                    DC6=CTR(C15,),                          ;RELOAD EACH TIME THROUGH
10711            P2=U,   IR_DBUF,                                ;(DON'T CARE HERE)
```



```
10712            P3,     DBUF_D,                                 ;COPY IR-DATA FROM D ==> DBUF
10713            NEXT,   BUTF(D14-00-EQ-0),                      ;TEST D<14:00>=:
10714                    J/DNONZERO410                           ;ZERO ==> DZERO410, NONZERO ==> DNONZERO410
    (6773)  DC6[1,00,1,0,0,0]  BM[0000,,00,11,,10,11,,110,,001...0,0,0,,0,,0...1,1010...0,,0000,0,,,01,101,,,101,011,001]
10715
10716    ;ENTER HERE IF D<14:00> WAS DETECTED AS NON-ZERO
10717    6531:
10718    DNONZERO410:
10719            P2=T,   D_ASPLO[DNONZERO]=PLUS=1,               ;BUMP NON ZERO COUNTER
10720            P2=U,   IR_DBUF,                                ;COPY IR=DATA FROM DBUF ==> IR
10721            P3,     ASPLO[11]_D,                            ;SAVE NON ZERO COUNTER
10722                    DBUF_D,                                 ;(DON'T CARE HERE)
10723            NEXT,   GOTO=PAGE(7),                           ;XFR TO 7 FOR DAD
10724                    J/NEXTPAT410                            ;
    (6531)  DC6[0,00,0,0,0,0]  BM[1001,,01,11,,10,01,,000,,111...0,1,0,,0,,0...1,1010,,,0,,0001,0,,,11,100,,,000,110,101]
10725
10726    ;ENTER HERE IF D<14:00> WAS DETECTED AS ZERO
10727    6533:
10728    DZERO410:
10729            P2=T,   D_BSPLO[DZERO]=PLUS=1,                  ;BUMP ZERO COUNTER
10730            P2=U,   IR_DBUF,                                ;COPY IR=DATA FROM DBUF ==> IR
10731            P3,     BSPLO[11]_D,                            ;SAVE ZERO COUNTER
10732                    DBUF_D,                                 ;(DON'T CARE HERE)
10733            NEXT,   GOTO=PAGE(7),                           ;XFR TO 7 FOR DAD
10734                    J/NEXTPAT410                            ;
    (6533)  DC6[0,00,0,0,0,0]  BM[1001,,00,11,,11,01,,000,,111...0,1,0,,0,,0...1,1010,,,0,,0110,0,,,11,100,,,000,110,101]
10735
10736    7065:  ;(FREE)
10737    NEXTPAT410:
10738            SETUP,  FIRST=1=OR=2,                           ;SELECT DAD BITS FOR "BYTEFIRSTR410"
10739            P2=T,   D_BSPLO[IR=DATA]=PLUS=2,                ;(INCRE DATA FOR =NEXT= TIME THRU)
10740                    D[C]_COUT:5,                            ;WHEN THIS SETS WE'RE DONE
10741            P3,     BSPLO[10]_D,                            ;SAVE NEXT
10742            NEXT,   J/BYTEFIRST410                          ;
    (7065)  DC6[0,01,0,0,0,0]  BM[1100,,00,10,,11,01,,000,,110...0,1,0,,0,,0...0,0000,,,0,,0110,0,,,11,000,,,000,111,000]
10743
10744    7070:  ;(FREE)
10745    BYTEFIRST410:
10746            SETUP,  SECOND=1=OR=2,                          ;SELECT DAD BITS FOR "BYTESECOND410"
10747            P2=T,   D_ASPHI(BYTE=FIRST)=PLUS=CSP(1=0),      ;BYTE-FIRST SELECTS EITHER
10748                    SAVE=D[C],                              ;CSP[17]=1 OR CSP[13]=0
10749            P3,     ASPHI[10]_D,                            ;WRITE BACK
10750            NEXT,   BUTF(BYTE),                             ;BYTE ==> "BYTESECOND410"
10751                    J/WORD410                               ;=BYTE ==> "WORD410
    (7070)  DC6[0,10,0,0,0,0]  BM[1001,,10,00,,11,00,,000,,111...0,1,0,,0,,0...0,0100,,,0,,1001,0,,,01,001,,,011,110,110]
10752

10753    ;ENTER HERE IF BYTE=H NOT ASSERTED, IE IR=(WORD)
10754    7366:
10755    WORD410:
10756            SETUP,  SECOND=1=OR=2,                          ;SELECT DAD BITS FOR "BYTESECOND410"
10757            P2=T,   D_BSPHI[WORD]=PLUS=1,                   ;BUMP WORD=BYTE COUNTER
10758                    SAVE=D[C],                              ;SAVE PAST CARRYOUT STATUS
```

```
            P3,     BSPHI[10]_D,                              !WRITE BACK
            NEXT,   J/BYTESECOND410                           !NOW GO TRY BYTE-SECOND
     (7366)  DCS[0,10,0,0,0,0]  BM[1001,,01,10,,11,01,,000,,111,,,0,1,0,,0,,0,,,0,0000,,,0,,1110,0,,,11,000,,,011,110,111]

     !ENTER HERE IF BYTE-N WAS ASSERTED, IE IR=(BYTE)
     7367:
     BYTESECOND410:
            SETUP,  NO-DAD,                                   !KEEP FOR NOISE
            P2-T,   D_ASPLO[BYTE-SECOND]-PLUS-CSP[1-0],       !BYTE-SECOND SELECTS EITHER
                    SAVE-D[C],                                !CSP[17]=1 OR CSP[13]=0
            P3,     ASPLO[10]_D,                              !WRITE BACK
            NEXT,   BUTR(D[C]-B),                             !IF SET, SKIP OUT TO TEST-410A
                    J/GOFOR410                                !IF CLEAR, FALL THRU TO  NEXT
     (7367)  DCS[0,00,0,0,0,0]  BM[1001,,10,00,,10,00,,000,,111,,,0,1,0,,0,,0,,,0,0100,,,0,,0001,0,,,10,011,,,011,110,001]

     7361:                                                    !01, PT
     GOFOR410:
            P2-T,   D_BSPLO[IR-DATA],                         !GET DATA FOR IR INTO D
            NEXT,   GOTO-PAGE(6),                             !
                    J/TESTD410                                !LOOP BACK FOR NEXT
     (7361)  DCS[0,00,0,0,0,0]  BM[1010,,00,10,,00,00,,000,,110,,,0,1,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,111,111,011]




     ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

     !TEST 410A CHECKS THAT D<14:00>=ZERO WAS ONLY ASSERTED THICE
     7363:                                                    !11, PT
     EXPEC410A:
            NEXT,   GOTO-PAGE(6),                             !FOR LOADING DCS-CTR
                    J/TEST410A                                !
     (7363)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,110,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,111,001,001]

     6711:
     TEST410A:
            P0,     LOAD-ENUA(ZTARGET422),                    !FOR IR=(000002) W/INSTRS
                    LOAD-ERROR(TEST410A),                     !ERROR DIRECTORY KEY
                    DCS-CTR(C7,),                             !COMPARE AT TARGET
            NEXT,   J/COMP410A                                !
     (6711)  DCS[1,00,1,0,0,0]  BM[1000,,00,11,,11,00,,010,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,000,110]

     6106:  !(FREE)
     COMP410A:
            P2-T,   D_BSPLO[DZERO],                           !GET DATA
            NEXT,   J/INTOIR410A                              !
     (6106)  DCS[0,00,0,0,0,0]  BM[1010,,00,11,,00,00,,000,,000,,,0,1,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,000,111]

     6107:  !(FREE)
     INTOIR410A:
            SETUP,  RETURN/TEST410B,                          !COPY D --> IR, RESET BUSDIN_EXIT
            NEXT,   CALL[DINTOIR-5]                           ! AND CHECK ITS ALL ZERO
```



```
     (6107)  DCS[0,00,0,0,0,0]  BM[0110,,00,11,,10,00,,100,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,111,011]




     ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

     !TEST 410B CHECKS THAT D<14:00>=ZERO WAS NOT ASSERTED
     ! 32768,-2, = 32766, (077776) TIMES
     6704:
     TEST410B:
            P0,     LOAD-ENUA(ZTARGET434),                    !FOR IR=(000000) W/INSTRS
                    LOAD-ERROR(TEST410B),                     !ERROR DIRECTORY KEY
                    DCS-CTR(C8,),                             !COMPARE AT TARGET
                    BUMP-VERIFY,                              !COUNT
            NEXT,   J/EXPEC410B                               !
     (6704)  DCS[1,00,1,0,0,1]  BM[0111,,00,11,,11,00,,011,,100,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,001,000]

     6110:  !(FREE)
     EXPEC410B:
            P3,     CSPD[17]_EXIT, EXIT/077776,               !EXPECTED NUMBER OF TIMES
            NEXT,   J/COMP410B                                !D<14:00> WAS NON ZERO
     (6110)  DCS[0,00,0,0,0,0]  BM[0111,,10,11,,11,11,,111,,110,,,0,0,0,,0,,0,,,0,0000,,,1,,0000,0,,,11,000,,,001,001,001]

     6111:  !(FREE)
     COMP410B:
            P3-T,   D_ASPLO[DNONZERO]-MINUS-CSPD[17],         !COMPARE RECEIVED:EXPECTED
            NEXT,   J/GOPUT410B                               !
     (6111)  DCS[0,00,0,0,0,0]  BM[1101,,10,00,,10,01,,000,,000,,,1,1,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,001,010]

     6112:  !(FREE)
     GOPUT410B:
            SETUP,  RETURN/TEST410C,                          !GO PUT D --> IR
            NEXT,   CALL[DINTOIR-5]                           !  AND CHECK IT-8 ALL ZERO
     (6112)  DCS[0,00,0,0,0,0]  BM[0110,,00,11,,10,00,,111,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,111,011]




     ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

     !TEST 410C CHECKS THAT BYTE-N WAS NOT ASSERTED 21696,,
     !OR WAS ASSERTED 11072, TIMES IN 32768, ITERATIONS
     6707:
     TEST410C:
            P0,     LOAD-ENUA(ZTARGET402),                    !SETUP FOR D=ZERO TEST

                    LOAD-ERROR(TEST410C),                     !ERROR DIRECTORY KEY
                    DCS-CTR(C6,),                             !COMPARE AT TARGET
            NEXT,   J/EXPEC410C                               !
     (6707)  DCS[1,00,1,0,0,0]  BM[1001,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,001,011]

     6113:  !(FREE)
```

```
        EXPEC410C:
                P3,     CSPD[17]_EMIT, EMIT/52300,              ;32768.-$TIMES BYTE-H ASSERTED
                NEXT,   J/ASIDE410C                             ;  21696.=(52300)
    (6113)  DCS[0,00,0,0,0,0]  BM[0101,,10,01,,00,11,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,1,,0000,0,,,11,000,,,001,001,100]

        6114: ;(FREE)
        ASIDE410C:
                P2-T,   SP_CSPD(D17),                           ;GET ONTO A-SIDE
                NEXT,   J/COMP410C                              ;
    (6114)  DCS[0,00,0,0,0,0]  BM[1010,,10,00,,00,00,,000,,000,,,0,0,1,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,001,101]

        6115: ;(FREE)
        COMP410C:
                P3-T,   D_SP-MINUS-BSPHI[WORD],                 ;COMPARE EXPEC:RECEIVED
                NEXT,   J/GOBUT410C                             ;
    (6115)  DCS[0,00,0,0,0,0]  BM[1101,,01,10,,00,00,,000,,000,,,1,1,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,001,110]

        6116: ;(FREE)
        GOBUT410C:
                SETUP,  RETURN/TEST410D,                        ;RETURN TO START OF NEXT SUBTEST
                NEXT,   GOTO-PAGE(7),                           ;BUT TABLE
                        J/BUTD-IS-ZERO                          ;GO CHECK EQUALITY
    (6116)  DCS[0,00,0,0,0,0]  BM[0110,,00,11,,10,10,,111,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,100,001]




        ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        ;TEST 410D CHECKS THAT BYTE-CONSTANT WAS ASSERTED (4270)=2232.
        ;TIMES UNDER "FIRST-1-OR-2"
        6727:
        TEST410D:
                P0,     LOAD-ENUA(ZTARGET402),                  ;SETUP FOR D=ZERO TEST
                        LOAD-ERRDR(TEST410D),                   ;ERROR DIRECTORY KEY
                        DCS-CTR(CS,),                           ;COMPARE AT TARGET
                NEXT,   J/EXPEC410D                             ;
    (6727)  DCS[1,00,1,0,0,0]  BM[1010,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,001,111]

        6117: ;(FREE)
        EXPEC410D:
                P0,     BUMP-VERIFY,                            ;COUNT
                P3,     CSPD[17]_EMIT,                          ;
                        EMIT/004270,                            ;(4270)=2232, TIMES FOR FIRST
                NEXT,   J/COMP410D                              ;
    (6117)  DCS[0,00,0,0,0,1]  BM[0000,,10,10,,00,10,,111,,000,,,0,0,0,,0,,0,,,0,0000,,,1,,0000,0,,,11,000,,,001,010,000]

        6120: ;(FREE)
        COMP410D:
                P3-T,   D_ASPHI[BYTE-FIRST]-MINUS-CSPD[17],     ;COMPARE RECEIVED:EXPEC
                NEXT,   J/GOBUT410D                             ;
    (6120)  DCS[0,00,0,0,0,0]  BM[1101,,10,00,,11,00,,000,,000,,,1,1,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,010,001]

```



```
        6121: ;(FREE)
        GOBUT410D:
                SETUP,  RETURN/TEST410E,                        ;RETURN TO START OF NEXT SUBTEST
                NEXT,   GOTO-PAGE(7),                           ;BUT TABLE
                        J/BUTD-IS-ZERO                          ;CHECK EQUALITY
    (6121)  DCS[0,00,0,0,0,0]  BM[0110,,00,11,,10,00,,101,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,100,001]




        ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        ;TEST 410E CHECKS THAT BYTE-CONSTANT WAS ASSERTED (3600)=1920.
        ;TIMES UNDER "SECOND-1-OR-2"
        6705:
        TEST410E:
                P0,     LOAD-ENUA(ZTARGET402),                  ;SETUP FOR D=ZERO TEST
                        LOAD-ERRDR(TEST410E),                   ;ERROR DIRECTORY KEY
                        DCS-CTR(CS,),                           ;COMPARE AT TARGET
                NEXT,   J/EXPEC410E                             ;
    (6705)  DCS[1,00,1,0,0,0]  BM[1010,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,010,010]

        6122: ;(FREE)
        EXPEC410E:
                P0,     BUMP-VERIFY,                            ;COUNT
                P3,     CSPD[17]_EMIT,                          ;
                        EMIT/003600,                            ;(3600)=1920, TIMES FOR SECOND
                NEXT,   J/COMP410E                              ;
    (6122)  DCS[0,00,0,0,0,1]  BM[0000,,10,01,,11,10,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,1,,0000,0,,,11,000,,,001,010,011]

        6123: ;(FREE)
        COMP410E:
                P3-T,   D_ASPLO[BYTE-SECOND]-MINUS-CSPD[17],    ;COMPARE RECEIVED:EXPEC
                NEXT,   J/GOBUT410E                             ;
    (6123)  DCS[0,00,0,0,0,0]  BM[1101,,10,00,,10,00,,000,,000,,,1,1,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,010,100]

        6124: ;(FREE)
        GOBUT410E:
                SETUP,  RETURN/SCOPE410,                        ;RETURN TO SCOPE LOOP TEST WORD
                NEXT,   GOTO-PAGE(7),                           ;BUT TABLE
                        J/BUTD-IS-ZERO                          ;CHECK EQUALITY
    (6124)  DCS[0,00,0,0,0,0]  BM[0110,,00,00,,10,10,,101,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,100,001]


        6125: ;(FREE)
        SCOPE410:

                P0,     BUSDIN_EMIT-[I],                        ;RESET PROC UCON
                        EN-CLK-IR[15-00],                       ;
                NEXT,   BUTO[SCOPE],                            ;NO ERROR:  "TEST500" (+1, WORDS)
                        J/TEST500                               ;   ERROR:  "SETBYTE8410" (-34, WORDS)
    (6125)  DCS[0,00,0,1,0,0]  BM[0000,,00,00,,00,01,,000,,100,,,0,0,0,,0,,0,,,1,1001,,,0,,0000,0,,,11,000,,,101,110,001]

```

```
10943
10944
10945
10946
10947
10948
10949    !.PAGE********************************************************************************
10950
10951
10952    !*** VERSION /V101A0/ ***
10953
10954    .TOC   *   TEST500:  PREFETCH/OVERLAP/SP DEFEAT
10955
10956    !*****************************************************************************
10957    !*
10958    !*      TESTS:  500C - 500F                             UWORDS: 034 + 013
10959    !*
10960    !*      FUNCTIONS:      TESTS 500C - 500F RUN PATTERNS THRU THE IR TO TEST
10961    !*                      THE PREFETCH/OVERLAP ROM & THE SP WRITE DEFEAT LOGIC.
10962    !*
10963    !*****************************************************************************
10964
10965
10966
10967    6561:
10968    TEST500:
10969            P0,     LOAD=ENUA(SETFETCHG500),                ;INTERMEDIATE COMPARE AT START OF LOOP
10970                    LOAD=ERROR(TEST500),                    ;ERROR DIRECTORY KEY
10971                    DCS=CTR(C5,);                           ;COMPARE AT TARGET
10972            NEXT,   J/SETFETCHB500                          ;
     (6561)  DCS[1,00,1,0,0,0]  BM[1010,,00,11,,00,01,,011,,001,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,110,111,110]
10973
10974    6575:
10975    SETFETCHB500:
10976            P2=T,   D_ZERO, D[C]_ALU15,                     ;ZERO
10977                    SR_ZERO,                                ;TO FLAG JAMUPPS AS ILLEGAL -- KEEP ZERO THRU "TEST505"
10978            P3,     ASBSPLO[OVERLAP]_D,                     ;ZERO OVERLAP -A, -B COUNTERS
10979            NEXT,   J/SETFETCHD500                          ;
     (6575)  DCS[0,00,0,0,0,0]  BM[0011,,00,00,,10,00,,000,,100,,,0,1,1,,0,,0,,,0,0000,,,0,,0011,0,,,11,000,,,001,010,110]
10980
10981    6126:  ;(FREE)
10982    SETFETCHD500:
10983            P3,     ASBSPHI[PATTERN]_D,                     ;ZERO PATTERN (A AND B)
10984            NEXT,   J/SETFETCHE500                          ;
     (6126)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,11,01,,011,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,1011,0,,,11,000,,,001,010,111]
10985
10986    6127:  ;(FREE)
10987    SETFETCHE500:
10988            P3,     ASBSPHI[PREFETCH]_D,                    ;ZERO PREFETCH-A COUNTER, ALSO B SIDE
10989            NEXT,   J/SETFETCHF500                          ;
     (6127)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,11,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,1011,0,,,11,000,,,001,011,000]
10990
10991    6130:  ;(FREE)
10992    SETFETCHF500:
```



```
10993            P0,     BUMP=VERIFY,                            ;COUNT
10994            P3,     C&PD[17]_020010,                        ;
10995            NEXT,   J/SETFETCHG500                          ;MASK CONSTANT FOR DATA PATTERN
     (6130)  DCS[0,00,0,0,0,1]  BM[0010,,10,00,,00,00,,001,,000,,,0,0,0,,0,,0,,,0,0000,,,1,,0000,0,,,11,000,,,001,011,001]
10996
10997    6131:  ;(FREE)
10998    SETFETCHG500:
10999            P0,     BUMP=VERIFY,                            ;COUNT
11000            P2=U,   IR_DBUF=(I),                            ;SETUP THESE UCON'S,
11001            P3,     DBUF_D=(I),                             ;DON'T WORRY ABOUT EFFECT HERE
11002            NEXT,   J/TESTD500                              ;
     (6131)  DCS[0,00,0,0,0,1]  BM[0100,,00,00,,00,01,,000,,100,,,0,0,0,,0,,0,,,1,1011,,,0,,0000,0,,,11,000,,,110,111,001]
11003
11004
11005    !*** LOOP-BACK ENTRY POINT ***
11006
11007    6671:
11008    TESTD5001:
11009            P0,     LOAD=ENUA(NEXTPAT500),                  ;COMPARE POINT IN LOOP
11010                    LOAD=ERROR(TESTD500),                   ;ERROR DIRECTORY KEY
11011                    DCS=CTR(C4,);                           ;COMPARE AT TARGET
11012            P2=U,   IR_DBUF,                                ;JUST HAPPENS, DON'T CARE NOW
11013            P3,     DBUF_D,                                 ;PUT PATTERN LEFT IN D INTO DBUF
11014            NEXT,   J/LOADIP500                             ;
     (6671)  DCS[1,00,1,0,0,0]  BM[1011,,00,11,,00,11,,111,,111,,,0,0,0,,0,,0,,,1,1010,,,0,,0000,0,,,11,000,,,001,011,010]
11015
11016    6132:  ;(FREE)
11017    LOADIP500:
11018            P2=U,   IR_DBUF,                                ;PUT PATTERN NOW IN DBUF TO IR
11019            P3,     DBUF_D,                                 ;JUST HAPPENS, DON'T CARE NOW
11020            NEXT,   J/TESTINNBSP500                         ;
     (6132)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,1,1010,,,0,,0000,0,,,11,000,,,001,011,011]
11021
11022    6133:  ;(FREE)
11023    TESTINNBSP500:
11024            P2=T,   D_BSPLO[OVERLAP]-PLUS-1,                ;GET BUMPED OVERLAP COUNT, B-SIDE
11025                    D[C]_0,                                 ;
11026            P3,     BSPLO[OVERLAP]_D,                       ;ONLY WRITTEN BACK IF
11027                    BUT&[INSTR-1],                          ;"OVERLAP L" ASSERTED
11028            NEXT,   J/TESTINNASP500                         ;(BRANCH EFFECT OF BUT MASKED)
     (6133)  DCS[0,00,0,0,0,0]  BM[1001,,00,10,,11,01,,000,,000,,,0,1,0,,0,,0,,,0,0000,,,0,,0110,0,,,00,110,,,111,111,111]
11029
11030    6777:
11031    TESTINNASP500:
11032            P2=T,   D_ASPLO[OVERLAP]-PLUS-1,                ;GET BUMPED OVERLAP COUNT, A-SIDE
11033                    SAVE=D[C],                              ;

11034            P3,     ASPLO[OVERLAP]_D,                       ;ONLY WRITTEN BACK IF
11035                    BUT&[INSTR-1),                          ;  "OVERLAP L" ASSERTED
11036            NEXT,   J/NEXTPAT500                            ;(BRANCH EFFECT OF BUT MASKED)
     (6777)  DCS[0,00,0,0,0,0]  BM[1001,,01,11,,10,00,,000,,111,,,0,1,0,,0,,0,,,0,0000,,,0,,0001,0,,,00,110,,,011,111,111]
11037
11038    6377:
11039    NEXTPAT500:
11040            P2=T,   D_ASPHI[PATTERN]-PLUS-020010-PLUS-1,    ;GET NEXT DATA PATTERN IN D
```

```
                D[C]_COUT15,                            ;GET D[C]_OVERFLOW OUT OF BIT<15>
        P3,     ASBSPHI[PATTERN]_D,                     ;WRITE BACK UNMASKED NEXT PATTERN
        NEXT,   BUTR(PREFETCH-L),                       ;IF ASSERTED, "ASSERTFOV500"
                J/ASSERTFOV500                          ;OTHERWISE, "FIXPAT500"
(6377)  DCS[0,00,0,0,0,0]  BM[1100,,11,00,,11,01,,011,,110,,,0,1,0,,0,,0,,,0,0000,,,0,,1011,0,,,10,000,,,101,010,101]

    ;ENTER HERE IF "PREFETCH-L" WAS ASSERTED, LOW
    6525:
    ASSERTFOV500:
        P2-T,   D_ASPHI[PREFETCH]-PLUS-1,               ;"PREFETCH(0)H" ASSERTED, BUMP COUNT
                SAVE-D[C],                              ;(SAVE CARRYOUT)
        P3,     ASPHI[PREFETCH]_D,                      ;
        NEXT,   J/FIXPAT500                             ;
(6525)  DCS[0,00,0,0,0,0]  BM[1001,,01,11,,11,00,,000,,111,,,0,1,0,,0,,0,,,0,0000,,,0,,1001,0,,,11,000,,,101,010,111]

    ;ENTER HERE IF "PREFETCH-L" WAS NOT ASSERTED, HIGH
    6527:
    FIXPAT500:
        P2-T,   D_ASPHI[PATTERN]-AND-NOT-020010,        ;CLEAR UNNEEDED BITS
                SAVE-D[C],                              ;KEEP D[C] CONSTANT FOR "BUT"
        P3,     ASBSPHI[PATTERN]_D,                     ;RESTORE PATTERN, ALSO LEAVE IN D
        NEXT,   BUTR(D[C]-B),                           ;TEST CARRYOUT:
                J/TESTD500                              ;IF CLEAR, NEXT PATTERN AT: "TESTD500" (-6,WORDS)
(6527)  DCS[0,00,0,0,0,0]  BM[0111,,11,00,,11,01,,011,,111,,,0,1,0,,0,,0,,,0,0000,,,0,,1011,0,,,10,011,,,110,111,001]
                                                        ;IF SET, EXIT LOOP TO: "TEST500A" (+1, WORD)



    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

    ;*** TEST 500A ***
    ;CHECK THAT REGISTER OPPOSITE ASPHI[PREFETCH], IN BSPHI, STILL ZERO
    6673:
    TEST500A:
        P0,     LOAD-ENUA(ZTARGET434),                  ;SETUP FOR IR=(000000)/INSTRS TEST
                LOAD-ERROR(TEST500A),                   ;ERROR DIRECTORY KEY
                DCS-CTR(C7,);                           ;COMPARE AT TARGET
                BUMP-VERIFY,                            ;COUNT
        NEXT,   J/COMP500A                              ;
(6673)  DCS[1,00,1,0,0,1]  BM[1000,,00,11,,11,00,,011,,100,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,011,100]

    6134:  ;(FREE)
    COMP500A:
        P2-T,   D_BSPHI[PREFETCH],                      ;GET REGISTER
                SAVE-D[C],                              ;
        NEXT,   J/GOBUT500A                             ;
(6134)  DCS[0,00,0,0,0,0]  BM[1010,,01,10,,00,00,,000,,111,,,0,1,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,011,101]

    6135:  ;(FREE)
    GOBUT500A:
        P0,     BUMP-VERIFY,                            ;COUNT
        SETUP,  RETURN/TEST500C,                        ;RETURN TO START OF NEXT SUBTEST
        NEXT,   CALL[DINTOIR-5]                         ;PUT D => DBUF => IR, DO INSTRS BUT FOR (000000)
(6135)  DCS[0,00,0,0,0,1]  BM[0110,,00,11,,10,10,,110,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,111,011]
```



```
                                                        ;
                                                        ;*** ALSO SETS UP BUSDIN <- EMIT UPON EXIT ***




    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

    ;*** TEST 500C ***
    ;CHECK THAT ASSERTING "OVERLAP L" ONLY ALLOWED A SP REWRITE (DURING
    ;* BUT[INSTR-1]) IN ASPLO 4005, TIMES / 16384, DATA PATTERNS
    6776:
    TEST500C:
        P0,     LOAD-ENUA(ZTARGET402),                  ;SETUP FOR D =0 TEST
                LOAD-ERROR(TEST500C),                   ;ERROR DIRECTORY KEY
                DCS-CTR(C5,);                           ;COMPARE AT TARGET
                BUMP-VERIFY,                            ;COUNT
        NEXT,   J/EXPEC500C                             ;
(6776)  DCS[1,00,1,0,0,1]  BM[1010,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,011,110]

    6136:  ;(FREE)
    EXPEC500C:
        P3,     CSPD[EXPEC]_EMIT,                       ;EXPECTED COUNT =
                EMIT/7645,                              ;  (7645) = 4005,
        NEXT,   J/COMP500C                              ;
(6136)  DCS[0,00,0,0,0,0]  BM[0000,,10,11,,11,10,,100,,101,,,0,0,0,,0,,0,,,0,0000,,,1,,0000,0,,,11,000,,,001,011,111]

    6137:  ;(FREE)
    COMP500C:
        P0,     BUMP-VERIFY,                            ;COUNT
        P3-T,   D_ASPLO[OVERLAP]-MINUS-CSPB[EXPEC],     ;COMPARE RECEIVED:EXPECTED
                SAVE-D[C],                              ;
        NEXT,   J/GOBUT500C                             ;
(6137)  DCS[0,00,0,0,0,1]  BM[1101,,11,00,,10,00,,000,,111,,,1,1,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,100,000]

    6140:  ;(FREE)
    GOBUT500C:
        SETUP,  RETURN/TEST500D,                        ;RETURN TO START OF NEXT SUBTEST
        NEXT,   GOTO-PAGE(7),                           ;BUT TABLE
                J/BUTD-IS-ZERO                          ;CHECK EQUALITY
(6140)  DCS[0,00,0,0,0,0]  BM[0110,,00,11,,11,00,,110,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,100,001]





    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

    ;*** TEST 500D ***

    ;CHECK THAT ASSERTING "OVERLAP L" ALLOWED THE SAME NUMBER OF SP REWRITES
    ;TO ASPLO (CHECKED ABOVE) AS TO BSPLO (CHECKED HERE)
    6746:
    TEST500D:
```

```
        P0,     LOAD-ENUA(ZTARGET402),                  ;SETUP FOR D=0 TEST
                LOAD-ERROR(TEST500D),                   ;ERROR DIRECTORY KEY
                DCS-CTR(C4,),                           ;COMPARE AT TARGET
        NEXT,   J/COMP500D                              ;
 (6746) DCS[1,00,1,0,0,0]  BM[1011,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,100,001]

   6141: ;(FREE)
   COMP500D:
        P3-T,   D_ASPLO[OVERLAP]-MINUS-BSPLO[OVERLAP],  ;COMPARE A=COUNT;B=COUNT
                SAVE-D[C],                              ;
        NEXT,   J/GOBUT500D                             ;
 (6141) DCS[0,00,0,0,0,0]  BM[1101,,00,10,,10,00,,000,,111,,,1,1,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,100,010]

   6142: ;(FREE)
   GOBUT500D:
        P0,     BUMP-VERIFY,                            ;COUNT
        SETUP,  RETURN/TEST500E,                        ;RETURN TO START OF NEXT SUBTEST
        NEXT,   GOTO-PAGE(7),                           ;BUT TABLE
                J/BUTO-IS-ZERO                          ;CHECK EQUALITY
 (6142) DCS[0,00,0,0,0,1]  BM[0110,,00,11,,10,11,,110,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,100,001]




   ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

   ;*** TEST 500E ***
   ;CHECK THAT "PREFETCH(0)H" WAS ONLY ASSERTED 357, TIMES/16384, DATA PATTERNS
   6736:
   TEST500E:
        P0,     LOAD-ENUA(ZTARGET402),                  ;SETUP FOR D=0 TEST
                LOAD-ERROR(TEST500E),                   ;ERROR DIRECTORY KEY
                DCS-CTR(C5,),                           ;COMPARE AT TARGET
        NEXT,   J/EXPEC500E                             ;
 (6736) DCS[1,00,1,0,0,0]  BM[1010,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,100,011]

   6143: ;(FREE)
   EXPEC500E:
        P0,     BUMP-VERIFY,                            ;
        P3,     CSPD[EXPEC]_EMIT,                       ;EXPECTED COUNT=
                EMIT/545,                               ; (545)=357,
        NEXT,   J/COMP500E                              ;
 (6143) DCS[0,00,0,0,0,1]  BM[0000,,10,00,,01,01,,100,,101,,,0,0,0,,0,,0,,,0,0000,,,1,,0000,0,,,11,000,,,001,100,100]

   6144: ;(FREE)
   COMP500E:
        P3-T,   D_ASPHI[PREFETCH]-MINUS-CSPB[EXPEC],    ;COMPARE RECEIVED:EXPECTED
                SAVE-D[C],                              ;
        NEXT,   J/GOBUT500E                             ;
 (6144) DCS[0,00,0,0,0,0]  BM[1101,,11,00,,11,00,,000,,111,,,1,1,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,100,101]

   6145: ;(FREE)
```



```
   GOBUT500E:
        SETUP,  RETURN/SCOPE500F,                       ;RETURN TO SCOPE LOOP TEST WORD
        NEXT,   GOTO-PAGE(7),                           ;BUT TABLE
                J/BUTO-IS-ZERO                          ;CHECK EQUALITY
 (6145) DCS[0,00,0,0,0,0]  BM[0110,,00,00,,11,00,,110,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,100,001]



   6146: ;(FREE)
   SCOPE500F:
        P0,     BUMP-VERIFY,                            ;COUNT
                BUSDIN_EMIT-[I],                        ;RESET PROC UCON
                EN-CLK-IR[15=00],                       ;
        NEXT,   BUTD[SCOPE],                            ;NO ERROR: "TEST503A" (+1, WORDS)
                J/TEST503A                              ;   ERROR: "SETFETCH500" (-27,WORDS)
 (6146) DCS[0,00,0,1,0,1]  BM[0000,,00,00,,00,01,,000,,100,,,0,0,0,,0,,0,,,1,1001,,,0,,0000,0,,,11,000,,,110,111,111]





   ;.PAGE**********************************************************************************************

   .TOC    *   TEST503-510: PROCESSOR UCON TESTS (FLAGS, FPS, PS, BUTN) & ASSOC LOGIC

   ;****************************************************************************
   ;*
   ;*      TESTS:  503A - 510F                            UWORDS: 214 + 245
   ;*
   ;*      FUNCTIONS:      TESTS 503A - 510F MANIPULATE THE VARIOUS PROCESSOR UCON
   ;*                      FUNCTIONS (FLAGS, EXFLAGS, FPS, PS, CUA) AND RELATED "BUT"
   ;*                      TESTS TO SEE THAT ALL ARE FUNCTIONAL,
   ;*
   ;****************************************************************************







   ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

   ;*** TEST 503 ***

   ;TESTS 503 A-I USE DATA PATTERNS OF:
   ;       FLAG<8:4,2:0>H = "10101010", EXFLAG<2:1>H = "01",
   ;       FPS<7:0>H = "1010 1010"

   ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

   ;*** TEST 503A ***
```

```
        ;LOAD FLAGS<8:4,2:0> WITH "10101010", EXFLAGS<2:1> WITH "01", FPS<7:4> WITH "1010",
        ;FPS<3:0> WITH "1010", AND READ BACK THRU "FLAGS&FPS" PORT OF PROCESSOR MUX
        6677:
        TEST503A:
                P0,     LOAD-ENUA(LOADNEW5),                    ;COMPARE 1/2 WAY THRU BK SUBR
                        LOAD-ERROR(TEST503A),                   ;ERROR DIRECTORY KEY
                        DCS-CTR(C10.),                          ;COMPARE IN ...
                        BUMP-VERIFY,                            ;COUNT
                NEXT,   J/LOADFLAGS03A                          ;
  (6677)  DCS[1,00,1,0,0,1]  BM[0101,,00,10,,00,11,,011,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,110,001,100]

        6614:
        LOADFLAGS03A:
                P2-U,   IR_EMIT,                                ;[125122]=NOT-FLTPT-N INSTR (NOTE UCON/CSPADDR OVLP)
                                                                ;ALSO NOTE IR=(NOT-PREFETCH), SO PREFETCH-SAVE WILL GET "0"
                                                                ;AFTER SUBSEQUENT JAMUPP (IN LOADFPSCC ROUTINE EXIT)
                P3,     CSPD(05)_EMIT,                          ;INITIAL VALUE:
                        EMIT/125122,                            ;FLAG<8:4,2:0>="10101010", EXFLAG<2:1>="01"
                NEXT,   J/LOADFPS503A                           ;"10101010 0101 0010"
  (6614)  DCS[0,00,0,0,0,0]  BM[1010,,10,10,,10,01,,010,,010,,,0,0,0,,0,,0,,,1,1010,,,1,,0000,0,,,11,000,,,001,100,111]

        6147:  ;(FREE)
        LOADFPS503A:
                P0,     BUMP-VERIFY,                            ;COUNT
                P3,     CSPD(06)_EMIT,                          ;INITIAL VALUE IN FPS<7:4>:
                        EMIT/052655,                            ;"01010101 1010 1101"
                NEXT,   J/EXPEC503A                             ;
  (6147)  DCS[0,00,0,0,0,1]  BM[0101,,10,01,,01,10,,101,,101,,,0,0,0,,0,,0,,,0,1001,,,1,,0000,0,,,11,000,,,001,101,000]

        6150:  ;(FREE)
        EXPEC503A:
                ;*** NOTE:  FLAG<8>, UBREAK ENABLE, GETS SET HERE,  KEEP SR=(600000)  (SET IN TEST500) TO FLAG
                ;        THAT ANY SPURIOUS UBREAKS ARE ILLEGAL (IE, CAUSING A JAMUPP SEQUENCE  ***
                P3,     CSPD(02)_EMIT,                          ;EXPECTED VALUE LOADS FLAGS, FPS<7:0>
                        EMIT/125252,                            ;OF "FLAGS&FPS" PORT
                NEXT,   J/MASK503A                              ;"10101010 1010 1010"
  (6150)  DCS[0,00,0,0,0,0]  BM[1010,,10,10,,10,10,,101,,010,,,0,0,0,,0,,0,,,0,1101,,,1,,0000,0,,,11,000,,,001,101,001]

        6151:  ;(FREE)
        MASK503A:
                P3,     CSPD(04)_EMIT,                          ;MASK OUT (TO ZEROS)
                        EMIT/177777,                            ;LOOK AT ALL THE BITS
                NEXT,   J/LOADFCC503A                           ;
  (6151)  DCS[0,00,0,0,0,0]  BM[1111,,10,11,,11,11,,111,,111,,,0,0,0,,0,,0,,,0,1011,,,1,,0000,0,,,11,000,,,001,101,010]

        6152:  ;(FREE)
        LOADFCC503A:
                P3,     CSPD(15)_EMIT,                          ;FPS<3:0> COME FROM CSP(15)<3:0>(MD)
                        EMIT/000012,                            ;"1010"
                NEXT,   J/DOFCC503A                             ;
  (6152)  DCS[0,00,0,0,0,0]  BM[0000,,10,00,,00,00,,001,,010,,,0,0,0,,0,,0,,,0,0010,,,1,,0000,0,,,11,000,,,001,101,011]

        6153:  ;(FREE)
        DOFCC503A:
```



```
                P3,     CSPD(00)_EMIT,                          ;CALL ON SUBR WHICH DOES THE LOAD
                        RETURN/TESTA503A,                       ;RETURN INLINE
                NEXT,   CALL(LOADFPSCC)                         ;
  (6153)  DCS[0,00,0,0,0,0]  BM[0110,,10,11,,10,01,,100,,111,,,0,0,0,,0,,0,,,0,1111,,,1,,0000,0,,,11,100,,,010,110,100]

        ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        ;NOW CHECK ALL THE RITE BITS WERE SET BY READING THEM BACK

        6714:
        TESTA503A:
                P0,     LOAD-ENUA(ZTARGET402),                  ;SETUP FOR D=ZERO COMPARE
                        LOAD-ERROR(TESTA503A),                  ;ERROR DIRECTORY KEY
                        DCS-CTR(C14.),                          ;COMPARE AT TARGET
                P3,     BUTA(CUA-TRACK),                        ;RESET CUA TRACKING AFTER JAMUPP
                NEXT,   J/GOPUT503A                             ;
  (6714)  DCS[1,00,1,0,0,0]  BM[0001,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,001,,,001,101,100]

        6154:  ;(FREE)
        GOPUT503A:
                SETUP,  RETURN/GOBUT503A,                       ;GO TO SUBR WHICH:
                NEXT,   CALL(FLAGFPSSEQLOD)                     ;1) CSP(05) -> FLAGS, EXFLAGS
  (6154)  DCS[0,00,0,0,0,0]  BM[0110,,00,00,,11,01,,101,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,000,111,011]
                                                                ;2) CSP(06) -> FPS<7:4>

        6155:  ;(FREE)
        GOBUT503A:
                SETUP,  RETURN/TEST503B,                        ;RETURN TO START OF NEXT SUBTEST
                NEXT,   CALL(FLAGFPSTOD)                        ; FLAGS&FPS.XOR.CSP(02) => D, BUT(D=0)
  (6155)  DCS[0,00,0,0,0,0]  BM[0110,,00,11,,00,11,,000,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,101,010]




        ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        ;*** TEST 503B ***
        ;DO THE "MULTIPLE BUT" ON "FLAG7-H" TO CHECK IT'S CLEAR
        6630:
        TEST503B:
                P0,     LOAD-ENUA(ZTARGET406),                  ;BIT<00> CLEAR
                        LOAD-ERROR(TEST503B),                   ;ERROR DIRECTORY KEY
                        DCS-CTR(C4.),                           ;COMPARE AT TARGET
                NEXT,   J/GOBUT503B                             ;
  (6630)  DCS[1,00,1,0,0,0]  BM[1011,,00,11,,11,00,,000,,110,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,101,110]

        6156:  ;(FREE)
        GOBUT503B:
                SETUP,  RETURN/TEST503C,                        ;RETURN TO START OF NEXT SUBTEST
                NEXT,   GOTO-PAGE(7),                           ;BUT TABLE
                        J/BUTWFLAG7                             ;FLAG 7-H IN BIT<00>
```

```
     (6156)  DCS[0,00,0,0,0,0]  BM[0110,,00,11,,00,11,,010,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,010,101]




    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

    ;*** TEST 503C ***
    ;DO THE "BUT" ON "FLTPT-PROC-H"*"FLAGS-H*EXFLAG2-L" TO CHECK IT'S CLEAN
    6632:
    TEST503C:
            P0,     LOAD-ENUA(ZTARGET402),              ;BIT<00> CLEAR
                    LOAD-ERROR(TEST503C),               ;ERROR DIRECTORY KEY
                    DCS-CTR(C3,),                       ;COMPARE AT TARGET
                    BUMP-VERIFY,                        ;COUNT
            NEXT,   J/GOBUT503C                         ;
     (6632)  DCS[1,00,1,0,0,1]  BM[1100,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,101,111]

    6157:  ;(FREE)
    GOBUT503C:
            SETUP,  RETURN/TEST503DA,                   ;RETURN TO START OF NEXT SUBTEST
            NEXT,   GOTO-PAGE(7),                       ;BUT TABLE
                    J/BUTFPPROC                         ;FLAGS-H*EXFLAG2-L IN BIT<00>
     (6157)  DCS[0,00,0,0,0,0]  BM[0110,,00,11,,00,00,,001,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,101,100]




    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

    ;*** TEST 503DA ***
    ;DO THE "MULTIPLE BUT" ON "FLTPT" TO CHECK IT'S CLEAR, IR=[125122], NOT-FLTPT-H INSTR
    6601:
    TEST503DA:
            P0,     LOAD-ENUA(ZTARGET406),              ;BIT<00> CLEAR
                    LOAD-ERROR(TEST503DA),              ;ERROR DIRECTORY KEY
                    DCS-CTR(C4,),                       ;COMPARE AT TARGET
            NEXT,   J/GOBUT503DA                        ;
     (6601)  DCS[1,00,1,0,0,0]  BM[1011,,00,11,,11,00,,000,,110,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,110,000]

    6160:  ;(FREE)
    GOBUT503DA:
            SETUP,  RETURN/TEST503D,                    ;RETURN TO START OF NEXT SUBTEST
            NEXT,   GOTO-PAGE(7),                       ;BUT TABLE
                    J/BUTNFLTPTS                        ;FLTPT-H IN BIT<00>
     (6160)  DCS[0,00,0,0,0,0]  BM[0110,,00,11,,01,11,,000,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,011,001]




    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
```



```

    ;*** TEST 503D ***
    ;DO AN INSTR-1 FLOATING POINT DECODE, TO CHECK THAT FLAG<4:5> => BIT<1:0>
    6670:
    TEST503D:
            P0,     LOAD-ENUA(ZTARGET476),              ;INSTR-1 FLTPT, BIT<1:0>="10"
                    LOAD-ERROR(TEST503D),               ;ERROR DIRECTORY KEY
                    DCS-CTR(C4,),                       ;COMPARE AT TARGET
            NEXT,   J/LOADIR503D
     (6670)  DCS[1,00,1,0,0,0]  BM[1011,,00,11,,11,00,,111,,110,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,110,001]

    6161:  ;(FREE)
    LOADIR503D:
            P0,     BUMP-VERIFY,                        ;COUNT
            P2-U,   IR_EMIT,                            ;IR <- FLTPT INSTR
                    EMIT/175252,                        ;
            NEXT,   J/GOBUT503D                         ;
     (6161)  DCS[0,00,0,0,0,1]  BM[1111,,00,10,,10,10,,101,,010,,,0,0,0,,0,,0,,,1,1010,,,0,,0000,0,,,11,000,,,001,110,010]

    6162:  ;(FREE)
    GOBUT503D:
            SETUP,  RETURN/TEST503E,                    ;RETURN TO START OF NEXT SUBTEST
            NEXT,   GOTO-PAGE(7),                       ;BUT TABLE
                    J/BUTINSTR1                         ;FULL WIDTH
     (6162)  DCS[0,00,0,0,0,0]  BM[0110,,00,11,,10,00,,000,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,000,110]




    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

    ;*** TEST 503E ***
    ;READ EXFLAGS<2:1> = "01" IN CUA-PORT<2:1>,
    ;ALSO "PREFETCH-SAVE-H" = "0"  FROM PREVIOUS SETUP
    6700:
    TEST503E:
            P0,     LOAD-ENUA(ZTARGET402),              ;SETUP FOR D=0 TEST
                    LOAD-ERROR(TEST503E),               ;ERROR DIRECTORY KEY
                    DCS-CTR(C10,),                      ;COMPARE AT TARGET
            NEXT,   J/EXPEC503E                         ;
     (6700)  DCS[1,00,1,0,0,0]  BM[0101,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,110,011]

    6163:  ;(FREE)
    EXPEC503E:
            P0,     BUMP-VERIFY,                        ;COUNT
            P1,     CSPD[02]_EMIT,                      ;CUA PORT READS AS:

                    EMIT/073732,                        ;D=CUA<11:00>=EXFLAG<2:1>=PREFETCH
            NEXT,   J/GOPUT503E                         ;
     (6163)  DCS[0,00,0,0,0,1]  BM[0111,,10,01,,11,11,,011,,010,,,0,0,0,,0,,0,,,0,1101,,,1,,0000,0,,,11,000,,,001,110,100]

    6164:  ;(FREE)
    GOPUT503E:
            SETUP,  RETURN/TEST503F,                    ;GO TO SUBR WHICH:
```

```
        NEXT,   CALL(CUATOD)                                 ;PUTS CUASEXFLAGS.XOR.CSP(02) => D, BUT(D=ZERO)
(6164)  DCS[0,00,0,0,0,0]  BN[0110,,00,11,,10,01,,000,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,101,111]




    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

    ;*** TEST 503F ***
    ;DO THE "MULTIPLE BUT" ON EXFLAG<2> TO CHECK IT'S CLEAR
    6710:
    TEST503F:
        PO,     LOAD-ENUA(ZTARGET406),                       ;BIT<00> CLEAR
                LOAD-ERROR(TEST503F),                        ;ERROR DIRECTORY KEY
                DCS-CTR(C4,),                                ;COMPARE AT TARGET
        NEXT,   J/GOBUT503F                                  ;
(6710)  DCS[1,00,1,0,0,0]  BN[1011,,00,11,,11,00,,000,,110,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,110,101]

    6165:   ;(FREE)
    GOBUT503F:
        SETUP,  RETURN/TEST503G,                             ;RETURN TO START OF NEXT SUBTEST
        NEXT,   GOTO-PAGE(7),                                ;BUT TABLE
                J/BUTNEXFLAG2                                ;EXFLAG<2>N IN BIT<00>
(6165)  DCS[0,00,0,0,0,0]  BN[0110,,00,11,,10,10,,000,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,011,011]




    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

    ;*** TEST 503G ***
    ;DO THE "MULTIPLE BUT" ON EXFLAG<1> TO CHECK IT'S SET
    6720:
    TEST503G:
        PO,     LOAD-ENUA(ZTARGET407),                       ;BIT<00> SET
                LOAD-ERROR(TEST503G),                        ;ERROR DIRECTORY KEY
                DCS-CTR(C4,),                                ;COMPARE AT TARGET
                BUMP-VERIFY,                                 ;COUNT
        NEXT,   J/GOBUT503G                                  ;
(6720)  DCS[1,00,1,0,0,1]  BN[1011,,00,11,,11,00,,000,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,110,110]

    6166:   ;(FREE)
    GOBUT503G:
        SETUP,  RETURN/TEST503H,                             ;RETURN TO START OF NEXT SUBTEST
        NEXT,   GOTO-PAGE(7),                                ;BUT TABLE
                J/BUTNEXFLAG1                                ;EXFLAG<1>N IN BIT<00>
(6166)  DCS[0,00,0,0,0,0]  BN[0110,,00,11,,10,11,,010,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,010,111]
```



```
    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

    ;*** TEST 503H ***
    ;CHECK FPS<5> SET, VIA BUTR
    6732:
    TEST503H:
        PO,     LOAD-ENUA(ZTARGET407),                       ;BIT<00> SET
                LOAD-ERROR(TEST503H),                        ;ERROR DIRECTORY KEY
                DCS-CTR(C3,),                                ;COMPARE AT TARGET
                BUMP-VERIFY,                                 ;COUNT
        NEXT,   J/GOBUT503H                                  ;
(6732)  DCS[1,00,1,0,0,1]  BN[1100,,00,11,,11,00,,000,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,110,111]

    6167:   ;(FREE)
    GOBUT503H:
        SETUP,  RETURN/TEST503I,                             ;RETURN TO START OF NEXT SUBTEST
        NEXT,   GOTO-PAGE(7),                                ;BUT TABLE
                J/BUTFPS05                                   ;FPS <05> IN BIT<00>
(6167)  DCS[0,00,0,0,0,0]  BN[0110,,00,11,,11,00,,010,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,001,010]




    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

    ;*** TEST 503I ***
    ;CHECK FLTPT-FD-H = F(FPS<7:6>,FLAG<5,2,1,EX:5>) IS CLEAR
    6742:
    TEST503I:
        PO,     LOAD-ENUA(ZTARGET403),                       ;BIT<00> CLEAR
                LOAD-ERROR(TEST503I),                        ;ERROR DIRECTORY KEY
                DCS-CTR(C3,),                                ;COMPARE AT TARGET
        NEXT,   J/GOBUT503I                                  ;
(6742)  DCS[1,00,1,0,0,0]  BN[1100,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,111,000]

    6170:   ;(FREE)
    GOBUT503I:
        PO,     BUMP-VERIFY,                                 ;COUNT
        SETUP,  RETURN/TEST503J,                             ;RETURN TO START OF NEXT SUBTEST
        NEXT,   GOTO-PAGE(7),                                ;BUT TABLE
                J/BUTFPFD                                    ;FLTPT-FD-H IN BIT<00>
(6170)  DCS[0,00,0,0,0,1]  BN[0110,,00,11,,11,01,,110,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,111,001]

    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

    ;*** TEST 503J ***

    ;DO THE "MULTIPLE BUT" ON "D<00>" TO CHECK IT'S CLEAR  (D WAS ZEROED IN TEST503E, ABOVE)
    6756:
    TEST503J:
        PO,     LOAD-ENUA(ZTARGET406),                       ;BIT<00> CLEAR
                LOAD-ERROR(TEST503J),                        ;ERROR DIRECTORY KEY
                DCS-CTR(C4,),                                ;COMPARE AT TARGET
```

```
             NEXT,    J/GOBUT503J                                          ;
     (6756)  DCS[1,00,1,0,0,0]  BM[1011,,00,11,,11,00,,000,,110,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,111,001]

     6171:  ;(FREE)
     GOBUT503J:
             SETUP,   RETURN/TEST503K,                                     ;RETURN TO START OF NEXT SUBTEST
             NEXT,    GOTO-PAGE(7),                                        ;BUT TABLE
                      J/BUTMD00                                            ;D<00>=K IN BIT<00>
     (6171)  DCS[0,00,0,0,0,0]  BM[0110,,00,11,,01,00,,110,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,010,001]


     ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

     ;*** TEST 503K ***
     ;DO THE "MULTIPLE BUT" ON "FLTPT" TO CHECK IT'S SET, IR=(1752B3), FLTPT-K INSTR
     6646:
     TEST503K:
             P0,      LOAD-ENUA(ZTARGET407),                               ;BIT<00> SET
                      LOAD-ERROR(TEST503K),                                ;ERROR DIRECTORY KEY
                      DCS-CTR(C4,),                                        ;COMPARE AT TARGET
                      BUMP-VERIFY,                                         ;COUNT
             NEXT,    J/GOBUT503K                                          ;
     (6646)  DCS[1,00,1,0,0,1]  BM[1011,,00,11,,11,00,,000,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,111,010]

     6172:  ;(FREE)
     GOBUT503K:
             SETUP,   RETURN/SCOPE503,                                     ;RETURN TO SCOPE LOOP TEST WORD
             NEXT,    GOTO-PAGE(7),                                        ;BUT TABLE
                      J/BUTKFLTPTS                                         ;FLTPT-K IN BIT<00>
     (6172)  DCS[0,00,0,0,0,0]  BM[0110,,00,00,,11,11,,011,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,011,001]






     6173:  ;(FREE)
     SCOPE503:
             P0,      BUSDIN_EMIT-[I],                                     ;RESET PROC UCONS
                      EN-CLK-IR[15-00],                                    ;
             NEXT,    BUTD[SCOPE],                                         ;NO ERROR: "TEST504A" (+1, WORDS)
                      J/TEST504A                                           ;   ERROR: "LOADFLAGS03A" (+34, WORDS)
     (6173)  DCS[0,00,0,1,0,0]  BM[0000,,00,00,,00,01,,000,,100,,,0,0,0,,0,,0,,,1,1001,,,0,,0000,0,,,11,000,,,110,001,101]
```



```
     ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

     ;*** TEST 504 ***

     ;TESTS 504 A-I USE DATA PATTERNS OF:
     ;        FLAG<8:4,2:0>H = "01010101", EXFLAG<2:1>H = "10",
     ;        FPS<7:0>H = "0101 0101"

     ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

     ;*** TEST 504A ***
     ;LOAD FLAGS<8:4,,2:0> WITH "01010101", EXFLAGS<2:1> WITH "10", FPS<7:4> WITH "0101",
     ;FPS<3:0> WITH "0101", AND READ BACK THRU "FLAGS&FPS" PORT OF PROCESSOR MUX
     6615:
     TEST504A:
             P0,      LOAD-ENUA(4777),                                     ;SETUP FOR COMPARE 1/2 WAY + 1 THRU BM SUBR
                      LOAD-ERROR(TEST504A),                                ;ERROR DIRECTORY KEY
                      DCS-CTR(C11,),                                       ;COMPARE AT ...
             NEXT,    J/LOADFLAGS04A                                       ;
     (6615)  DCS[1,00,1,0,0,0]  BM[0100,,00,10,,01,11,,111,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,110,010,010]

     6627:
     LOADFLAGS04A:
             P2-U,    IP_EMIT,                                             ;ALSO NOTE IR=(NOT-PREFETCH), SO PREFETCH-SAVE WILL GET "0"
                                                                           ;AFTER SUBSEQUENT JAMUPP (IN LOADFPSCC ROUTINE EXIT)
             P1,      CSPD[05]_EMIT,                                       ;INITIAL VALUE:
                      EMIT/052644,                                         ;FLAG<8:4,2:0>="01010101", EXFLAG<2:1>="10"
             NEXT,    J/LOADFPS504A                                        ;"01010101 1010 0100"
     (6627)  DCS[0,00,0,0,0,0]  BM[0101,,10,01,,01,10,,100,,100,,,0,0,0,,0,,0,,,1,1010,,,1,,0000,0,,,11,000,,,001,111,100]

     6174:  ;(FREE)
     LOADFPS504A:
             P3,      CSPD[06]_EMIT,                                       ;INITIAL VALUE IN FPS<7:4>:
                      EMIT/125133,                                         ;"10101010 0101 1011"
             NEXT,    J/EXPEC504A                                          ;
     (6174)  DCS[0,00,0,0,0,0]  BM[1010,,10,10,,10,01,,011,,011,,,0,0,0,,0,,0,,,0,1001,,,1,,0000,0,,,11,000,,,001,111,101]

     6175:  ;(FREE)
     EXPEC504A:
             P3,      CSPD[02]_EMIT,                                       ;EXPECTED VALUE LOADS FLAGS, FPS<7:0>
                      EMIT/052525,                                         ;OF "FLAGS&FPS" PORT
             NEXT,    J/LOADFCC504A                                        ;"01010101 0101 0101"
     (6175)  DCS[0,00,0,0,0,0]  BM[0101,,10,01,,01,01,,010,,101,,,0,0,0,,0,,0,,,0,1101,,,1,,0000,0,,,11,000,,,001,111,110]

     6176:  ;(FREE)
     LOADFCC504A:

             P3,      CSPD[15]_EMIT,                                       ;FPS<3:0> COME FROM CSP(15)<3:0>(ND)
                      EMIT/000005,                                         ;"0101"
             NEXT,    J/DOFCC504A                                          ;
     (6176)  DCS[0,00,0,0,0,0]  BM[0000,,10,00,,00,00,,000,,101,,,0,0,0,,0,,0,,,0,0010,,,1,,0000,0,,,11,000,,,001,111,111]

     6177:  ;(FREE)
     DOFCC504A:
             P3,      CSPD[00]_EMIT,                                       ;CALL BM SUBR WHICH DOES THE LOAD
```

```
                        RETURN/TEST&B04A,                       !RETURN INLINE
                NEXT,   CALL[LOADFPSCC]                         !
        (6177)  DCS[0,00,0,0,0,0]  BM[0110,,00,11,,11,10,,110,,111...0,0,0,,0,,0,,,0,1111,,,1,,0000,0,,,11,100,,,010,110,100]

        ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        !NOW CHECK ALL THE BITS WERE SET BY READING THEM BACK

        67661
        TEST&504A:
                PO,     LOAD=ENUA(%TARGET402),                  !SETUP FOR D=ZERO TEST
                        LOAD=ERROR(TEST&504A),                  !ERROR DIRECTORY KEY
                        DCS=CTR(C),,                            !COMPARE AT TARGET
                        BUMP=VERIFY,                            !COUNT
                P3,     BUTA(CUA=TRACK),                        !RESET CUA TRACKING AFTER JAKUPP
                NEXT,   J/GOBUT&04A                             !
        (6766)  DCS[1,00,1,0,0,1]  BM[0001,,00,11,,11,00,,000,,010...0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,001,,,010,000,000]

        6200:   ;(FREE)
        GOBUT504A:
                SETUP,  RETURN/GOBUT504A,                       !GO TO SUBR WHICH:
                NEXT,   CALL[FLAGFP&&EGLOD]                     !1) CSP(00) -> FLAGS, EXFLAGS
        (6200)  DCS[0,00,0,0,0,0]  BM[0110,,00,01,,00,00,,001,,111...0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,000,111,011]
                                                                !2) CSP(06) -> FP&<7:4>

        6201:   ;(FREE)
        GOBUT504A:
                SETUP,  RETURN/TEST504B,                        !RETURN TO START OF NEXT SUBTEST
                NEXT,   CALL[FLAGFP&TOD]                        ! FLAG&&FP&,XOR,CSP(02) -> D; BUT(D=0)
        (6201)  DCS[0,00,0,0,0,0]  BM[0110,,00,11,,10,11,,000,,111...0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,101,010]

        ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        !*** TEST 504B ***
        !DO THE "MULTIPLE BUT" ON "FLAG7-H" TO CHECK IT'S SET
        67301
        TEST504B:
                PO,     LOAD=ENUA(%TARGET407),                  !BIT<00> SET
                        LOAD=ERROR(TEST504B),                   !ERROR DIRECTORY KEY
                        DCS=CTR(C4,),                           !COMPARE AT TARGET
                NEXT,   J/GOBUT504B                             !
        (6730)  DCS[1,00,1,0,0,0]  BM[1011,,00,11,,11,00,,000,,111...0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,010,000,010]

        6202:   ;(FREF)
        GOBUT504B:
                SETUP,  RETURN/TEST504C,                        !RETURN TO START OF NEXT SUBTEST
                NEXT,   GOTO=PAGE(7),                           !BUT TABLE
                        J/BUTMFLAG7                             !FLAG7-H IN BIT<00>
```



```
        (6702)  DCS[0,00,0,0,0,0]  BM[0110,,00,11,,11,00,,000,,111...0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,010,101]

        ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        !*** TEST 504C ***
        !DO THE "BUT" ON "FLTPT-PROC-H"="FLAG<5>-H*EXFLAG<1>-L" TO CHECK IT'S SET
        6740:
        TEST504C:
                PO,     LOAD=ENUA(%TARGET403),                  !BIT<00> SET
                        LOAD=ERROR(TEST504C),                   !ERROR DIRECTORY KEY
                        DCS=CTR(C3,),                           !COMPARE AT TARGET
                        BUMP=VERIFY,                            !COUNT
                NEXT,   J/GOBUT504C                             !
        (6740)  DCS[1,00,1,0,0,1]  BM[1100,,00,11,,11,00,,000,,011...0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,010,000,011]

        6203:   ;(FREE)
        GOBUT504C:
                SETUP,  RETURN/TEST504D,                        !RETURN TO START OF NEXT SUBTEST
                NEXT,   GOTO=PAGE(7),                           !BUT TABLE
                        J/BUTFPPROC                             !FLAG5-H*EXFLAG1-L IN BIT<00>
        (6203)  DCS[0,00,0,0,0,0]  BM[0110,,00,11,,11,01,,000,,111...0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,101,100]

        ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        !*** TEST 504D ***
        !DO AN INSTR=1 FLOATING POINT DECODE, TO CHECK THAT FLAG<4:5> -> BIT<1:0>
        6750:
        TEST504D:
                PO,     LOAD=ENUA(%TARGET475),                  !INSTR=1 FLTPT, BIT<1:0>="01"
                        LOAD=ERROR(TEST504D),                   !ERROR DIRECTORY KEY
                        DCS=CTR(C4,),                           !COMPARE AT TARGET
                NEXT,   J/LOADIR504D                            !
        (6750)  DCS[1,00,1,0,0,0]  BM[1011,,00,11,,11,00,,111,,101...0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,010,000,100]

        6204:   ;(FREE)
        LOADIR504D:
                PO,     BUMP=VERIFY,                            !COUNT
                P2=U,   IR_EMIT,                                !IR <- FLTPT INSTR
                        EMIT/172528,                            !
                NEXT,   J/GOBUT504D                             !

        (6204)  DCS[0,00,0,0,0,1]  BM[1111,,00,01,,01,01,,010,,101...0,0,0,,0,,0,,,1,1010,,,0,,0000,0,,,11,000,,,010,000,101]

        6205:   ;(FREF)
        GOBUT504D:
                SETUP,  RETURN/TEST504E,                        !RETURN TO START OF NEXT SUBTEST
                NEXT,   GOTO=PAGE(7),                           !BUT TABLE
                        J/BUTINSTR;                             !FULL WIDTH
```

```
(6205)  DCS[0,00,0,0,0,0]  BM[0110,,00,11,,11,10,,000,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,000,110]



        ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        !*** TEST 504E ***
        !READ EXFLAGS<2:1>="10" IN CUA-PORT<2:1>
        !ALSO "PREFETCH-SAVE-H"="0" FROM PREVIOUS SETUP
        6760:
        TEST504E:
                P0,     LOAD-ENUA(ZTARGET402),                  !SETUP FOR D=0 TEST
                        LOAD-ERROR(TEST504E),                   !ERROR DIRECTORY KEY
                        DCS-CTR(C10,),                          !COMPARE AT TARGET
                NEXT,   J/EXPEC504E                             !
(6760)  DCS[1,00,1,0,0,0]  BM[0101,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,010,000,110]

        6206:  !(FREE)
        EXPEC504E:
                P0,     BUMP-VERIFY,                            !COUNT
                P3,     CSPD[02]_EMIT,                          !CUA PORT READS AS:
                        EMIT/073734,                            !0&CUA<1:0>&EXFLAG<2:1>&PREFETCH
                NEXT,   J/GOPUT504E                             !
(6206)  DCS[0,00,0,0,0,1]  BM[0111,,10,01,,11,11,,011,,100,,,0,0,0,,0,,0,,,0,1101,,,1,,0000,0,,,11,000,,,010,000,111]

        6207:  !(FREE)
        GOPUT504E:
                SETUP,  RETURN/TEST504F,                        !GO TO SUBR WHICH:
                NEXT,   CALL[CUATOD]                            !PUTS CUA&EXFLAGS.XOR.CSP(02) => D, BUT(D=ZERO)
(6207)  DCS[0,00,0,0,0,0]  BM[0110,,00,11,,01,11,,010,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,101,111]




        ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        !*** TEST 504F ***
        !DO THE "MULTIPLE BUT" ON EXFLAG<2> TO CHECK IT'S SET
        6672:
        TEST504F:
                P0,     LOAD-ENUA(ZTARGET407),                  !BIT<00> SET
                        LOAD-ERROR(TEST504F),                   !ERROR DIRECTORY KEY
                        DCS-CTR(C4,),                           !COMPARE AT TARGET
                NEXT,   J/GOBUT504F                             !
(6672)  DCS[1,00,1,0,0,0]  BM[1011,,00,11,,11,00,,000,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,010,001,000]

        6210:  !(FREE)
        GOBUT504F:
                SETUP,  RETURN/TEST504G,                        !RETURN TO START OF NEXT SUBTEST
                NEXT,   GOTO-PAGE(7),                           !BUT TABLE
                        J/BUTNEXFLAG2                           !EXFLAG<2>N IN BIT 0
```



```
(6210)  DCS[0,00,0,0,0,0]  BM[0110,,00,11,,10,00,,010,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,011,011]




        ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        !*** TEST 504G ***
        !DO THE "MULTIPLE BUT" ON EXFLAG<1> TO CHECK IT'S CLEAR
        6702:
        TEST504G:
                P0,     LOAD-ENUA(ZTARGET406),                  !BIT<00> CLEAR
                        LOAD-ERROR(TEST504G),                   !ERROR DIRECTORY KEY
                        DCS-CTR(C4,),                           !COMPARE AT TARGET
                        BUMP-VERIFY,                            !COUNT
                NEXT,   J/GOBUT504G                             !
(6702)  DCS[1,00,1,0,0,1]  BM[1011,,00,11,,11,00,,000,,110,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,010,001,001]

        6211:  !(FREE)
        GOBUT504G:
                SETUP,  RETURN/TEST504H,                        !RETURN TO START OF NEXT SUBTEST
                NEXT,   GOTO-PAGE(7),                           !BUT TABLE
                        J/BUTNEXFLAG1                           !EXFLAG<1>N IN BIT 0
(6211)  DCS[0,00,0,0,0,0]  BM[0110,,00,11,,10,00,,110,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,010,111]




        ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        !*** TEST 504H ***
        !CHECK FPS<5> CLEAR, VIA BUTP
        6706:
        TEST504H:
                P0,     LOAD-ENUA(ZTARGET406),                  !BIT<00> CLEAR
                        LOAD-ERROR(TEST504H),                   !ERROR DIRECTORY KEY
                        DCS-CTR(C3,),                           !COMPARE AT TARGET
                        BUMP-VERIFY,                            !COUNT
                NEXT,   J/GOBUT504H                             !
(6706)  DCS[1,00,1,0,0,1]  BM[1100,,00,11,,11,00,,000,,110,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,010,001,010]

        6212:  !(FREE)
        GOBUT504H:
                SETUP,  RETURN/TEST504I,                        !RETURN TO START OF NEXT SUBTEST
                NEXT,   GOTO-PAGE(7),                           !BUT TABLE
                        J/BUTFPS05                              !FPS<05> IN BIT<00>

(6212)  DCS[0,00,0,0,0,0]  BM[0110,,00,11,,10,01,,110,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,001,010]
```

```
	; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

	;*** TEST 504I ***
	;CHECK FLTPT-FD-H = F(FPS<7:6>,FLAG<5,2,1,EX1>) IS SET
	6716:
	TEST504I:
		P0,	LOAD-ENUA(ZTARGET403),			;BIT<00> SET
			LOAD-ERROR(TEST504I),			;ERROR DIRECTORY KEY
			DCS-CTR(C3,),				;COMPARE AT TARGET
		NEXT,	J/GOBUT504I				;
(6716)	DCS[1,00,1,0,0,0]	BM[1100,,00,11,,11,00,,000,,011,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,010,001,011]

	6213:	;(FREE)
	GOBUT504I:
		P0,	BUMP-VERIFY,				;COUNT
		SETUP,	RETURN/SCOPE504,			;RETURN TO SCOPE LOOP TEST WORD
		NEXT,	GOTO-PAGE(7);				;BUT TABLE
			J/BUTFPFD				;FLTPT-FD-H IN BIT<00>
(6213)	DCS[0,00,0,0,0,1]	BM[0110,,00,01,,00,01,,100,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,111,001]



	6214:	;(FREE)
	SCOPE504:
		P0,	BUSDIN_EMIT-[I],			;RESET PROC UCON
			EN-CLK-IR[15-00],			;
		NEXT,	BUTD[SCOPE],				;NO ERROR: "TEST505A" (+1, WORDS)
			J/TEST505A				;   ERROR: "LOADFLAG504A" (-27, WORDS)
(6214)	DCS[0,00,0,1,0,0]	BM[0000,,00,00,,00,01,,000,,100,,,0,0,0,,0,,0,,,1,1001,,,0,,0000,0,,,11,000,,,110,010,011]










	; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

	;*** TEST 505 ***

	;TESTS 505 A-C USE DATA PATTERNS OF:
	;	FLAG<8:4,2:0>H = "11111000", EXFLAG<2:1>H = "10",
	;	FPS<7:0>H = "0101 0101"

	; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

	;*** TEST 505A ***
	;CHECK THAT BUTA(CLEAR-FLAG-RES-UCON) ONLY CLEARS SHORT-TERM FLAGS
	6623:
	TEST505A:
		P0,	LOAD-ENUA(ZTARGET402),			;SETUP FOR D=ZERO TEST
```



```
			LOAD-ERROR(TEST505A),			;ERROR DIRECTORY KEY
			DCS-CTR(C15,),				;COMPARE AT TARGET
		NEXT,	J/LOAD505A				;
(6623)	DCS[1,00,1,0,0,0]	BM[0000,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,110,010,000]

	6620:
	LOAD505A:
		P3,	CSPD[05]_EMIT,				;LOAD FLAGS<8:4,2:0>,
			EMIT/177406,				;EXFLAG<2:1> WITH ALL 1-S
		NEXT,	J/EXPEC505A				;
(6620)	DCS[0,00,0,0,0,0]	BM[1111,,10,11,,11,00,,000,,110,,,0,0,0,,0,,0,,,0,1010,,,1,,0000,0,,,11,000,,,010,001,101]

	6215:	;(FREE)
	EXPEC505A:
		P0,	BUMP-VERIFY,				;COUNT
		P3,	CSPD[02]_EMIT,				;AFTER BUTA(CLR-FLAG-,,,), EXPECT THIS IN FLAGS:FPS PORT:
			EMIT/174125,				;"11111000 0101 0101"
		NEXT,	J/SETFLAGS05A				;
(6215)	DCS[0,00,0,0,0,1]	BM[1111,,10,10,,00,01,,010,,101,,,0,0,0,,0,,0,,,0,1101,,,1,,0000,0,,,11,000,,,010,001,110]

	6216:	;(FREE)
	SETFLAGS05A:
		P0,	BUMP-VERIFY,				;COUNT
		SETUP,	RETURN/BUTCLR505A,			;GO TO SUBR WHICH:
		NEXT,	CALL[FLAGLOD]				;PUTS CSP(05) INTO THE FLAGS
(6216)	DCS[0,00,0,0,0,1]	BM[0110,,00,01,,00,01,,111,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,000,111,110]

	6217:	;(FREE)
	BUTCLR505A:
		P0,	DCS-CTR[C9,],				;POINT COUNTER AT TARGET
		P3,	BUTA(CLR-FLAG-RES-UCON),		;CLEAR SHORT-TERM FLAGS
		NEXT,	J/GOPUT505A				;
(6217)	DCS[0,00,1,0,0,0]	BM[0110,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,010,,,010,010,000]

	6220:	;(FREE)
	GOPUT505A:
		SETUP,	RETURN/TEST505B,			;GO TO SUBR WHICH:
		NEXT,	CALL[FLAGFPSTOD]			;FLAGS:FPS,XOR,CSP(2) -> D, BUT(D=ZERO)
(6220)	DCS[0,00,0,0,0,0]	BM[0110,,00,11,,10,01,,010,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,101,010]





	; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

	;*** TEST 505B ***

	;CHECK EXFLAG<2> NOT CLEARED, VIA BUTN
	6712:
	TEST505B:
		P0,	LOAD-ENUA(ZTARGET407),			;BIT<00> SET
			LOAD-ERROR(TEST505B),			;ERROR DIRECTORY KEY
			DCS-CTR(C4,),				;COMPARE AT TARGET
```

```
                BUMP-VERIFY,                        !COUNT
        NEXT,   J/GOBUTS05B                         !
 (6712) DCS[1,00,1,0,0,1]  BM[1011,,00,11,,11,00,,000,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,010,010,001]

    6221: !(FREE)
    GOBUTS05B:
        SETUP,  RETURN/TESTS05C,                    !RETURN TO START OF NEXT SUBTEST
        NEXT,   GOTO-PAGE(7),                       !BUT TABLE
                J/BUTHEXFLAG2                       !EXFLAG<2> IN BIT<00>
 (6221) DCS[0,00,0,0,0,0]  BM[0110,,00,11,,10,10,,010,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,011,011]




    ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

    !*** TEST 505C ***
    !CHECK EXFLAG<1> WAS CLEARED, VIA BUTH
    6722:
    TESTS05C:
        P0,     LOAD-ENUA(ETARGET406),              !BIT<00> CLEAR
                LOAD-ERROR(TESTS05C),               !ERROR DIRECTORY KEY
                DCS-CTR(C4,),                       !COMPARE AT TARGET
        NEXT,   J/GOBUTS05C                         !
 (6722) DCS[1,00,1,0,0,0]  BM[1011,,00,11,,11,00,,000,,110,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,010,010,011]

    6223: !(FREE)
    GOBUTS05C:
        SETUP,  RETURN/SCOPE505,                    !RETURN TO SCOPE LOOP TEST WORD
        NEXT,   GOTO-PAGE(7),                       !BUT TABLE
                J/BUTHEXFLAG1                       !EXFLAG<1> IN BIT<00>
 (6223) DCS[0,00,0,0,0,0]  BM[0110,,00,01,,00,10,,100,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,010,111]

    6224: !(FREE)
    SCOPE505:
        P0,     BUMP-VERIFY,                        !COUNT
                BUSDIN_EMIT-[I],                    !EMIT FOR CONSTANTS
        P3,     FLAG[8-0]_D[15-8]-[I],              !ZERO ALL FLAGS, D WAS LEFT
                FPS[7-4]_D[7-4]-[I],                !  ZERO FROM TESTS05A, IF ALL OK,
        NEXT,   BUTD[SCOPE],                        !NO ERROR: "TESTS06A" (+1, WORDS)
                J/TESTS06A                          !   ERROR: "LOADS06A" (-9, WORDS)
 (6224) DCS[0,00,0,1,0,1]  BM[0000,,00,00,,00,01,,100,,001,,,0,0,0,,0,,0,,,1,1011,,,0,,0000,0,,,11,000,,,110,010,001]






    ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

    !*** TEST 506 ***

```



```
    !TESTS 506 A-E USE A "0-1-1010-1010" PATTERN IN PS<15,13,7:4,3:0>

    ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

    !* TEST 506A *
    !LOAD UP PS<HI,MID,LO> IN ORDER, READ BACK THRU PS PORT OF PROC MUX
    !  ALSO CHECK THAT BUTA(CLR-FLAG-RES-UCON) CLEARS UCON REGISTER, SO THAT:
    !     (1) BUSDIN_EMIT IS SELECTED, VIA UCON-SELECT(1)H=LOW, AND
    !     (2) MAKE SURE THAT THE OTHER UCON BIT LATCHES ARE ALSO CLEARED
    6621:
    TESTS06A:
        P0,     LOAD-ENUA(ETARGET402),              !SETUP FOR D=ZERO TEST
                LOAD-ERROR(TESTS06A),               !ERROR DIRECTORY KEY
                DCS-CTR(C15,),                      !STALL FOR NOW
        NEXT,   J/EXPECS06A                         !
 (6621) DCS[1,00,1,0,0,0]  BM[0000,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,110,001,010]

    6612:
    EXPECS06A:
        P3,     CSPD[02]_EMIT,                      !EXPECTED VALUE TO BE READ OUT OF
                EMIT/030252,                        !PS AFTER LOADING:
        NEXT,   J/SETUCONS06A                       !"0011 0000 1010 1010"
 (6612) DCS[0,00,0,0,0,0]  BM[0011,,10,00,,00,10,,101,,010,,,0,0,0,,0,,0,,,0,1101,,,1,,0000,0,,,11,000,,,010,010,101]

    6225: !(FREE)
    SETUCONS06A:
        P0,     BUSDIN_FLAG&FPS-[I],                !TAKE EMIT OFF BUSDIN, FLAG:FPS=(000005)
        P3,     BUTA(CLR-FLAG-RES-UCON),            !THIS SHOULD NOW CLEAR THE UCON REGISTER,
                                                    !  SETTING UCON-SELECT(1)H=L; FORCING BUSDIN_EMIT
        NEXT,   J/PSHIS06A                          !
 (6225) DCS[0,00,0,0,0,0]  BM[0000,,00,00,,11,01,,000,,000,,,0,0,0,,0,,0,,,1,1001,,,0,,0000,0,,,11,010,,,010,010,110]

    6226: !(FREE)
    PSHIS06A:
        P0,     BUMP-VERIFY,                        !COUNT
        P3,     CSPD[05]_EMIT,                      !VALUE IN D WHEN LOAD PS<15,13>:
                EMIT/063125,                        !"0110 0110 0101 0101"
        NEXT,   J/PSMIDS06A                         !
 (6226) DCS[0,00,0,0,0,1]  BM[0110,,10,01,,10,01,,010,,101,,,0,0,0,,0,,0,,,0,1010,,,1,,0000,0,,,11,000,,,010,010,111]

    6227: !(FREE)
    PSMIDS06A:
        P3,     CSPD[06]_EMIT,                      !VALUE IN D WHEN LOAD PS<7:4>:
                EMIT/143245,                        !"1100 0110 1010 0101"
        NEXT,   J/PSLOS06A                          !
 (6227) DCS[0,00,0,0,0,0]  BM[1100,,10,01,,10,10,,100,,101,,,0,0,0,,0,,0,,,0,1001,,,1,,0000,0,,,11,000,,,010,011,000]


    6230: !(FREE)
    PSLOS06A:
        P0,     BUMP-VERIFY,                        !COUNT
        P3,     CSPD[07]_EMIT,                      !VALUE IN D WHEN LOAD PS<3:0>:
                EMIT/143132,                        !"1100 0110 0101 1010"
        NEXT,   J/FUDGEPSS06A                       !
 (6230) DCS[0,00,0,0,0,1]  BM[1100,,10,01,,10,01,,011,,010,,,0,0,0,,0,,0,,,0,1000,,,1,,0000,0,,,11,000,,,010,011,001]
```

```
    6231:  ;(FREE)
    FUDGEPS506A:
            SETUP,  RETURN/LOADUCON506A,                        ;GO TO SUBR WHICH:
                                                                ;1) CSP(05) => PS<15,13>
                                                                ;2) CSP(06) => PS<7:4>
            NEXT,   CALL(PSSEQLOD]                              ;3) CSP(07) => PS<3:0>
  (6231)  DCS[0,00,0,0,0,0]  BM[0111,,00,00,,01,10,,111,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,001,000,000]

    7067:  ;(FREE)
    LOADUCON506A:
            SETUP,  UCON-PROC,                                  ;SETUP FOR UCON-PROC:
            ENABLE, BUSDIN_FLAGS&FPS-(I),                       ;FLAGS&FPS=(000005)
                    EN-CLK-PS[15-12], EN-CLK-PS[7-4],           ;SETUP FOR CLOCKING PS BITS
                    EN-CLK-PS[3-0],                             ;
            P0,     SET-UCON-CONTROL,                           ;WRITE THE UCON REGISTER AT P0,
            P3,     BUTA(CLR-FLAG-RES-UCON),                    ;BUT AT P3, NOW CLEAR IT OUT
            NEXT,   J/DOIT506A                                  ;
  (7067)  DCS[0,00,0,0,0,0]  BM[1000,,00,00,,11,01,,010,,010,,,0,0,0,,0,,0,,,1,1001,,,0,,0000,0,,,11,010,,,000,111,010]

    7072:  ;(FREE)
    DOIT506A:
            P2-T,   D_ZERO,                                     ;SETUP D WITH A (000000)
            NEXT,   GOTO-PAGE(6),                               ;XFER
                    J/LOADDCS506A                               ;
  (7072)  DCS[0,00,0,0,0,0]  BM[0011,,00,00,,00,00,,000,,110,,,0,1,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,001,000,100]

    6104:  ;(FREE)
    LOADDCS506A:
            P0,     DCS-CTR(C9,),                               ;COMPARE AT TARGET
            P2,     UCON-OPERATION,                             ;IF THE UCON REGISTER WASN'T CLEARED, ABOVE,
            NEXT,   GOTO-PAGE(7),                               ;  THE PS SHOULD BE OVER-WRITTEN WITH (000000),
                    J/GOBUT506A                                 ;    IN SOME COMBINATION OF PS<15:12,7:4,3:0>
  (6104)  DCS[0,00,1,0,0,0]  BM[0110,,00,00,,00,00,,000,,111,,,0,0,0,,0,,0,,,1,1010,,,0,,0000,0,,,11,100,,,000,111,001]

    7071:  ;(FREE)
    GOBUT506A:
            SETUP,  RETURN/TEST506B,                            ;RETURN TO START OF NEXT SUBTEST
            NEXT,   CALL[PSTOD]                                 ;  PS-XOR-CSP(02) => D, BUT(D=ZERO)
  (7071)  DCS[0,00,0,0,0,0]  BM[0110,,00,11,,11,01,,100,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,101,011]




    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

    ;* TEST 506B *
    ;CHECK THAT BUTP(PS15) SHOWS PS<15>H CLEAR
    6754:
    TEST506B:
            P0,     LOAD-ENUA(%TARGET402),                      ;BIT<00> CLEAR
```



```
                    LOAD-ERROR(TEST506B),                       ;ERROR DIRECTORY KEY
                    DCS-CTR(C3,),                               ;COMPARE AT TARGET
            NEXT,   J/GOBUT506B                                 ;
  (6754)  DCS[1,00,1,0,0,0]  BM[1100,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,010,011,011]

    6233:  ;(FREE)
    GOBUT506B:
            SETUP,  RETURN/TEST506C,                            ;RETURN TO START OF NEXT SUBTEST
            NEXT,   GOTO-PAGE(7),                               ;BUT TABLE HERE
                    J/BUTPS15                                   ;PS<15>H IN BIT<00>
  (6233)  DCS[0,00,0,0,0,0]  BM[0110,,00,11,,11,00,,100,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,100,011]




    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

    ;* TEST 506C *
    ;CHECK THAT BUTN(PS03) SHOWS PS<03>H SET
    6744:
    TEST506C:
            P0,     LOAD-ENUA(%TARGET407),                      ;BIT<00> SET
                    LOAD-ERROR(TEST506C),                       ;ERROR DIRECTORY KEY
                    DCS-CTR(C4,),                               ;COMPARE AT TARGET
                    BUMP-VERIFY,                                ;COUNT
            NEXT,   J/GOBUT506C                                 ;
  (6744)  DCS[1,00,1,0,0,1]  BM[1011,,00,11,,11,00,,000,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,010,011,100]

    6234:  ;(FREE)
    GOBUT506C:
            SETUP,  RETURN/TEST506D,                            ;RETURN TO START OF NEXT SUBTEST
            NEXT,   GOTO-PAGE(7),                               ;BUT TABLE HERE
                    J/BUTNPS[N]                                 ;PS[N]=PS<03>H IN BIT<00>
  (6234)  DCS[0,00,0,0,0,0]  BM[0110,,00,11,,10,11,,100,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,010,011]




    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

    ;* TEST 506D *
    ;CHECK THAT WHEN FLAG<0>H=L AND PS<04>H=L,
    ;  MASKED-PS[T]=H*FLAG<0>L+PS<04>H, IS LOW
    6734:
    TEST506D:
            P0,     LOAD-ENUA(%TARGET406),                      ;BIT<00> CLEAR

                    LOAD-ERROR(TEST506D),                       ;ERROR DIRECTORY KEY
                    DCS-CTR(C4,),                               ;COMPARE AT TARGET
            NEXT,   J/GOBUT506D                                 ;
  (6734)  DCS[1,00,1,0,0,0]  BM[1011,,00,11,,11,00,,000,,110,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,010,011,101]

    6235:  ;(FREE)
```

```
	GOBUT506D:
		P0,	BUMP-VERIFY,			;COUNT
		SETUP,	RETURN/TEST506E,		;RETURN TO START OF NEXT SUBTEST
		NEXT,	GOTO-PAGE(7),			;BUT TABLE HERE
			J/BUTNMASKPS[T]			;FLAG..........N IN BIT<00>
 (6235)	DCS[0,00,0,0,0,1]  BN[0110,,00,11,,00,00,,101,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,001,111]




	; - - - - - - - - - - - - - - - - - - - - - - - - - - - -

	;* TEST 506E *
	;CHECK THAT INTR-HIGH-H=SERVICE(0)H*NOT(MASKED-PS(T)-H) IS HIGH,
	;  WHEN SERVICE(0)H=HIGH, MASKED-PS[T]=H=LOW
	;NOTE:  THE SUBR CALLED SHOULD SET SERVICE(0)H=HIGH (NEGATED) BY CLEARING OUT ALL ITS INPUT CONDITIONS
	6605:
	TEST506E:
		P0,	LOAD-ENUA(ZTARGET403),		;BIT<03> SET
			LOAD-ERROR(TEST506E),		;ERROR DIRECTORY KEY
			DCS-CTR(C9,),			;COMPARE AT TARGET
		NEXT,	J/CLEAR506E			;
 (6605)	DCS[1,00,1,0,0,0]  BN[0110,,00,11,,11,00,,000,,011,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,010,011,110]

	6236:	;(FREE)
	CLEAR506E:
		P0,	BUMP-VERIFY,			;COUNT
		SETUP,	RETURN/GOBUT506E,		;GO TO SUBR WHICH CLEARS OUT
		NEXT,	CALL[CLEAR-I-O-B]		;  ALL THE UNIBUS/I-O/SERVICE LATCHES  (NO UNIBUS INIT)
 (6236)	DCS[0,00,0,0,0,1]  BN[0110,,00,01,,00,11,,111,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,011,000]

	6237:	;(FREE)
	GOBUT506E:
		SETUP,	RETURN/SCOPE506,		;RETURN TO SCOPE LOOP TEST WORD
		NEXT,	GOTO-PAGE(7),			;BUT TABLE HERE
			J/BUTINTRHIGH			;INTR HIGH H IN BIT<01>
 (6237)	DCS[0,00,0,0,0,0]  BN[0110,,00,01,,01,00,,000,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,101,101]




	6240:	;(FREE)
	SCOPE506:
		P0,	BUSDIN_EMIT-[I],		;KEEP EMIT FOR CONSTANTS
		P2,	PS_D-[I],			;ZERO PS; D LEFT ZERO FROM
							;PREVIOUS TESTS IF ALL OK
		NEXT,	BUTD[SCOPE],			;NO ERROR: "TEST507A" (+1, WORDS)
			J/TEST507A			;  ERROR: "MASK506A" (-16, WORDS)
 (6240)	DCS[0,00,0,1,0,0]  BN[1000,,00,00,,00,01,,010,,010,,,0,0,0,,0,,0,,,1,1011,,,0,,0000,0,,,11,000,,,110,001,011]
```



```




	; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

	;*** TEST 507 ***

	;TESTS 507 A-F USE A "1-0-0101-0101" PATTERN IN PS<15,13,7:4,3:0>

	; - - - - - - - - - - - - - - - - - - - - - - - - - - - -

	;* TEST 507A *
	;LOAD UP PS<HI,MID,LO> IN ORDER, READ BACK THRU PS PORT OF PROC MUX
	6613:
	TEST507A:
		P0,	LOAD-ENUA(ZTARGET402),		;SETUP FOR D=ZERO TEST
			LOAD-ERROR(TEST507A),		;ERROR DIRECTORY KEY
			DCS-CTR(C15,),			;STALL FOR NOW
		NEXT,	J/EXPEC507A			;
 (6613)	DCS[1,00,1,0,0,0]  BN[0000,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,110,000,010]

	6602:
	EXPEC507A:
		P3,	CSPD[02]_EMIT,			;EXPECTED VALUE TO BE READ OUT OF
			EMIT/140125,			;PS AFTER LOADING:
		NEXT,	J/PSHI507A			;"1100 0000 0101 0101"
 (6602)	DCS[0,00,0,0,0,0]  BN[1100,,10,00,,00,01,,010,,101,,,0,0,0,,0,,0,,,0,1101,,,1,,0000,0,,,11,000,,,010,100,001]

	6241:	;(FREE)
	PSHI507A:
		P0,	BUMP-VERIFY,			;COUNT
		P3,	CSPD[05]_EMIT,			;VALUE IN D WHEN LOAD PS<15,13>:
			EMIT/113252,			;"1001 0110 1010 1010"
		NEXT,	J/PSMID507A			;
 (6241)	DCS[0,00,0,0,0,1]  BN[1001,,10,01,,10,10,,101,,010,,,0,0,0,,0,,0,,,0,1010,,,1,,0000,0,,,11,000,,,010,100,010]

	6242:	;(FREE)
	PSMID507A:
		P3,	CSPD[06]_EMIT,			;VALUE IN D WHEN LOAD PS<7:4>:
			EMIT/033132,			;"0011 0110 0101 1010"
		NEXT,	J/PSLO507A			;
 (6242)	DCS[0,00,0,0,0,0]  BN[0011,,10,01,,10,01,,010,,010,,,0,0,0,,0,,0,,,0,1001,,,1,,0000,0,,,11,000,,,010,100,011]

	6243:	;(FREE)
	PSLO507A:

		P0,	BUMP-VERIFY,			;COUNT
		P3,	CSPD[07]_EMIT,			;VALUE IN D WHEN LOAD PS<3:0>:
			EMIT/033245,			;"0011 0110 1010 0101"
		NEXT,	J/FUDGEPS507A			;
 (6243)	DCS[0,00,0,0,0,1]  BN[0011,,10,01,,10,10,,100,,101,,,0,0,0,,0,,0,,,0,1000,,,1,,0000,0,,,11,000,,,010,100,100]

	6244:	;(FREE)
	FUDGEPS507A:
```

```
12178           SETUP,  RETURN/LOADDC5507A,                     ;GO TO SUBR WHICH:
12179           NEXT,   CALL[PSSEQLOD]                          ;1) CSP(05) -> PS<15,13>
     (6244)  DCS[0,00,0,0,0,0]  BM[0110,,00,01,,01,00,,101,,111,,,0,0,0,,0,,0,,0,0000,,,0,,0000,0,,,11,100,,,001,000,000]
12180                                                           ;2) CSP(06) -> PS<7:4>
12181                                                           ;3) CSP(07) -> PS<3:0>
12182
12183   6245:  ;(FREE)
12184   LOADDC5507A:
12185           P0,     DCS=CTR(C9,),                           ;COMPARE AT TARGET
12186           NEXT,   J/GOBUT507A                             ;
     (6245)  DCS[0,00,1,0,0,0]  BM[0110,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,0,0000,,,0,,0000,0,,,11,000,,,010,100,110]
12187
12188   6246:  ;(FREE)
12189   GOBUT507A:
12190           SETUP,  RETURN/TEST507B,                        ;RETURN TO START OF NEXT SUBTEST
12191           NEXT,   CALL[PSTOD]                             ;  CSP(03)=XOR-PS -> D, BUT(D=ZERO)
     (6246)  DCS[0,00,0,0,0,0]  BM[0110,,00,11,,10,10,,100,,111,,,0,0,0,,0,,0,,0,0000,,,0,,0000,0,,,11,100,,,010,101,011]
12192
12193
12194
12195
12196   ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
12197
12198   ;* TEST 5070 *
12199   ;CHECK THAT BUTN(PS15) SHOWS PS<15>H SET
12200   6724:
12201   TEST507B:
12202           P0,     LOAD=ENUA(ETARGET403),                  ;BIT<00> SET
12203                   LOAD=ERROR(TEST507B),                   ;ERROR DIRECTORY KEY
12204                   DCS=CTR(C3,),                           ;COMPARE AT TARGET
12205           NEXT,   J/GOBUT507B                             ;
     (6724)  DCS[1,00,1,0,0,0]  BM[1100,,00,11,,11,00,,000,,011,,,0,0,0,,0,,0,,0,0000,,,0,,0000,0,,,11,000,,,010,100,111]
12206
12207   6247:  ;(FREE)
12208   GOBUT507B:
12209           SETUP,  RETURN/TEST507C,                        ;RETURN TO START OF NEXT SUBTEST
12210           NEXT,   GOTO=PAGE(7),                           ;BUT TABLE HERE
12211                   J/BUTPS15                               ;PS<15>H IN BIT<00>
     (6247)  DCS[0,00,0,0,0,0]  BM[0110,,00,11,,00,01,,001,,111,,,0,0,0,,0,,0,,0,0000,,,0,,0000,0,,,11,100,,,011,100,011]
12212
12213
12214
12215
12216   ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
12217
12218   ;* TEST 507C *
12219   ;CHECK THAT BUTN(PS03) SHOWS PS<03>H CLEAR
12220   6611:
12221   TEST507C:
12222           P0,     LOAD=ENUA(ETARGET406),                  ;BIT<00> CLEAR
12223                   LOAD=ERROR(TEST507C),                   ;ERROR DIRECTORY KEY
```



```
12224                   DCS=CTR(C4,),                           ;COMPARE AT TARGET
12225                   BUMP=VERIFY,                            ;COUNT
12226           NEXT,   J/GOBUT507C                             ;
     (6611)  DCS[1,00,1,0,0,1]  BM[1011,,00,11,,11,00,,000,,110,,,0,0,0,,0,,0,,0,0000,,,0,,0000,0,,,11,000,,,010,101,000]
12227
12228   6250:  ;(FREE)
12229   GOBUT507C:
12230           SETUP,  RETURN/TEST507D,                        ;RETURN TO START OF NEXT SUBTEST
12231           NEXT,   GOTO=PAGE(7),                           ;BUT TABLE HERE
12232                   J/BUTNPS[N]                             ;PS[N]=PS<03>N IN BIT<00>
     (6250)  DCS[0,00,0,0,0,0]  BM[0110,,00,10,,11,01,,111,,111,,,0,0,0,,0,,0,,0,0000,,,0,,0000,0,,,11,100,,,011,010,011]
12233
12234
12235
12236
12237   ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
12238
12239   ;* TEST 507D *
12240   ;CHECK THAT WHEN FLAG<0>H=L AND PS<04>H=N,
12241   ;  MASKED-PS(T)-H=FLAG<0>L=PS<04>H, IS HIGH
12242   6557:
12243   TEST507D:
12244           P0,     LOAD=ENUA(ETARGET407),                  ;BIT<00> SET
12245                   LOAD=ERROR(TEST507D),                   ;ERROR DIRECTORY KEY
12246                   DCS=CTR(C4,),                           ;COMPARE AT TARGET
12247           NEXT,   J/GOBUT507D                             ;
     (6557)  DCS[1,00,1,0,0,0]  BM[1011,,00,11,,11,00,,000,,111,,,0,0,0,,0,,0,,0,0000,,,0,,0000,0,,,11,000,,,010,101,001]
12248
12249   6251:  ;(FREE)
12250   GOBUT507D:
12251           P0,     BUMP=VERIFY,                            ;COUNT
12252           SETUP,  RETURN/TEST507E,                        ;RETURN TO START OF NEXT SUBTEST
12253           NEXT,   GOTO=PAGE(7),                           ;BUT TABLE HERE
12254                   J/BUTNMASKPS(T                          ;FLAG<0>L=PS<04>H IN BIT<00>
     (6251)  DCS[0,00,0,0,0,1]  BM[0110,,00,11,,01,11,,100,,111,,,0,0,0,,0,,0,,0,0000,,,0,,0000,0,,,11,100,,,011,001,111]
12255
12256
12257
12258
12259   ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
12260
12261   ;* TEST 507E *
12262   ;CHECK THAT INTR=HIGH=H=SERVICE(0)H=NOT(MASKED-PS(T)-H) IS LOW,
12263   ;  WHEN SERVICE(0)H=HIGH, MASKED-PS(T)-H=HIGH
12264   4674:
12265   TEST507F:

12266           P0,     LOAD=ENUA(ETARGET401),                  ;BIT<01> CLEAR
12267                   LOAD=ERROR(TEST507E),                   ;ERROR DIRECTORY KEY
12268                   DCS=CTR(C3,),                           ;COMPARE AT TARGET

12269                   BUMP=VERIFY,                            ;COUNT
12270           NEXT,   J/GOBUT507E                             ;
     (6674)  DCS[1,00,1,0,0,1]  BM[1100,,00,11,,11,00,,000,,001,,,0,0,0,,0,,0,,0,0000,,,0,,0000,0,,,11,000,,,010,101,011]
```

```
12271
12272    6253:  ;(FREE)
12273    GOBUT507E:
12274            SETUP,  RETURN/TEST507F;                        ;RETURN TO START OF NEXT SUBTEST
12275            NEXT,   GOTO=PAGE(7);                           ;BUT TABLE HERE
12276                    J/BUTINTRHIGH                           ;INTR HIGH H IN BIT<01>
      (6253)  DCS[0,00,0,0,0,0]  BM[0110,,00,11,,01,00,,11],,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,101,101]
12277
12278
12279
12280    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
12281
12282    ;* TEST 507F *
12283    ;CHECK THAT PS(C)=H CAN BE READ AS A "1" THRU D(C)_CINMUX=PS(C), AND D(C)_PS(C)
12284    ;  IT WAS PREVIOUSLY CHECKED, IN ALU LOGIC TESTS, THAT THESE WERE OR WITH PS(C)="0"
12285    6647:
12286    TEST507F:
12287            P0,     LOAD=ENUA(ETARGET407),                  ;SETUP FOR SR<3:0> = "0111"
12288                    LOAD=ERROR(TEST507F),                   ;ERROR DIRECTORY KEY
12289                    DCS=CTR(C7,);                           ;COMPARE AT TARGET
12290            NEXT,   J/CIN507F                               ;
      (6647)  DCS[1,00,1,0,0,0]  BM[1000,,00,11,,11,00,,000,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,010,101,100]
12291
12292    6254:  ;(FREE)
12293    CIN507F:
12294            P0,     BUMP=VERIFY,                            ;COUNT
12295            P2=T,   D_ZERO, D(C)_CINMUX,                    ;SET D(C)=CINMUX=PS(C)=0(1)
12296                    SR_ZERO;                                ;SET SR<3:0>="0000"
12297            P3,     BSPLC[17]_D,                            ;B-SIDE COPY OF SR
12298            NEXT,   J/SETPS507F                             ;
      (6254)  DCS[0,00,0,0,0,1]  BM[0011,,00,11,,00,00,,011,,000,,,0,1,1,,0,,0,,,0,0000,,,0,,0110,0,,,11,000,,,010,101,110]
12299
12300    6255:  ;(FREE)
12301    SETPS507F:
12302            P2=T,   D_A=PLUS=B=PLUS=D(C), D(C)_PS(C),        ;SET D(C)=PS(C)=0(1)
12303                    SR_A=PLUS=B=PLUS=D(C),                  ;D, SR, BSP <= SR-LEFT-1, SR<02> <= PREV D(C) = CINMUX/PS(C)
12304                            BUS=A_SR,                       ;
12305                            BUS=B_BSPLC(R17),               ;
12306            P3,     BSPLC[17]_D,                            ;SAVE ON B-SIDE
12307            NEXT,   J/SETSR507F
      (6255)  DCS[0,00,0,0,0,0]  BM[0100,,00,11,,00,00,,011,,001,,,0,1,1,,0,,0,,,0,0000,,,0,,0110,0,,,11,000,,,010,101,111]
12308
12309    6256:  ;(FREE)
12310    SETSR507F:
12311            P0,     BUMP=VERIFY,                            ;COUNT
12312            P2=T,   D_A=PLUS=B=PLUS=D(C), D(C)_ALU15,       ;SET D(C)=0
12313                    SR_A=PLUS=B=PLUS=D(C),                  ;D, SR, BSP <= SR-LEFT-1, SR<01> <= PREV D(C) = PS(C)
12314                            BUS=A_SR,                       ;
12315                            BUS=B_BSPLC(R17),               ;
12316            P3,     BSPLC[17]_D,                            ;SAVE ON B-SIDE
12317            NEXT,   J/SETCIN507F                            ;
      (6256)  DCS[0,00,0,0,0,1]  BM[0100,,00,11,,00,00,,011,,100,,,0,1,1,,0,,0,,,0,0000,,,0,,0110,0,,,11,000,,,010,110,000]
12318
```



```
12319    6257:  ;(FREE)
12320    SETCIN507F:
12321            P2=T,   D_A=PLUS=B=PLUS=PS(C), D(C)_ALU15,      ;SET D(C)=0
12322                    SR_A=PLUS=B=PLUS=PS(C),                 ;D, SR, BSP <= SR-LEFT-1, SR<00> <= CIN/PS(C)
12323                            BUS=A_SR,                       ;
12324                            BUS=B_BSPLC(R17),               ;
12325            NEXT,   J/GOBUT507F                             ;
      (6257)  DCS[0,00,0,0,0,0]  BM[0001,,00,11,,00,00,,011,,100,,,0,1,1,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,010,110,001]
12326
12327    6261:  ;(FREE)
12328    GOBUT507F:
12329            SETUP,  RETURN/SCOPE507;                        ;RETURN TO SCOPE LOOP TEST WORD
12330            NEXT,   GOTO=PAGE(7);                           ;BUT TABLE
12331                    J/BUTSR3=0                              ;CHECK THAT WE GOT "0111" = (07)
      (6261)  DCS[0,00,0,0,0,0]  BM[0110,,00,01,,01,10,,010,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,111,110]
12332
12333
12334
12335
12336
12337    6262:  ;(FREE)
12338    SCOPE507:
12339            P0,     BUSOIN_INIT=(1),                        ;KEEP INIT FOR CONSTANTS
12340            P1,     PS_D=(1),                               ;ZERO PS; D LEFT ZERO FROM
12341                                                            ;PREVIOUS TESTS IF ALL OK
12342            NEXT,   BUTO(SCOPE),                            ;NO ERROR: "TEST510A" (+1, WORD8)
12343                    J/TEST510A                              ;   ERROR: "EXPEC507A" (-21, WORD8)
      (6262)  DCS[0,00,0,1,0,0]  BM[1000,,00,00,,00,01,,010,,010,,,0,0,0,,0,,0,,,1,1011,,,0,,0000,0,,,11,000,,,110,000,011]
12344
12345
12346
12347
12348
12349
12350    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
12351
12352    ;*** TEST 510 ***
12353
12354    ;TESTS 510 A-F USE PS<7:5>H="111", PS<4>H="1", VARIOUS FLAG<7,0>H COMBINATIONS,
12355    ;  TO TEST THE INTR-HIGH-H, SERVICE-H, AND MASKED=PS(T)-H LOGIC
12356
12357    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
12358
12359    ;* TEST 510A *
12360    ;CHECK THAT SG-SERVICE(0)H=HIGH WHEN PS<7:5>H="111" (PSW PRIORITY 7); SO, BR>PS=H=LOW, SINCE NO
12361    ;  EXTERNAL UNIBUS DEVICE CAN THEN REQUEST AN INTERRUPT (IE, IT IS MASKED OUT)

12362    6603:
12363    TEST510A:
12364            P0,     LOAD=ENUA(ETARGET407),                  ;BIT<02> SET
12365                    LOAD=ERROR(TEST510A),                   ;ERROR DIRECTORY KEY
12366                    DCS=CTR(C5,);                           ;COMPARE AT TARGET
12367            NEXT,   J/SETONE510A                            ;
      (6603)  DCS[1,00,1,0,0,0]  BM[1010,,00,11,,11,00,,000,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,110,000,110]
```

```
12368
12369    6606:
12370    SETONES$10A:
12371            P2=T,   D_ALL-ONES; SAVE=D[C],                  ;ALL ONES FOR LOAD PRIORITY 7
12372            NEXT,   J/LOADPRIO$10A                          ;
     (6606)  DCS[0,00,0,0,0,0]  BM[1111,,00,00,,11,01,,101,,111,,,0,1,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,010,110,011]
12373
12374    6263:  ;(FREE)
12375    LOADPRIO$10A:
12376            P0,     BUSDIN_EMIT-[I],                        ;KEEP IT ON
12377            P3=T,   P$[7-4]_D[7-4]-[I],                     ;PRIO 7, T-BIT (P$04) SET
12378            NEXT,   J/GOBUT$10A                             ;
     (6263)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,01,,010,,000,,,1,0,0,,0,,0,,,1,1011,,,0,,0000,0,,,11,000,,,010,110,100]
12379
12380    6264:  ;(FREE)
12381    GOBUT$10A:
12382            SETUP,  RETURN/TEST$10B,                        ;RETURN TO START OF NEXT SUBTEST
12383            NEXT,   GOTO=PAGE(7),                           ;BUT TABLE
12384                    J/BUTSGSERVL                            ;BG-SERVICE(0)H IN BIT<02>
     (6264)  DCS[0,00,0,0,0,0]  BM[0110,,00,11,,01,11,,101,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,001,110]
12385
12386
12387
12388
12389
12390    ;  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -
12391
12392    ;* TEST 510B *
12393    ;CHECK THAT BUT[ NOT(BG-SERVICE(0)H) + NOT(FLTPT-SERVICE-L) ] IN BIT<03>
12394    ;  SETS BG-SERVICE(0)H=HIGH, FLTPT-SERVICE-L=HIGH, AND THUS BG NOT ASSERTED (=LOW)
12395    6675:
12396    TEST$10B:
12397            P0,     LOAD=ENUA(ETARGET407),                  ;BIT<03> CLEAR
12398                    LOAD=ERROR(TEST$10B),                   ;ERROR DIRECTORY KEY
12399                    DCS=CTR(C),,                            ;COMPARE AT TARGET
12400            NEXT,   J/GOBUT$10B                             ;
     (6675)  DCS[1,00,1,0,0,0]  BM[1100,,00,11,,11,00,,000,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,010,110,101]
12401
12402    6265:  ;(FREE)
12403    GOBUT$10F:
12404            SETUP,  RETURN/TEST$10C,                        ;RETURN TO START OF NEXT SUBTEST
12405            NEXT,   GOTO=PAGE(7),                           ;BUT TABLE
12406                    J/BUTBGFPSERV                           ;REBUT IN BIT<03>
     (6265)  DCS[0,00,0,0,0,0]  BM[0110,,00,11,,00,01,,110,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,000,111]
12407
12408
12409
12410
12411
12412    ;  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -
12413
12414    ;* TEST 510C *
```



```
12415    ;CHECK THAT SERVICE-H=INTR-HIGH-H+BG-SERVICE(0)H+NOT(FLAG<7>H) IS HIGH,
12416    ;  WHEN INTR-HIGH-H=LOW, BG-SERVICE(0)H=HIGH, FLAG<7>H=LOW
12417    6616:
12418    TEST$10C:
12419            P0,     LOAD=ENUA(ETARGET403),                  ;BIT<00> SET
12420                    LOAD=ERROR(TEST$10C),                   ;ERROR DIRECTORY KEY
12421                    DCS=CTR(C),,                            ;COMPARE AT TARGET
12422            NEXT,   J/GOBUT$10C                             ;
     (6616)  DCS[1,00,1,0,0,0]  BM[1100,,00,11,,11,00,,000,,011,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,010,110,110]
12423
12424    6266:  ;(FREE)
12425    GOBUT$10C:
12426            SETUP,  RETURN/TEST$10D,                        ;RETURN TO START OF NEXT SUBTEST
12427            NEXT,   GOTO=PAGE(7),                           ;BUT TABLE
12428                    J/BUTSERVICE                            ;SERVICE=H IN BIT<00>
     (6266)  DCS[0,00,0,0,0,0]  BM[0110,,00,11,,00,00,,000,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,100,101]
12429
12430
12431
12432    ;  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -
12433
12434    ;* TEST 510D *
12435    ;SET ONLY FLAG<0>H, FLAG<7>H, LEAVE P$<04>H SET,
12436    ;CHECK MASKED-P$(T)-H=P$<04>H+FLAG<0>L IS LOW
12437    6600:
12438    TEST$10D:
12439            P0,     LOAD=ENUA(ETARGET406),                  ;BIT<00> CLEAR
12440                    LOAD=ERROR(TEST$10D),                   ;ERROR DIRECTORY KEY
12441                    DCS=CTR(C7,),                           ;COMPARE AT TARGET
12442                    BUMP=VERIFY,                            ;COUNT
12443            NEXT,   J/FLAG$10D                              ;
     (6600)  DCS[1,00,1,0,0,1]  BM[1000,,00,11,,11,00,,000,,110,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,010,110,111]
12444
12445    6267:  ;(FREE)
12446    FLAG$10D:
12447            P3,     CSPD(01)_EMIT,                          ;
12448                    EMIT/040401,                            ;BITS FOR FLAG<7,0> ONLY, BIT<00> OF D
12449            NEXT,   J/ENFLAG$10D                            ;
     (6267)  DCS[0,00,0,0,0,0]  BM[0100,,10,00,,01,00,,000,,001,,,0,0,0,,0,,0,,,0,1110,,,1,,0000,0,,,11,000,,,010,111,000]
12450
12451    6270:  ;(FREE)
12452    ENFLAG$10D:
12453            P3=T,   D_CSPD(D01), D[C]_0,                    ;INTO D
12454            NEXT,   J/SETFLAG$10D                           ;
     (6270)  DCS[0,00,0,0,0,0]  BM[1010,,10,00,,00,00,,000,,000,,,1,1,0,,0,,0,,,0,1110,,,0,,0000,0,,,11,000,,,010,111,001]
12455
12456    6271:  ;(FREE)
12457    SETFLAG$10D:
12458            P3=T,   FLAG[8-0]_D[15-8]-[I],                   ;SET FLAGS<7,0>H ONLY
12459            NEXT,   J/GOBUT$10D                             ;
     (6271)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,01,,000,,001,,,1,0,0,,0,,0,,,1,1011,,,0,,0000,0,,,11,000,,,010,111,010]
12460
12461    6272:  ;(FREE)
```

```
12462    GOBUT510D:
12463            SETUP,  RETURN/TEST510DA,                       ;RETURN TO START OF NEXT SUBTEST
12464            NEXT,   GOTO-PAGE(7),                           ;BUT TABLE HERE
12465                    J/BUTMMASKPS[7]                         ;FLAG<03>&PS<04>=M IN BIT<00>
     (6272)  DCS[0,00,0,0,0,0]  BM[0110,,00,11,,00,01,,111,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,001,111]
12466
12467
12468
12469
12470    ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
12471
12472    !*** TEST 510DA ***
12473    !DO THE "MULTIPLE BUT" ON "D<00>=M" TO CHECK IT'S SET  (DURING TEST510D, ABOVE)
12474    66171
12475    TEST510DA:
12476            PO,     LOAD=ENUA(ETARGET407),                  ;BIT<00> SET
12477                    LOAD=ERROR(TEST510DA),                  ;ERROR DIRECTORY KEY
12478                    DCS=CTR(C4,),                           ;COMPARE AT TARGET
12479                    BUMP=VERIFY,                            ;COUNT
12480            NEXT,   J/GOBUT510DA                            ;
     (6617)  DCS[1,00,1,0,0,1]  BM[1011,,00,11,,11,00,,000,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,010,111,011]
12481
12482    6273:  ;(FREE)
12483    GOBUT510DA:
12484            SETUP,  RETURN/TEST510E,                        ;RETURN TO START OF NEXT SUBTEST
12485            NEXT,   GOTO-PAGE(7),                           ;BUT TABLE
12486                    J/BUTMD00                               ;D<00>=M IN BIT<00>
     (6273)  DCS[0,00,0,0,0,0]  BM[0110,,00,11,,00,00,,100,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,010,001]
12487
12488
12489
12490
12491    ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
12492
12493    !* TEST 510E *
12494    !CHECK THAT SERVICE-H=INTR-HIGH-H=BG-SERVICE(0)H=NOT(FLAG<7>H) IS HIGH,
12495    !  WHEN INTR-HIGH-H=HIGH, BG-SERVICE(0)H=HIGH, FLAG<7>H=HIGH
12496    66041
12497    TEST510E:
12498            PO,     LOAD=ENUA(ETARGET403),                  ;BIT<00> SET
12499                    LOAD=ERROR(TEST510E),                   ;ERROR DIRECTORY KEY
12500                    DCS=CTR(C4,),                           ;COMPARE AT TARGET
12501            NEXT,   J/ZERODS10E                             ;
     (6604)  DCS[1,00,1,0,0,0]  BM[1011,,00,11,,11,00,,000,,011,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,010,111,100]
12502
12503    6274:  ;(FREE)
12504    ZERODS10E:
12505            PZ=T,   D_ZERO, DIC1_ALU15,                     ;ZEROES FOR BELOW
12506            NEXT,   J/GOBUT510E                             ;
     (6274)  DCS[0,00,0,0,0,0]  BM[0011,,00,00,,00,00,,000,,100,,,0,1,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,010,111,101]
12507
12508    6275:  ;(FREE)
```



```
12509    GOBUT510E:
12510            PO,     BUMP=VERIFY,                            ;COUNT
12511            SETUP,  RETURN/TEST510F,                        ;RETURN TO START OF NEXT SUBTEST
12512            NEXT,   GOTO-PAGE(7),                           ;BUT TABLE HERE
12513                    J/BUTSERVICE                            ;SERVICE=M IN BIT<00>
     (6275)  DCS[0,00,0,0,0,1]  BM[0110,,00,11,,00,01,,000,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,100,101]
12514
12515
12516
12517
12518    ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
12519
12520    !* TEST 510F *
12521    !CHECK THAT SERVICE-H=INTR-HIGH-H=BG-SERVICE(0)H=NOT(FLAG<7>H) IS LOW,
12522    !  WHEN INTR-HIGH-H=HIGH, BG-SERVICE(0)H=HIGH, FLAG<7>H=LOW
12523    66101
12524    TEST510F:
12525            PO,     LOAD=ENUA(ETARGET402),                  ;BIT<00> CLEAR
12526                    LOAD=ERROR(TEST510F),                   ;ERROR DIRECTORY KEY
12527                    DCS=CTR(C4,),                           ;COMPARE AT TARGET
12528            NEXT,   J/ZEROFLAGS10E                          ;
     (6610)  DCS[1,00,1,0,0,0]  BM[1011,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,010,111,110]
12529
12530    62761  ;(FREE)
12531    ZEROFLAGS10E:
12532            P2,     PS_D=(I),                               ;ZERO PS=T-BIT, AND ALL THE OTHERS,
12533                    FPS(7-4)_D(7-4)=(I),                    ;ZERO THE FPS
12534            P3,     FLAG(8-0)_D(15-8)=(I),                  ;ZERO FLAGS (D ZEROED ABOVE)
12535            NEXT,   J/GOBUT510F                             ;
     (6276)  DCS[0,00,0,0,0,0]  BM[1000,,00,00,,00,01,,110,,011,,,0,0,0,,0,,0,,,1,1011,,,0,,0000,0,,,11,000,,,010,111,111]
12536
12537    6277:  ;(FREE)
12538    GOBUT510F:
12539            SETUP,  RETURN/SCOPE510,                        ;RETURN TO SCOPE LOOP TEST WORD
12540            NEXT,   GOTO-PAGE(7),                           ;BUT TABLE HERE
12541                    J/BUTSERVICE                            ;SERVICE=M IN BIT<00>
     (6277)  DCS[0,00,0,0,0,0]  BM[0110,,00,01,,10,00,,000,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,100,101]
12542
12543
12544
12545
12546
12547    63001  ;(FREE)
12548    SCOPE510:
12549            PO,     BUSDIN_EMIT=(I),                        ;KEEP EMIT FOR CONSTANTS
12550            P3,     PS_D=(I),                               ;ZERO PS; D LEFT ZERO FROM

12551                                                            ;PREVIOUS TESTS IF ALL OK
12552            NEXT,   BUTO(SCOPE),                            ;NO ERROR: "TEST511A" (+11, WORDS)
12553                    J/TEST511A                              ;   ERROR: "BETONEB510A" (-20, WORDS)
     (6300)  DCS[0,00,0,1,0,0]  BM[1000,,00,00,,00,01,,010,,010,,,0,0,0,,0,,0,,,1,1011,,,0,,0000,0,,,11,000,,,110,000,111]
12554
12555
```

```
    ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

    !
    !       THIS FIRST SUBROUTINE COPIES:
    !               CSP(06) -> FPS<7:4>
    !               CSP(05) -> FLAGS<8:4,2:0>, EXFLAGS<2:3>
    !       THEN RETURNS
    !
    7073:  !(FREE)
    FLAGFPSEQLOD:
            P3-T,   D_CSPD(D06), D[C]_0,                    !GET VALUE TO LOAD TO FPS<7:4>H
            NEXT,   J/FLAGFPS02                             !
 (7073)  DCS[0,00,0,0,0,0]  BM[1010,,10,00,,00,00,,000,,000,,,1,1,0,,0,,0,,,0,1001,,,0,,0000,0,,,11,000,,,000,111,101]

    7075:  !(FREE)
    FLAGFPS02:
            P0,     BUSDIN_EMIT-[I],                        !
            P3-T,   FPS[7-4]_D[7-4]-[I],                    !LOAD FPS<7:4>H FROM D<7:4>H
            NEXT,   J/FLAGFPS03                             !
 (7075)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,01,,100,,000,,,1,0,0,,0,,0,,,1,1011,,,0,,0000,0,,,11,000,,,000,111,110]

    7076:  !(FREE)
    FLAGFPS03:
            P3-T,   D_CSPD(D05), D[C]_0,                    !GET VALUE TO LOAD TO FLAGS<8:0>H,
            NEXT,   J/FLAGFPS04                             !EXFLAGS<2:1>H
 (7076)  DCS[0,00,0,0,0,0]  BM[1010,,10,00,,00,00,,000,,000,,,1,1,0,,0,,0,,,0,1010,,,0,,0000,0,,,11,000,,,000,111,111]

    7077:  !(FREE)
    FLAGFPS04:
            P0,     BUSDIN_EMIT-[I],                        !KEEP IT ON
            P3-T,   FLAG[8-0]_D[15-8]-[I],                  !LOAD FLAGS FROM D
            NEXT,   BUTA(RETURN),                           !AND RETURN
                    J/BUTERROR7                             !
 (7077)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,01,,000,,001,,,1,0,0,,0,,0,,,1,1011,,,0,,0000,0,,,11,111,,,011,111,110]

    !
    !       THIS SECOND SUBROUTINE COPIES:
    !               CSP(05) -> PS<15:12>
    !               CSP(06) -> PS<7:4>
    !               CSP(07) -> PS<3:0>
    !       THEN RETURNS
    !
    7100:  !(FREE)
    PSSEQLOD:
            P3-T,   D_CSPD(D05), D[C]_0,                    !GET VALUE TO LOAD PS<15:12>H
            NEXT,   J/PSSEQLOD02                            !
 (7100)  DCS[0,00,0,0,0,0]  BM[1010,,10,00,,00,00,,000,,000,,,1,1,0,,0,,0,,,0,1010,,,0,,0000,0,,,11,000,,,001,000,001]

    7101:  !(FREE)
    PSSEQLOD02:
            P0,     BUSDIN_EMIT-[I],                        !
            P3-T,   PS[15-12]_D[15-12]-[I],                 !LOAD PS<15-12>H FROM D<15,12>H
            NEXT,   J/PSSEQLOD03                            !
```



```
 (7101)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,01,,000,,010,,,1,0,0,,0,,0,,,1,1011,,,0,,0000,0,,,11,000,,,001,000,010]

    7102:  !(FREE)
    PSSEQLOD03:
            P3-T,   D_CSPD(D06), D[C]_0,                    !GET VALUE TO LOAD PS<7:4>H
            NEXT,   J/PSSEQLOD04                            !
 (7102)  DCS[0,00,0,0,0,0]  BM[1010,,10,00,,00,00,,000,,000,,,1,1,0,,0,,0,,,0,1001,,,0,,0000,0,,,11,000,,,001,000,011]

    7103:  !(FREE)
    PSSEQLOD04:
            P0,     BUSDIN_EMIT-[I],                        !
            P3-T,   PS[7-4]_D[7-4]-[I],                     !LOAD PS<7:4>H FROM D<7:4>H
            NEXT,   J/PSSEQLOD05                            !
 (7103)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,01,,010,,000,,,1,0,0,,0,,0,,,1,1011,,,0,,0000,0,,,11,000,,,001,000,100]

    7104:  !(FREE)
    PSSEQLOD05:
            P3-T,   D_CSPD(D07), D[C]_0,                    !GET VALUE TO LOAD TO PS<3:0>H
            NEXT,   J/PSSEQLOD06                            !
 (7104)  DCS[0,00,0,0,0,0]  BM[1010,,10,00,,00,00,,000,,000,,,1,1,0,,0,,0,,,0,1000,,,0,,0000,0,,,11,000,,,001,000,101]

    7105:  !(FREE)
    PSSEQLOD06:
            P0,     BUSDIN_EMIT-[I],                        !KEEP IT ON
            P2-T,   PS[3-0]_D[3-0]-[I],                     !LOAD PS<3:0>H FROM D<3:0>H
            NEXT,   BUTA(RETURN),                           !AND RETURN
                    J/BUTERROR7                             !
 (7105)  DCS[0,00,0,0,0,0]  BM[1000,,00,00,,00,01,,000,,000,,,0,0,0,,0,,0,,,1,1011,,,0,,0000,0,,,11,111,,,011,111,110]





    !.PAGE=====================================================================================================


    .TOC    *    TEST511:  NFSS LOGIC TESTS


    !*************************************************************************************
    !*
    !*      TESTS:  511 A - B                               UWORDS:  022 + 060
    !*
    !*      FUNCTIONS:
    !*
    !*      THE FOLLOWING TESTS VERIFY THAT THE "NF SAME STACK" LOGIC OPERATES

    !*      CORRECTLY, AND THAT THE "SR6-H" DECODE IS CORRECT.
    !*
    !*************************************************************************************



    !
    !       SUMMARY OF "NF SAME STACK H" LOGIC TESTS
```

```
;
;       MF SAME STACK H = FLAG2-H + IR6-H + IR7-H + IR8-L + (PS15-H*PS13-H)-H
;
;       TEST    PS15,13H  FLAG2H  IR8-6H/8R6H  MFSS-H   DATA
;       ----    --------  ------  -----------  ------   ------
;
;        A1        0-0       1       110-1        1     002601
;
;        A2        0-1       1       110-1        0     022600
;
;        A3        1-1       1       110-1        1     122601
;
;        A4        1-0       1       110-1        0     102600
;
;
;        B1        0-0       0       110-1        0     000600
;
;        B2        0-0       1       010-0        0     002200
;
;        B3        0-0       1       100-0        0     002400
;
;        B4        0-0       1       111-0        0     002700
;
;
;
;       TESTING SUBR USED FOR ABOVE TESTS USES THE DATA AS FOLLOWS:
;
;       DATA<15,13> -> PS<15,13>,  DATA<10> -> FLAG<2>,
;       DATA<8:6> -> IR<8:6>,  DATA<0> = EXPECTED MFSS-H OUTPUT


    6232:  ;(FREE)
    MFS801:
            P2-U,    IR_DBUF-[I],                          ;DONT CARE ABOUT AFFECT, ONLY SET
            P3,      DBUF_D-[I],                           ;THESE UCONS UP
            NEXT,    J/MFS802                              ;
  (6232)  DCS[0,00,0,0,0,0]  BM[0100,,00,00,,00,01,,000,,100,,,0,0,0,,0,,0,,,1,1011,,,0,,0000,0,,,11,000,,,011,000,010]

    6302:  ;(FREE)
    MFS802:
            P2-T,    D_CSPB(B17), D[C]_ALU00,              ;GET DATA INTO D, D[C]_EXPECTED MFSS-H
            P2-U,    IR_DBUF,                              ;IGNORE FOR NOW
            P3,      DBUF_D,                               ;GET DATA INTO DBUF, TO GO TO IR NEXT
            NEXT,    J/MFS803                              ;
  (6302)  DCS[0,00,0,0,0,0]  BM[1010,,11,00,,00,00,,000,,010,,,0,1,0,,0,,0,,,1,1010,,,0,,0000,0,,,11,000,,,011,000,011]

    6303:  ;(FREE)
    MFS803:
            P0,      BUMP-VERIFY,                          ;COUNT
            P2-U,    IR_DBUF,                              ;SETUP IR<8:6> FROM DATA<8:6>
            P3,      DBUF_D,                               ;IGNORE FOR NOW
            NEXT,    J/MFS804                              ;
  (6303)  DCS[0,00,0,0,0,1]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,1,1010,,,0,,0000,0,,,11,000,,,011,000,100]
```



```
    6304:  ;(FREE)
    MFS804:
            P0,      BUSDIN_EMIT-[I],                      ;RESET
            P3,      FLAG(8-0)_D(15-8)-[I],                ;SETUP FLAG<2> FROM DATA<10>
                     PS(15-12)_D(15:13)-[I],               ;SETUP PS<15,13> FROM D<15,13>
            NEXT,    BUTR(D[C]-A),                         ;IF EXPECT MFSS-H=(1), J/MFS806
                     J/MFS805                              ;IF EXPECT MFSS-H=(0), J/MFS805
  (6304)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,01,,000,,011,,,0,0,0,,0,,0,,,1,1011,,,0,,0000,0,,,01,111,,,101,101,010]

    ;* COME HERE IF EXPECT MFSS-H=(0)
    6552:
    MFS805:
            P0,      LOAD-ENUA(MFSSEXPEC0),                ;SETUP FOR (0)
                     LOAD-ERROR(MFS805),                   ;ERROR DIRECTORY KEY
                     DCS-CTR(C1,),                         ;COMPARE AT TARGET
                     BUMP-VERIFY,                          ;COUNT
            NEXT,    BUTR(MFSS),                           ;TEST MFSS:  (0)=MFSSEXPEC0, (1)=MFSSEXPEC1
                     J/MFSSEXPEC0                          ;
  (6552)  DCS[1,00,1,0,0,1]  BM[1110,,00,11,,01,01,,100,,101,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,01,100,,,101,100,101]

    ;* COME HERE IF EXPECT MFSS-H=(1)
    6553:
    MFS806:
            P1,      LOAD-ENUA(MFSSEXPEC1),                ;SETUP FOR (0)
                     LOAD-ERROR(MFS806),                   ;ERROR DIRECTORY KEY
                     DCS-CTR(C1,),                         ;COMPARE AT TARGET
            NEXT,    BUTR(MFSS),                           ;TEST MFSS:  (0)=MFSSEXPEC0, (1)=MFSSEXPEC1
                     J/MFSSEXPEC0                          ;
  (6553)  DCS[1,00,1,0,0,0]  BM[1110,,00,11,,01,01,,100,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,01,100,,,101,100,101]


    ;* COME HERE IF MFSS-H TESTS AS A (0)
    6545:
    MFSSEXPEC0:
            P0,      BUSDIN_EMIT-[I],                      ;RESET PROC UCON
            NEXT,    BUTA(RETURN),                         ;AND RETURN
                     J/BUTERROR6                           ;*** COMPARE DONE HERE ***
  (6545)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,01,,000,,000,,,0,0,0,,0,,0,,,1,1001,,,0,,0000,0,,,11,111,,,011,111,110]

    ;* COME HERE IF MFSS-H TESTS AS A (1)
    6547:
    MFSSEXPEC1:
            P0,      BUSDIN_EMIT-[I],                      ;RESET PROC UCON
            NEXT,    BUTA(RETURN),                         ;AND RETURN
                     J/BUTERROR6                           ;*** COMPARE DONE HERE ***
  (6547)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,01,,000,,000,,,0,0,0,,0,,0,,,1,1001,,,0,,0000,0,,,11,111,,,011,111,110]


    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

    ;       THE TESTS ACTUALLY START HERE:

    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
```

```
	6607:
	TEST511A:
		P0,	LOAD-ERROR(TEST511A),		;ERROR DIRECTORY KEY
		P3,	CSPD[17]_EMIT, EMIT/002601,	;* TEST 511 A1 DATA *
		NEXT,	J/GOTEST511A1			;
 (6607)	DCS[1,00,0,0,0,0]	BM[0000,,10,01,,01,10,,000,,001,,,0,0,0,,0,,0,,,0,0000,,,1,,0000,0,,,11,000,,,101,101,000]

	6550:
	GOTEST511A1:
		SETUP,	RETURN/TEST511A2,		;GO DO THE TEST
		NEXT,	CALL[MPSS-TEST]			;
 (6550)	DCS[0,00,0,0,0,0]	BM[0111,,00,00,,01,11,,100,,110,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,011,010]


	7074:	;(FREE)
	TEST511A2:
		P3,	CSPD[17]_EMIT, EMIT/022600,	;* TEST 511 A2 DATA *
		NEXT,	J/GOTEST511A2			;
 (7074)	DCS[0,00,0,0,0,0]	BM[0010,,10,01,,01,10,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,1,,0000,0,,,11,000,,,001,000,111]

	7107:	;(FREE)
	GOTEST511A2:
		SETUP,	RETURN/TEST511A3,		;GO DO THE TEST
		NEXT,	CALL[MPSS-TEST]			;
 (7107)	DCS[0,00,0,0,0,0]	BM[0111,,00,00,,10,01,,000,,110,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,011,010]


	7110:	;(FREE)
	TEST511A3:
		P3,	CSPD[17]_EMIT, EMIT/122601,	;* TEST 511 A3 DATA *
		NEXT,	J/GOTEST511A3			;
 (7110)	DCS[0,00,0,0,0,0]	BM[1010,,10,01,,01,10,,000,,001,,,0,0,0,,0,,0,,,0,0000,,,1,,0000,0,,,11,000,,,001,001,001]

	7111:	;(FREE)
	GOTEST511A3:
		SETUP,	RETURN/TEST511A4,		;GO DO THE TEST
		NEXT,	CALL[MPSS-TEST]			;
 (7111)	DCS[0,00,0,0,0,0]	BM[0111,,00,00,,10,01,,010,,110,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,011,010]


	7112:	;(FREE)
	TEST511A4:
		P3,	CSPD[17]_EMIT, EMIT/102600,	;* TEST 511 A4 DATA *
		NEXT,	J/GOTEST511A4			;
 (7112)	DCS[0,00,0,0,0,0]	BM[1000,,10,01,,01,10,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,1,,0000,0,,,11,000,,,001,001,011]

	7113:	;(FREE)
	GOTEST511A4:
		SETUP,	RETURN/SCOPE511A,		;GO DO THE TEST
		NEXT,	CALL[MPSS-TEST]			;
 (7113)	DCS[0,00,0,0,0,0]	BM[0110,,00,01,,10,00,,001,,110,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,011,010]
```



```


	6301:	;(FREE)
	SCOPE511A:
		P3,	CSPD[17]_EMIT, EMIT/002601,	;RESET DATA FOR TEST 511 A1
		NEXT,	AUTO[SCOPE],			;NO ERROR:  "TEST511B1" (+1. WORDS)
			J/TEST511B1			;   ERROR:  "GOTEST511A1" (-7. WORDS)
 (6301)	DCS[0,00,0,1,0,0]	BM[0000,,10,01,,01,10,,000,,001,,,0,0,0,,0,,0,,,0,0000,,,1,,0000,0,,,11,000,,,101,101,001]




	; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -


	6551:
	TEST511B1:
		P3,	CSPD[17]_EMIT, EMIT/000600,	;* TEST 511 B1 DATA *
		NEXT,	J/GOTEST511B1			;
 (6551)	DCS[0,00,0,0,0,0]	BM[0000,,10,00,,01,10,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,1,,0000,0,,,11,000,,,101,101,100]

	6554:
	GOTEST511B1:
		SETUP,	RETURN/TEST511B2,		;GO DO THE TEST
		NEXT,	CALL[MPSS-TEST]			;
 (6554)	DCS[0,00,0,0,0,0]	BM[0111,,00,00,,10,00,,110,,110,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,011,010]


	7106:	;(FREE)
	TEST511B2:
		P3,	CSPD[17]_EMIT, EMIT/002200,	;* TEST 511 B2 DATA *
		NEXT,	J/GOTEST511B2			;
 (7106)	DCS[0,00,0,0,0,0]	BM[0000,,10,01,,00,10,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,1,,0000,0,,,11,000,,,001,001,101]

	7115:	;(FREE)
	GOTEST511B2:
		SETUP,	RETURN/TEST511B3,		;GO DO THE TEST
		NEXT,	CALL[MPSS-TEST]			;
 (7115)	DCS[0,00,0,0,0,0]	BM[0111,,00,00,,10,01,,110,,110,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,011,010]


	7116:	;(FREE)
	TEST511B3:
		P3,	CSPD[17]_EMIT, EMIT/002400,	;* TEST 511 B3 DATA *
		NEXT,	J/GOTEST511B3			;
 (7116)	DCS[0,00,0,0,0,0]	BM[0000,,10,01,,01,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,1,,0000,0,,,11,000,,,001,001,111]

	7117:	;(FREE)
	GOTEST511B3:
		SETUP,	RETURN/TEST511B4,		;GO DO THE TEST
		NEXT,	CALL[MPSS-TEST]			;
 (7117)	DCS[0,00,0,0,0,0]	BM[0111,,00,00,,10,10,,000,,110,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,011,010]
```

```
        7120:  !(FREE)
        TESTS11B4:
                P3,     CSPD[17]_EXIT, EXIT/002700,              !* TEST 511 B4 DATA *
                NEXT,   J/GOTESTS11B4                            !
 (7120) DCS[0,00,0,0,0,0]  BM[0000,,10,01,,01,11,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,1,,0000,0,,,11,000,,,001,010,001]

        7121:  !(FREE)
        GOTESTS11B4:
                SETUP,  RETURN/SCOPES11B,                        !GO DO THE TEST
                NEXT,   CALL[NFS6-TEST]                          !
 (7121) DCS[0,00,0,0,0,0]  BM[0110,,00,01,,10,00,,101,,110,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,011,010]


        6305:  !(FREE)
        SCOPES11B:
                P3,     CSPD[17]_EXIT, EXIT/000800,              !RESET DATA FOR TEST 511 B1
                NEXT,   BUTD[SCOPE],                             !NO ERROR;  "SETUPS12A" (+1, WORDS)
                        J/SETUPS12A                              !   ERROR;  "GOTESTS11B1" (-7, WORDS)
 (6305) DCS[0,00,0,1,0,0]  BM[0000,,10,00,,01,10,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,1,,0000,0,,,11,000,,,101,101,101]





        ;.PAGE=====================================================================================================


        .TOC    *    TESTS12:  KT SRC/DST ADDRESSING LOGIC TESTS



        !*****************************************************************************************
        !*
        !*      TESTS:  512 A - E                                UWORDS:  044 + 062
        !*
        !*      FUNCTIONS:
        !*
        !*      THE FOLLOWING TEN TESTS VERIFY THE KT-SRC/DST-ADDRS ROM OUTPUT AND INPUT LINES
        !*      FUNCTION CORRECTLY, IN RESPECT TO NO STUCK ONE/ZERO CONDITIONS.
        !*
        !*****************************************************************************************


        !       KT SRC/DST LOGIC EQUATIONS:      (IMPLEMENTED IN ROM)
        !
        !       KT-SRC-ADDRS-1 =        NOT-F2,AND,SR6,AND,PS15
        !                        .OR.   F2,AND,SR6,AND,NOT-SN0,AND,PS15
        !                        .OR.   F2,AND,SR6,AND,SN0,AND,PS13,AND,NOT-FLTPT
        !                        .OR.   SR6,AND,PS15,AND,FLTPT
        !
        !       KT-DST-ADDRS-1 =        NOT-F1,AND,DR6,AND,PS15
```



```
        !                        .OR.   F1,AND,DR6,AND,NOT-DN0,AND,PS15
        !                        .OR.   F1,AND,DR6,AND,DN0,AND,PS13,AND,NOT-FLTPT
        !                        .OR.   DR6,AND,PS15,AND,FLTPT
        !
        !
        !       SUMMARY OF KT ASP/BSP SRC/DST STACK POINTER ADDRESSING LOGIC:
        !
        !       TEST    PS      FLAG                                                  KT-SRC  KT-DST
        !       NUMB    15:13H  2:1H    IR      FLTL    SN0H    SR6H    DN0H    DR6H   ADR-H   ADR-H   NOW READ:
        !       ----    ------  ----    ------  ----    ----    ----    ----    ----   -----   -----   ------------
        !
        !       A1      1,0     1,1     172206  0       0       0       1       1       0              ASPHI[SF]=(02)
        !       A2                                                                             1       BSPHI[DF]=(16)
        !
        !       B1      0,1     1,0     070606  1       1       1       1       1               0      ASPHI[DF]=(05)
        !       B2                                                                     1               BSPHI[SF]=(16)
        !
        !       C1      0,1     1,1     134606  1       0       1       1       1               1      ASPHI[DF]=(16)
        !       C2                                                                     0               BSPHI[SF]=(06)
        !
        !       D1      1,0     0,1     160612  1       1       1       0       0       1              ASPHI[SF]=(16)
        !       D2                                                                             0       BSPHI[DF]=(02)
        !
        !       E1      1,0     1,1     150626  1       1       1       0       1               1      ASPHI[DF]=(16)
        !       E2                                                                     0               BSPHI[SF]=(06)
        !
        !
        !  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -
        !
        !
        !       KT SRC/DST STACK POINTER ADDRESS MODE TEST SUBROUTINE:
        !
        !       ENTER WITH:     CSP(17) = VALUE TO GO INTO IR, TO SETUP FLTPT/SN0/SR6/DN0/DR6
        !                       CSP(16) = BIT<15,13> => PS<15,13>
        !                                 BIT<10:09> => FLAG<2:1>
        !                              ** BIT<00> IS AN INTERNAL FLAG TO INDICATE WHICH REGISTER
        !                                     TO PUT IN THE SR ON EXIT:
        !                                     BIT<00> = (1) => ASPHI[DF],   BIT<00> = (0) => ASPHI[SF]
        !

        7114:  !(FREE)
        KTSRCDST01:
                P2-T,   D_CSPD(D17), D(C)_0,                     !INITIAL DATA TO GO INTO IR
                NEXT,   J/KTSRCDST02                             !
 (7114) DCS[0,00,0,0,0,0]  BM[1010,,10,00,,00,00,,000,,000,,,0,1,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,010,011]
1

        7123:  !(FREE)
        KTSRCDST02:
                P2-U,   IR_DBUF-[I],                             !IGNORE FOR NOW
                P3,     DBUF_D-[I],                              !COPY IR DATA FROM D -> DBUF
                NEXT,   J/KTSRCDST03                             !
 (7123) DCS[0,00,0,0,0,0]  BM[0100,,00,00,,00,01,,000,,100,,,0,0,0,,0,,0,,,1,1011,,,0,,0000,0,,,11,000,,,001,010,100]

        7124:  !(FREE)
```

```
        KTSRCDST03:
                P2-U,   IR_DBUF,                                        !COPY IR DATA FROM DBUF => IR
                P3-T,   D_CSPS(S16), D(C)_ALU00,                        !GET PS/FLAGS/REGISTER SELECT DATA
                P3,     DBUF_D,                                         !D[C] = REGISTER SELECT FLAG
                NEXT,   J/KTSRCDST04                                    !
 (7124) DCS[0,00,0,0,0,0] BM[1010..11.01..00,00..000..010...1.1.0..0..0...1.1010...0..0000,0...11,000...001,010,101]

        7125: !(FREE)
        KTSRCDST04:
                P0,     BUSDIN_EMIT-[I],                                !KEEP IT ON
                P3,     PS[15-12]_D[15:13]-[I],                         !SETUP PS<15,12> AS REQUIRED
                        FLAG[3-0]_D[15-8]-[I],                          !SETUP FLAG<2:1> AS REQUIRED
                NEXT,   J/KTSRCDST04B                                   !
 (7125) DCS[0,00,0,0,0,0] BM[0000..00.00..00,01..000..011...0.0.0..0..0...1,1011...0..0000,0...11,000...001,010,110]

        7126: !(FREE)
        KTSRCDST04B:
                                                                        !THIS WORD NEEDED FOR 1 UWORD DELAY FOR ROM TO SETTLE
                P2-T,   D_CSPD(C052525), SAVE-D[C],                     !DATA PATTERN IN SCRATCH PAD ADDRESS [02]
                P3,     A:BSPHI[02]_D,                                  !BIT<3:0> = (05)
                NEXT,   BUTR(D[C]-B),                                   !KEY=(1) => READ ASPHI[DF] => SR
                        J/KTSRCDST05                                    !KEY=(0) => READ ASPHI[SF] => SR
 (7126) DCS[0.00,0,0,0,0] BM[1010..10.10..00,00..101..111...0.1.0..0..0...0,0111...0..1011,0...10,011...011,011,101]

        !ENTER HERE IF D[C] CLEAR, SF SELECTED
        7335:
        KTSRCDST05:
                P2-T,   SR_ASPHI[SF],                                   !USE ASP/SF
                NEXT,   J/BUTSR3=0                                      !AND NOW GO CHECK WHOM WAS READ
 (7335) DCS[0,00,0,0,0,0] BM[1111..00,00..11,11..000..000...0.0.1..0..0...0.0000...0..0000,0...11,000...010,111,110]

        !ENTER HERE IF D[C] SET, DF SELECTED
        7337:
        KTSRCDST06:
                P2-T,   SR_ASPHI[DF],                                   !USE ASP/DF
                NEXT,   J/BUTSR3=0                                      !AND NOW GO CHECK WHOM WAS READ
 (7337) DCS[0,00,0,0,0,0] BM[1111..00,00..11,10..000..000...0.0.1..0..0...0.0000...0..0000,0...11,000...010,111,110]


        !* WE ALSO NEED TWO ENTRY POINTS TO READ BSP SF & DF:

        7127: !(FREE)
        KTSRCDST07:
                P2-T,   SR_BSPHI[SF],                                   !USE BSP/SF
                NEXT,   J/BUTSR3=0                                      !AND NOW GO CHECK WHOM WAS READ
 (7127) DCS[0,00,0,0,0,0] BM[1010..01,01..00,00..000..000...0.0.1..0..0...0.0000...0..0000,0...11,000...010,111,110]


        7130: !(FREE)
        KTSRCDST08:
                P2-T,   SR_BSPHI[DF],                                   !USE BSP/DF
                NEXT,   J/BUTSR3=0                                      !AND NOW GO CHECK WHOM WAS READ
 (7130) DCS[0,00,0,0,0,0] BM[1010..01,00..00,00..000..000...0.0.1..0..0...0.0000...0..0000,0...11,000...010,111,110]
```



```
        ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        !       *** KT SRC/DST ENTERS HERE ***

        ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -


        !THESE FIRST TWO WORDS DO SOME PRELIMINAR SETUP OF SCRATCHPAD LOCATIONS (06)/(16)
        6555:
        SETUP512A:
                P2-T,   D_CSPD(C125252), D[C]_0,                        !
                P3,     A:BSPHI[06]_D,                                  !KERNAL SP LOCATION:
                NEXT,   J/SETUP512B                                     !  BIT<3:0> = (12) = SP-ADDRESS(06)
 (6555) DCS[0,00,0,0,0,0] BM[1010..10,10..00,00..111..000...0.1.0..0..0...0,0110...0..1011,0...11,000...011,000,111]

        6307: !(FREE)
        SETUP512B:
                P2-T,   D_CSPD(C000000), D[C]_0,                        !
                P3,     A:BSPHI[16]_D,                                  !USER SP LOCATION:
                NEXT,   J/TEST512A1                                     !  BIT<3:0> = (00) = SP-ADDRESS(16)
 (6307) DCS[0,00,0,0,0,0] BM[1010..10,10..00,00..011..000...0.1.0..0..0...0.0100...0..1011,0...11,000...101,100,010]

                                                                        !A,BSPHI[02] = (052525)
                                                                        !  BIT<3:0> = (05) = SP-ADDRESS(02)

                                                                        !A,BSPHI[12] = (000152)
                                                                        !  BIT<3:0> = (12) = SP-ADDRESS(12)
                                                                        !  (NOT USED UNLESS ERROR)


        ! -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -

        !* TEST 512 A1 *

        !       TEST 512 A 1-2 SETS UP FOR: ASP-SF-ADDRESS=(02), BSP-DF-ADDRESS=(16)
        6542:
        TEST512A1:
                P0,     LOAD-ENUA(ETARGET405),                          !BIT<3:0> = (05) = SP-ADDRESS(02)
                        LOAD-ERROR(TEST512A1),                          !ERROR DIRECTORY KEY
                        DCS-CTR(C11,),                                  !COMPARE AT TARGET
                NEXT,   J/SETIRS12A1                                    !
 (6542) DCS[1,00,1,0,0,0] BM[0100..00,11..11,00..000..101...0.0.0..0..0...0.0000...0..0000,0...11,000...101,110,100]

        6564:
        SETIRS12A1:

                P0,     BUMP-VERIFY,                                    !COUNT
                P3,     CSPD[17]_EMIT, EMIT/172206,                     !SETUP IR: FLTPT/SM0/SR6/DM0/DP6
                NEXT,   J/SETPSFLAGS12A1                                !
 (6564) DCS[0,00,0,0,0,1] BM[1111..10,01..00,10..000..110...0.0.0..0..0...0.0000...1..0000,0...11,000...011,001,000]

        6310: !(FREE)
        SETPSFLAGS12A1:
                P3,     CSPD[16]_EMIT, EMIT/103000,                     !BIT<15,13> -> PS<15,13>,
```

```
                                                                  ;BIT<10:9>  => FLAGS<2:1>
          NEXT,    J/GOTEST512A1                                  ;BIT<00> IS REGISTER KEY (SEE SUBR)
(6310)  DCS[0,00,0,0,0,0]  BM[1000,,10,01,,10,00,,000,,000,,,0,0,0,,0,,0,,,0,0001,,,1,,0000,0,,,11,000,,,011,001,001]

     6311: ;(FREE)
     GOTEST512A1:
          SETUP,   RETURN/TEST512A2,                              ;GO EXEC KT SRC/DST TEST SUBR AT
          NEXT,    CALL[KTSRCDST]                                 ;  INITIALIZATION POINT
(6311)  DCS[0,00,0,0,0,0]  BM[0110,,00,10,,11,00,,100,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,001,001,100]



     ;* TEST 512A2 NOW READS THE COMPLEMENTARY REGISTER TO THAT USED IN TEST512A1, INTO THE SR
     6544:
     TEST512A2:
          P0,      LOAD-ENUA(ZTARGET400),                         ;BIT<3:0> = (00) = SP-ADDRESS(16)
                   LOAD-ERROR(TEST512A2),                         ;ERROR DIRECTORY KEY
                   DCS-CTR(C4,),                                  ;COMPARE AT TARGET
          NEXT,    J/GOTEST512A2                                  ;
(6544)  DCS[1,00,1,0,0,0]  BM[1011,,00,11,,11,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,011,001,010]

     6312: ;(FREE)
     GOTEST512A2:
          SETUP,   RETURN/SCOPE512A,                              ;READ REGISTER BSPHI[DF] TO SR,
          NEXT,    CALL[KTDSTBSP]                                 ;  THEN DO BUT(SR3=0)
(6312)  DCS[0,00,0,0,0,0]  BM[0110,,00,01,,10,01,,011,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,001,011,000]


     6313: ;(FREE)
     SCOPE512A:
          P0,      BUSDIN_EMIT-[T],                               ;RESET PROC UCON
                   EN-CLR-IR(15-00),                              ;
          NEXT,    BUTO[SCOPE],                                   ;NO ERROR:  "TEST512B1"  (+1, WORDS)
                   J/TEST512B1                                    ;   ERROR:  "SETIR512A1"  (-5, WORDS)
(6313)  DCS[0,00,0,1,0,0]  BM[0000,,00,00,,00,01,,000,,100,,,0,0,0,,0,,0,,,1,1001,,,0,,0000,0,,,11,000,,,101,110,101]




     ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

     ;* TEST 512 B: *

     ;       TEST 512 B 1-2 SETS UP FOR:  ASP-DF-ADDRESS=(06), BSP-SF-ADDRESS=(16)
     6565:
     TEST512B1:
          P0,      LOAD-ENUA(ZTARGET412),                         ;BIT<3:0> = (12) = SP-ADDRESS(06)
                   LOAD-ERROR(TEST512B1),                         ;ERROR DIRECTORY KEY
                   DCS-CTR(C11,),                                 ;COMPARE AT TARGET
          NEXT,    J/SETIR512B1                                   ;
(6565)  DCS[1,00,1,0,0,0]  BM[0100,,00,11,,11,00,,001,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,101,011,110]
```



```
     6536:
     SETIR512B1:
          P0,      BUMP-VERIFY,                                   ;COUNT
          P3,      CSPD[17]_EMIT, EMIT/070606,                    ;SETUP IR:  FLTPT/SR0/SR6/DR0/DR6
          NEXT,    J/SETPSFLAG512B1                               ;
(6536)  DCS[0,00,0,0,0,1]  BM[0111,,10,00,,01,10,,000,,110,,,0,0,0,,0,,0,,,0,0000,,,1,,0000,0,,,11,000,,,011,001,100]

     6314: ;(FREE)
     SETPSFLAG512B1:
          P3,      CSPD[16]_EMIT, EMIT/022001,                    ;BIT<15,13> => PS<15,13>,
                                                                  ;BIT<10:9>  => FLAGS<2:1>
          NEXT,    J/GOTEST512B1                                  ;BIT<00> IS REGISTER KEY (SEE SUBR)
(6314)  DCS[0,00,0,0,0,0]  BM[0010,,10,01,,00,00,,000,,001,,,0,0,0,,0,,0,,,0,0001,,,1,,0000,0,,,11,000,,,011,001,101]

     6315: ;(FREE)
     GOTEST512B1:
          SETUP,   RETURN/TEST512B2,                              ;GO EXEC KT SRC/DST TEST SUBR AT
          NEXT,    CALL[KTSRCDST]                                 ;  INITIALIZATION POINT
(6315)  DCS[0,00,0,0,0,0]  BM[0110,,00,10,,11,00,,110,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,001,001,100]



     ;* TEST 512B2 NOW READS THE COMPLEMENTARY REGISTER TO THAT USED IN TEST512B1, INTO THE SR
     6546:
     TEST512B2:
          P0,      LOAD-ENUA(ZTARGET400),                         ;BIT<3:0> = (00) = SP-ADDRESS(16)
                   LOAD-ERROR(TEST512B2),                         ;ERROR DIRECTORY KEY
                   DCS-CTR(C4,),                                  ;COMPARE AT TARGET
          NEXT,    J/GOTEST512B2                                  ;
(6546)  DCS[1,00,1,0,0,0]  BM[1011,,00,11,,11,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,011,001,110]

     6316: ;(FREE)
     GOTEST512B2:
          SETUP,   RETURN/TEST512C1,                              ;READ REGISTER BSPHI[SF] TO SR,
          NEXT,    CALL[KTSRCBSP]                                 ;  THEN DO BUT(SR3=0)
(6316)  DCS[0,00,0,0,0,0]  BM[0110,,00,10,,11,01,,110,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,001,010,111]






     ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

     ;* TEST 512 C: *

     ;       TEST 512 C 1-2 SETS UP FOR:  ASP-DF-ADDRESS=(16), BSP-SF-ADDRESS=(06)
     6556:
     TEST512C1:
          P0,      LOAD-ENUA(ZTARGET400),                         ;BIT<3:0> = (00) = SP-ADDRESS(16)
                   LOAD-ERROR(TEST512C1),                         ;ERROR DIRECTORY KEY
                   DCS-CTR(C11,),                                 ;COMPARE AT TARGET
```

```
                NEXT,   J/SETIRS12C1                                   ;
        (6556)  DCS[1,00,1,0,0,0]  BM[0100,,00,11,,11,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,011,001,111]

        6317:  ;(FREE)
        SETIRS12C1:
                P0,     BUMP-VERIFY,                                   ;COUNT
                P3,     CSPD[17]_EMIT, EMIT/134606,                    ;SETUP IR:  FLTPT/SR0/SR6/DR0/DR6
                NEXT,   J/SETPSFLAGS12C1                               ;
        (6317)  DCS[0,00,0,0,0,1]  BM[1011,,10,10,,01,10,,000,,110,,,0,0,0,,0,,0,,,0,0000,,,1,,0000,0,,,11,000,,,011,010,000]

        6320:  ;(FREE)
        SETPSFLAGS12C1:
                P3,     CSPD[16]_EMIT, EMIT/023001,                    ;BIT<15,13> -> PS<15,13>,
                                                                       ;BIT<10:9>  -> FLAGS<2:1>
                NEXT,   J/GOTEST512C1                                  ;BIT<00> IS REGISTER KEY (SEE SUBR)
        (6320)  DCS[0,00,0,0,0,0]  BM[0010,,10,01,,10,00,,000,,001,,,0,0,0,,0,,0,,,0,0001,,,1,,0000,0,,,11,000,,,011,010,001]

        6321:  ;(FREE)
        GOTEST512C1:
                SETUP,  RETURN/TEST512C2,                              ;GO EXEC KT SRC/DST TEST SUBR AT
                NEXT,   CALL[KTSRCDST]                                 ;  INITIALIZATION POINT
        (6321)  DCS[0,00,0,0,0,0]  BM[0110,,00,10,,11,10,,110,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,001,001,100]



        ;* TEST 512C2 NOW READS THE COMPLEMENTARY REGISTER TO THAT USED IN TEST512C1, INTO THE SR
        6566:
        TEST512C2:
                P0,     LOAD-ENUA(ZTARGET412),                         ;BIT<3:0> = (12) = SP-ADDRESS(06)
                        LOAD-ERROR(TEST512C2),                         ;ERROR DIRECTORY KEY
                        DCS-CTR(C4,),                                  ;COMPARE AT TARGET
                NEXT,   J/GOTEST512C2                                  ;
        (6566)  DCS[1,00,1,0,0,0]  BM[1011,,00,11,,11,00,,001,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,011,010,010]

        6322:  ;(FREE)
        GOTEST512C2:
                SETUP,  RETURN/SCOPE512C,                              ;READ REGISTER BSPHI(SF) TO SR,
                NEXT,   CALL[KTSRCBSP]                                 ;  THEN DO BUT[SR3=0]
        (6322)  DCS[0,00,0,0,0,0]  BM[0110,,00,01,,10,10,,011,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,001,010,111]


        6323:  ;(FREE)
        SCOPE512C:
                P0,     BUSDIN_EMIT-[I],                               ;RESET PROC UCON
                        EN-CLK-IR[15-00],                              ;
                NEXT,   BUTD[SCOPE],                                   ;NO ERROR:  "TEST512D1"  (+1, WORDS)
                        J/TEST512D1                                    ;  ERROR:  "SETIRS12D1"  (-11, WORDS)
        (6323)  DCS[0,00,0,1,0,0]  BM[0000,,00,00,,00,01,,000,,100,,,0,0,0,,0,,0,,,1,1001,,,0,,0000,0,,,11,000,,,101,011,111]
```



```
        ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        ;* TEST 512 D1 *

        ;       TEST 512 D 1-2 SETS UP FOR:  ASP=SP-ADDRESS=(16), BSP=DP-ADDRESS=(02)
        6537:
        TEST512D1:
                P0,     LOAD-ENUA(ZTARGET400),                         ;BIT<3:0> = (00) = SP-ADDRESS(16)
                        LOAD-ERROR(TEST512D1),                         ;ERROR DIRECTORY KEY
                        DCS-CTR(C11,),                                 ;COMPARE AT TARGET
                NEXT,   J/SETIRS12D1                                   ;
        (6537)  DCS[1,00,1,0,0,0]  BM[0100,,00,11,,11,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,110,010,110]

        6626:
        SETIRS12D1:
                P0,     BUMP-VERIFY,                                   ;COUNT
                P3,     CSPD[17]_EMIT, EMIT/160612,                    ;SETUP IR:  FLTPT/SR0/SR6/DR0/DR6
                NEXT,   J/SETPSFLAGS12D1                               ;
        (6626)  DCS[0,00,0,0,0,1]  BM[1110,,10,00,,01,10,,001,,010,,,0,0,0,,0,,0,,,0,0000,,,1,,0000,0,,,11,000,,,011,010,100]

        6324:  ;(FREE)
        SETPSFLAGS12D1:
                P3,     CSPD[16]_EMIT, EMIT/101000,                    ;BIT<15,13> -> PS<15,13>,
                                                                       ;BIT<10:9>  -> FLAGS<2:1>
                NEXT,   J/GOTEST512D1                                  ;BIT<00> IS REGISTER KEY (SEE SUBR)
        (6324)  DCS[0,00,0,0,0,0]  BM[1000,,10,00,,10,00,,000,,000,,,0,0,0,,0,,0,,,0,0001,,,1,,0000,0,,,11,000,,,011,010,101]

        6325:  ;(FREE)
        GOTEST512D1:
                SETUP,  RETURN/TEST512D2,                              ;GO EXEC KT SRC/DST TEST SUBR AT
                NEXT,   CALL[KTSRCDST]                                 ;  INITIALIZATION POINT
        (6325)  DCS[0,00,0,0,0,0]  BM[0110,,00,10,,11,11,,110,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,001,001,100]



        ;* TEST 512D2 NOW READS THE COMPLEMENTARY REGISTER TO THAT USED IN TEST512D1, INTO THE SR
        6576:
        TEST512D2:
                P0,     LOAD-ENUA(ZTARGET405),                         ;BIT<3:0> = (05) = SP-ADDRESS(02)
                        LOAD-ERROR(TEST512D2),                         ;ERROR DIRECTORY KEY
                        DCS-CTR(C4,),                                  ;COMPARE AT TARGET
                NEXT,   J/GOTEST512D2                                  ;
        (6576)  DCS[1,00,1,0,0,0]  BM[1011,,00,11,,11,00,,000,,101,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,011,010,110]

        6326:  ;(FREE)
        GOTEST512D2:

                SETUP,  RETURN/TEST512E1,                              ;READ REGISTER BSPHI(DF) TO SR,
                NEXT,   CALL[KTDSTBSP]                                 ;  THEN DO BUT[SR3=0]
        (6326)  DCS[0,00,0,0,0,0]  BM[0110,,00,10,,11,11,,111,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,001,011,000]
```

```
        ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        ;* TEST 512 E1 *

        ;       TEST 512 E 1-2 SETS UP FOR:  ASP=DF-ADDRESS=(16), BSP=SF-ADDRESS=(06)
        5577:
        TEST512E1:
                P0,     LOAD-ENUA(ZTARGET400),                  ;BIT<3:0> = (00) = SP-ADDRESS(16)
                        LOAD-ERROR(TEST512E1),                  ;ERROR DIRECTORY KEY
                        DCS-CTR(C11.),                          ;COMPARE AT TARGET
                NEXT,   J/SETIR512E1                            ;
  (5577)  DCS[1,00,1,0,0,0]  BM[0100,,00,11,,11,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,011,010,111]

        6327:   ;(FREE)
        SETIR512E1:
                P0,     BUMP-VERIFY,                            ;COUNT
                P1,     CSPD[17]_EMIT, EMIT/150626,             ;SETUP IR:  FLTPT/SM0/SR6/DM0/DR6
                NEXT,   J/SETPSFLAG512E1                        ;
  (6327)  DCS[0,00,0,0,0,1]  BM[1101,,10,00,,01,10,,010,,110,,,0,0,0,,0,,0,,,0,0000,,,1,,0000,0,,,11,000,,,011,011,000]

        6330:   ;(FREE)
        SETPSFLAG512E1:
                P3,     CSPD[16]_EMIT, EMIT/103001,             ;BIT<15,13> -> PS<15,13>,
                                                                ;BIT<10:9>  -> FLAGS<2:1>
                NEXT,   J/GOTEST512E1                           ;BIT<00> IS REGISTER KEY (SEE SUBR)
  (6330)  DCS[0,00,0,0,0,0]  BM[1000,,10,01,,10,00,,000,,001,,,0,0,0,,0,,0,,,0,0001,,,1,,0000,0,,,11,000,,,011,011,001]

        6331:   ;(FREE)
        GOTEST512E1:
                SETUP,  RETURN/TEST512E2,                       ;GO EXEC KT SRC/DST TEST SUBR AT
                NEXT,   CALL[KTSRCDST]                          ;  INITIALIZATION POINT
  (6331)  DCS[0,00,0,0,0,0]  BM[0110,,00,10,,10,11,,101,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,001,001,100]



        ;* TEST 512E2 NOW READS THE COMPLEMENTARY REGISTER TO THAT USED IN TEST512E1, INTO THE SR
        5535:
        TEST512E2:
                P0,     LOAD-ENUA(ZTARGET412),                  ;BIT<3:0> = (12) = SP-ADDRESS(06)
                        LOAD-ERROR(TEST512E2),                  ;ERROR DIRECTORY KEY
                        DCS-CTR(C4.),                           ;COMPARE AT TARGET
                NEXT,   J/GOTEST512E2                           ;
  (5535)  DCS[1,00,1,0,0,0]  BM[1011,,00,11,,11,00,,001,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,011,011,010]

        6332:   ;(FREE)
        GOTEST512E2:
                SETUP,  RETURN/SCOPE512E,                       ;READ REGISTER BSPHI[SF] TO SR,
                NEXT,   CALL[KTSRCBSP]                          ;  THEN DO BUT(BR)=0)
  (6332)  DCS[0,00,0,0,0,0]  BM[0110,,00,01,,10,11,,011,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,001,010,111]
```



```
        6333:   ;(FREE)
        SCOPE512E:
                P0,     BUSDIN_EMIT-[I],                        ;RESET PROC UCON
                        EN-CLK-IR[15-00],                       ;
                NEXT,   BUTD[SCOPE],                            ;NO ERROR:  "TEST520A"  (+1, WORDS)
                        J/TEST520A                              ;   ERROR:  "SETIR512E1"  (-11, WORDS)
  (6333)  DCS[0,00,0,1,0,0]  BM[0000,,00,00,,00,01,,000,,100,,,0,0,0,,0,,0,,,1,1001,,,0,,0000,0,,,11,000,,,110,010,111]




        ;.PAGE===================================================================================================

        .TOC    *   TEST520A-520E:  TESTING THE "INSTR BRANCH" ROM

        ;**********************************************************************************
        ;*
        ;*      TESTS:  520A - 520E                             UWORDS: 020 + 031
        ;*
        ;*      FUNCTIONS:
        ;*
        ;*      THE FOLLOWING FIVE TESTS VERIFY THE VALIDITY OF THE INSTRUCTION BRANCH ROM
        ;*      INPUTS AND OUTPUTS, IN REGARD TO NO STUCK ONE/ZERO CONDITIONS.
        ;*
        ;**********************************************************************************

        ;       SUMMARY OF INSTR BRANCH ROM TESTS:
        ;
        ;       TEST    IR<15,10:08>H   B== ?   ON:                     PS<3:0>H        INSTR BRANCH L
        ;       NUMB                                                    N Z V C
        ;
        ;       520A    1 0 1 0         BMI     C.IOR.Z=0               0 1 0 0         1, NEGATED
        ;       520B    0 1 1 0         BGT     Z.IOR.(N.XOR.V)=0       0 0 0 0         0, ASSERTED
        ;       520C    0 1 0 1         BLT     N.XOR.V=1               1 1 1 0         1, NEGATED
        ;       520D    1 0 1 1         BLOS    C.IOR.Z=1               0 0 0 1         0, ASSERTED
        ;       520E    1 1 1 1         BLO     C=1                     0 1 1 0         1, NEGATED
        ;




        ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        ;*** TEST 520A ***
        ;TEST-520-A SETS UP IR<15,10:08>H="1010", PS<3:0>H="0100",
        ;  AND THEN BUTS ON "INSTR BRANCH L"
        6627:
        TEST520A:
                P0,     LOAD-ENUA(ZTARGET403),                  ;NOT ASSERTED
```

```
                        LOAD-ERROR(TEST520A),                   ;ERROR DIRECTORY KEY
                        DCS-CTR(C8.),                           ;COMPARE ENUA:THUA AT TARGET
                NEXT,   J/UCON520A                              ;
     (6627)  DCS[1.00.1.0.0.0]  BN[0111..00.11..11.00..000..011...0.0.0..0..0...0.0000...0..0000.0...11.000...110.010.100]

        6624:
        UCON520A:
                SELECT, UCON-PROC,                              ;PROCESSOR UCON:
                ENABLE, BUSDIN_EMIT[15-00],                     ;  EMIT ON BUSDIN
                        EN-CLK-IR[15-00],                       ;  AND CLOCKING IR
                P0,     BUMP-VERIFY,                            ;COUNT
                        SET-UCON-CONTROL,                       ;WRITE CONTROLS
                NEXT,   J/SETUP520A                             ;
     (6624)  DCS[0.00.0.0.0.1]  BN[0000..00.00..00.01..000..100...0.0.0..0..0...1.1001...0..0000.0...11.000...011.011.100]

        6334:   ;(FREE)
        SETUP520A:
                P0,     BUMP-VERIFY,                            ;COUNT
                EMITC,  EMIT/101004,                            ;PS<3:0>H="0100"
                P2=U,   IR_EMIT,                                ;
                P3,     CSPD[05]_EMIT,                          ;IR<15,10:08>H="1010"
                NEXT,   J/GOTEST520A                            ;
     (6334)  DCS[0.00.0.0.0.1]  BN[1000..10.00..10.00..000..100...0.0.0..0..0...1.1010...1..0000.0...11.000...011.011.101]

        6335:   ;(FREE)
        GOTEST520A:
                SETUP,  RETURN/TEST520B,                        ;RETURN TO START OF NEXT SUBTEST
                NEXT,   GOTO-PAGE(7),                           ;GO TO CODE THAT LOADS PS[CC], AND
                        J/SUCBRTEST01                           ;  THEN BUTS ON THE "INSTR BRANCH" ROM OUTPUT
     (6335)  DCS[0.00.0.0.0.0]  BN[0110..00.11..11.10..100..111...0.0.0..0..0...0.0000...0..0000.0...11.100...001.010.010]




        ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        ;*** TEST 520B ***
        ;TEST-520-B SETS UP IR<15,10:08>H="0110", PS<3:0>H="0000",
        ;  AND THEN BUTS ON "INSTR BRANCH L"
        6764:
        TEST520B:
                P0,     LOAD-ENUA(ZTARGET402),                  ;ASSERTED
                        LOAD-ERROR(TEST520B),                   ;ERROR DIRECTORY KEY
                        DCS-CTR(C7.),                           ;COMPARE ENUA:THUA AT TARGET
                NEXT,   J/SETUP520B                             ;
     (6764)  DCS[1.00.1.0.0.0]  BN[1000..00.11..11.00..000..010...0.0.0..0..0...0.0000...0..0000.0...11.000...011.011.110]

        6336:   ;(FREE)
        SETUP520B:
                EMITC,  EMIT/003000,                            ;PS<3:0>H="0000"
                P2=U,   IR_EMIT,                                ;
                P3,     CSPD[05]_EMIT,                          ;IR<15,10:08>H="0110"
                NEXT,   J/GOTEST520B                            ;
```



```
     (6336)  DCS[0.00.0.0.0.0]  BN[0000..10.01..10.00..000..000...0.0.0..0..0...1.1010...1..0000.0...11.000...011.011.111]

        6337:   ;(FREE)
        GOTEST520B:
                SETUP,  RETURN/TEST520C,                        ;RETURN TO START OF NEXT SUBTEST
                NEXT,   GOTO-PAGE(7),                           ;GO TO CODE THAT LOADS PS[CC], AND
                        J/SUCBRTEST01                           ;  THEN BUTS ON THE "INSTR BRANCH" ROM OUTPUT
     (6337)  DCS[0.00.0.0.0.0]  BN[0110..00.11..11.11..010..111...0.0.0..0..0...0.0000...0..0000.0...11.100...001.010.010]




        ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        ;*** TEST 520C ***
        ;TEST-520-C SETS UP IR<15,10:08>H="0101", PS<3:0>H="1110",
        ;  AND THEN BUTS ON "INSTR BRANCH L"
        6772:
        TEST520C:
                P0,     LOAD-ENUA(ZTARGET403),                  ;NEGATED
                        LOAD-ERROR(TEST520C),                   ;ERROR DIRECTORY KEY
                        DCS-CTR(C7.),                           ;COMPARE ENUA:THUA AT TARGET
                        BUMP-VERIFY,                            ;COUNT
                NEXT,   J/SETUP520C                             ;
     (6772)  DCS[1.00.1.0.0.1]  BN[1000..00.11..11.00..000..011...0.0.0..0..0...0.0000...0..0000.0...11.000...011.100.000]

        6340:   ;(FREE)
        SETUP520C:
                EMITC,  EMIT/002416,                            ;PS<3:0>H="1110"
                P2=U,   IR_EMIT,                                ;
                P3,     CSPD[05]_EMIT,                          ;IR<15,10:08>H="0101"
                NEXT,   J/GOTEST520C                            ;
     (6340)  DCS[0.00.0.0.0.0]  BN[0000..10.01..01.00..001..110...0.0.0..0..0...1.1010...1..0000.0...11.000...011.100.001]

        6341:   ;(FREE)
        GOTEST520C:
                SETUP,  RETURN/TEST520D,                        ;RETURN TO START OF NEXT SUBTEST
                NEXT,   GOTO-PAGE(7),                           ;GO TO CODE THAT LOADS PS[CC], AND
                        J/SUCBRTEST01                           ;  THEN BUTS ON THE "INSTR BRANCH" ROM OUTPUT
     (6341)  DCS[0.00.0.0.0.0]  BN[0110..00.11..11.10..010..111...0.0.0..0..0...0.0000...0..0000.0...11.100...001.010.010]




        ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -


        ;*** TEST 520D ***
        ;TEST-520-D SETS UP IR<15,10:08>H="1011", PS<3:0>H="0001",
        ;  AND THEN BUTS ON "INSTR BRANCH L"
        6762:
        TEST520D:
                P0,     LOAD-ENUA(ZTARGET402),                  ;ASSERTED
```

```
                LOAD-ERROR(TESTS20D),                         ;ERROR DIRECTORY KEY
                DCS-CTR(C7,);                                 ;COMPARE ENUA:TNUA AT TARGET
        NEXT,   J/SETUPS20D                                   ;
(6767) DCS[1,00,1,0,0,0] BM[1000,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,011,100,010]

6342: ;(FREE)
SETUPS20D:
        EMITC,  EMIT/101401,                                  ;PS<3:0>H="0001"
        P2-U,   IR_EMIT,                                      ;
        P1,     CSPD[DS]_EMIT,                                ;IR<15,10:08>H="1011"
        NEXT,   J/GOTESTS20D                                  ;
(6342) DCS[0,00,0,0,0,0] BM[1000,,10,00,,11,00,,000,,001,,,0,0,0,,0,,0,,,1,1010,,,1,,0000,0,,,11,000,,,011,100,011]

6343: ;(FREE)
GOTESTS20D:
        SETUP,  RETURN/TESTS20E,                              ;RETURN TO START OF NEXT SUBTEST
        NEXT,   GOTO-PAGE(7);                                 ;GO TO CODE THAT LOADS PS[CC], AND
                J/SUCBRTEST01                                 ;  THEN SUTS ON THE "INSTR BRANCH" ROM OUTPUT
(6343) DCS[0,00,0,0,0,0] BM[0110,,00,11,,11,11,,101,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,001,010,010]




; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

;*** TEST 520E ***
;TEST-520-F SETS UP IR<15,10:08>H="1111", PS<3:0>H="0110",
;  AND THEN SUTS ON "INSTR BRANCH L"
6775:
TEST520E:
        P0,     LOAD-ENUA(ZTARGET403],                        ;NEGATED
                LOAD-ERROR(TESTS20E),                         ;ERROR DIRECTORY KEY
                DCS-CTR(C7,),                                 ;COMPARE ENUA:TNUA AT TARGET
        NEXT,   J/SETUPS20E                                   ;
(6775) DCS[1,00,1,0,0,0] BM[1000,,00,11,,11,00,,000,,011,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,011,100,100]

6344: ;(FREE)
SETUPS20E:
        P0,     BUMP-VERIFY,                                  ;COUNT
        EMITC,  EMIT/103406,                                  ;PS<3:0>H="0110"
        P2-U,   IR_EMIT,                                      ;
        P1,     CSPD[DS]_EMIT,                                ;IR<15,10:08>H="1111"
        NEXT,   J/GOTEST520F                                  ;
(6344) DCS[0,00,0,0,0,1] BM[1000,,10,01,,11,00,,000,,110,,,0,0,0,,0,,0,,,1,1010,,,1,,0000,0,,,11,000,,,011,100,101]

6345: ;(FREE)
GOTEST520E:
        SETUP,  RETURN/SCOPE520,                              ;RETURN TO SCOPE LOOP TEST WORD
        NEXT,   GOTO-PAGE(7);                                 ;GO TO CODE THAT LOADS PS[CC], AND
                J/SUCBRTEST01                                 ;  THEN SUTS ON THE "INSTR BRANCH" ROM OUTPUT
(6345) DCS[0,00,0,0,0,0] BM[0110,,00,01,,11,00,,110,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,001,010,010]
```



```
6346: ;(FREE)
SCOPE520:
        P0,     BUSDIN_EMIT-[I],                              ;RESET PROC UCONS
                EN-CLK-IR[15-00],                             ;
        NEXT,   BUTD[SCOPE],                                  ;NO ERROR:  "TEST533A"  [+4, WORDS]
                J/TEST533A                                    ;   ERROR:  "UCON520A"  [-15, WORDS]
(6346) DCS[0,00,0,1,0,0] BM[0000,,00,00,,00,01,,000,,100,,,0,0,0,,0,,0,,,1,1001,,,0,,0000,0,,,11,000,,,110,010,101]



; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

;       TESTING SUBR (COMMON CODE) FOR ABOVE TESTS

7122: ;(FREE)
SUCBRTEST01:
        P2-T,   D_CSPD(D05], D[C]_0,                          ;GET PATTERN
        NEXT,   J/SUCBRTEST02                                 ;
(7122) DCS[0,00,0,0,0,0] BM[1010,,10,00,,00,00,,000,,000,,,0,1,0,,0,,0,,,0,1010,,,0,,0000,0,,,11,000,,,001,011,010]

7132: ;(FREE)
SUCBRTEST02:
        P2-T,   PS(3-0]_D[3-0]-[I],                           ;INTO PS[CC]
        NEXT,   J/SUCBRTEST03                                 ;
(7132) DCS[0,00,0,0,0,0] BM[1000,,00,00,,00,01,,000,,000,,,0,0,0,,0,,0,,,1,1011,,,0,,0000,0,,,11,000,,,001,011,011]

7133: ;(FREE)
SUCBRTEST03:
        SELECT, UCON-PROC,                                    ;LEAVE WITH EMIT ON BUSDIN,
        ENABLE, BUSDIN_EMIT[15-00],                           ;  CLOCK IR ENABLED
                EN-CLK-IR[15-00],                             ;
        P0,     SET-UCON-CONTROL,                             ;WRITE CONTROLS
        NEXT,   J/BUTINSTRBRANCH                              ;
(7133) DCS[0,00,0,0,0,0] BM[0000,,00,00,,00,01,,000,,100,,,0,0,0,,0,,0,,,1,1001,,,0,,0000,0,,,11,000,,,011,101,110]





;.PAGE=================================================================================================


.TOC    +   TESTS33-537:  SHIFT TREE


;*************************************************************************************
;*

;*      TESTS:  533A - 537A                                   UWORDS: 170 + 044
;*
;*      FUNCTIONS:      TESTS 533A - 537A VERIFY THE DATA AND CONTROL PATHS
;*                      OF THE 3 LEVEL BARREL SHIFTER (SHIFT TREE).
;*
```

```
13515    !*************************************************************************
13516
13517
13518
13519
13520
13521
13522    ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
13523
13524    !*** TEST 533A ***
13525    !READ D DIRECTLY THRU "D[HI]&D[LO]" PORT OF AMUX[HI&LO], BMUX=CMUX/DIRECT
13526    !IN(0)(052652), OUT(052652)
13527    6629:
13528    TEST533A:
13529            P0,     LOAD-EMUA(ETARGET402),                  !SETUP FOR D = ZERO TEST
13530                    LOAD-ERROR(TEST533A),                   !ERROR DIRECTORY KEY
13531                    DC8-CTR(C6,);                           !COMPARE AT TARGET
13532            NEXT,   J/INIT533A
     (6629)  DC8(1,00,1,0,0,0)  BM(1001,,00,11,,11,00,,000,,010...0,0,0,,0,,0...0,0000...0,,0000,0,,,11,000,,,110,110,110)
13533
13534    66661
13535    INIT533A:
13536            P0,     BUMP-VERIFY,                            !COUNT
13537            P3,     C8PD[17]_EMIT,                          !GET INITIAL PATTERN FOR D
13538                    EMIT/052652,                            !"0101 0101 1010 1010"
13539            NEXT,   J/INITD533A
     (6666)  DC8(0,00,0,0,0,1)  BM(0101,,10,01,,01,10,,101,,010...0,0,0,,0,,0...0,0000...1,,0000,0,,,11,000,,,011,100,111)
13540
13541    6634:  !(FREE)
13542    INITD533A:
13543            P2=T,   D_C8PD(D17),                            !INITIAL D =(052652)
13544                    D[C]_ALU15;                             !SETUP D[C] FOR SHIFT = "0"
13545            NEXT,   J/COMP533A
     (6347)  DC8(0,00,0,0,0,0)  BM(1010,,10,00,,00,00,,000,,100...0,1,0,,0,,0...0,0000...0,,0000,0,,,11,000,,,011,101,000)
13546
13547    6350:  !(FREE)
13548    COMP533A:
13549            P0,     BUMP-VERIFY,                            !COUNT
13550            &ETUP,  D-DIRECT,                               !AMUX-BMUX-CMUX ALL DIRECT
13551            P2=T,   D_D-&HIFTED-XOR-C8PB(D17),              !COMPARE D-&HIFTED;EXPECTED, BITWISE
13552            NEXT,   J/GOBUT533A                             !EXPECTED =(052652)
     (6350)  DC8(0,00,0,0,0,1)  BM(0110,,11,00,,01,01,,000,,000...0,1,0,,0,,0...0,0000...0,,0000,0,,,11,000,,,011,101,001)
13553
13554    6351:  !(FREE)
13555    GOBUT533A:
13556            SETUP,  RETURN/TEST533B,                        !RETURN TO START OF NEXT SUBTEST
13557            NEXT,   GOTO-PAGE(7);                           !BUT TABLE IS ON PAGE 7
13558                    J/BUTD-IS-ZERO                          !GO TEST D IS ALL ZERO
     (6351)  DC8(0,00,0,0,0,0)  BM(0110,,00,11,,00,11,,011,,111...0,0,0,,0,,0...0,0000...0,,0000,0,,,11,100,,,011,100,001)
13559
13560
13561
13562
```



```
13563    ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
13564
13565    !*** TEST 533B ***
13566    !READ D DIRECTLY THRU "D[HI]&D[LO]" PORT OF AMUX[HI&LO], BMUX=CMUX/DIRECT
13567    !IN(0)(125125), OUT(125125)
13568    6633:
13569    TEST533B:
13570            P0,     LOAD-EMUA(ETARGET402),                  !SETUP FOR D=ZERO TEST
13571                    LOAD-ERROR(TEST533B),                   !ERROR DIRECTORY KEY
13572                    DC8-CTR(C6,);                           !COMPARE AT TARGET
13573            NEXT,   J/INIT533B
     (6633)  DC8(1,00,1,0,0,0)  BM(1001,,00,11,,11,00,,000,,010...0,0,0,,0,,0...0,0000...0,,0000,0,,,11,000,,,011,101,010)
13574
13575    6352:  !(FREE)
13576    INIT533B:
13577            P3,     C8PD[16]_EMIT,                          !GET INITIAL PATTERN FOR D
13578                    EMIT/125125,                            !"1010 1010 0101 0101"
13579            NEXT,   J/INITD533B
     (6352)  DC8(0,00,0,0,0,0)  BM(1010,,10,10,,10,01,,010,,101...0,0,0,,0,,0...0,0001...1,,0000,0,,,11,000,,,011,101,011)
13580
13581    6353:  !(FREE)
13582    INITD533B:
13583            P0,     BUMP-VERIFY,                            !COUNT
13584            P2=T,   D_C8PD(D16),                            !INITIAL D=(125125)
13585                    D[C]_ALU07;                             !SETUP D[C] FOR SHIFT = "0"
13586            NEXT,   J/COMP533B
     (6353)  DC8(0,00,0,0,0,1)  BM(1010,,10,00,,00,00,,000,,011...0,1,0,,0,,0...0,0001...0,,0000,0,,,11,000,,,011,101,100)
13587
13588    6354:  !(FREE)
13589    COMP533B:
13590            &ETUP,  D-DIRECT,                               !AMUX-BMUX-CMUX ALL DIRECT
13591            P2=?,   D_D-&HIFTED-XOR-C8PB(B16),              !COMPARE D-&HIFTED;EXPECTED, BITWISE
13592            NEXT,   J/GOBUT533B                             !EXPECTED=(125125)
     (6354)  DC8(0,00,0,0,0,0)  BM(0110,,11,01,,01,01,,000,,000...0,1,0,,0,,0...0,0000,,,0,,0000,0,,,11,000,,,011,101,101)
13593
13594    6355:  !(FREE)
13595    GOBUT533B:
13596            SETUP,  RETURN/SCOPE533B,                       !RETURN TO SCOPE LOOP TEST WORD
13597            NEXT,   GOTO-PAGE(7),                           !BUT TABLE IS ON PAGE 7
13598                    J/BUTD-IS-ZERO                          !GO TEST D IS ALL ZERO
     (6355)  DC8(0,00,0,0,0,0)  BM(0110,,00,01,,11,01,,110,,111...0,0,0,,0,,0...0,0000...0,,0000,0,,,11,100,,,011,100,001)
13599
13600    6356:  !(FREE)
13601    SCOPE533B:
13602            P3,     C8PD[14]_EMIT, EMIT/000177,             !CONSTANT FOR USE BELOW
13603            NEXT,   BUTD[SCOPE];                            !NO ERROR: "TEST534A" (+1,WORD8)

13604                    J/TEST534A                              !   ERROR: "INIT533A" (-9,WORD8)
     (6356)  DC8(0,00,0,1,0,0)  BM(0000,,10,00,,00,11,,111,,111...0,0,0,,0,,0...0,0011...1,,0000,0,,,11,000,,,110,110,111)
13605
13606
13607
13608
13609    ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
```

```
        ;*** TEST 534A ***
        ;READ D THRU "D[LO]!D[HI]" PORT OF AMUX[HI,LO], BMUX=CMUX/DIRECT
        ;IN(0)(052652), OUT(125125)
        6667:
        TEST534A:
                P0,     LOAD-ENUA(ZTARGET402),                  ;SETUP FOR D=ZERO TEST
                        LOAD-ERROR(TEST534A),                   ;ERROR DIRECTORY KEY
                        DCS-CTR(C5,);                           ;COMPARE AT TARGET
                P3,     BUTA(CLR-FLAG-RES-UCON),                ;EXIT ON BUSDIN
                NEXT,   J/INITD534A                             ;
  (6667) DCS[1,00,1,0,0,0] BM[1010,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,010,,,110,110,100]

        6664:
        INITD534A:
                P0,     BUMP-VERIFY,                            ;COUNT
                P2-T,   D_CSPD(D17),                            ;INITIAL D=(052652)
                        D[C]_ALU15,                             ;SETUP D[C] FOR SHIFT = "0"
                NEXT,   J/COMP534A                              ;
  (6664) DCS[0,00,0,0,0,1] BM[1010,,10,00,,00,00,,000,,100,,,0,1,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,011,101,111]

        6357: ;(FREE)
        COMP534A:
                SETUP,  D-SWAB,                                 ;AMUX/SWAB, BMUX=CMUX/DIRECT
                P2-T,   D_D-SHIFTED-XOR-CSPD(B16),              ;COMPARE D-SHIFTED!EXPECTED, BITWISE
                NEXT,   J/GOBUT534A                             ;EXPECTED=(125125)
  (6357) DCS[0,00,0,0,0,0] BM[0110,,11,01,,01,01,,000,,000,,,0,1,0,,0,,0,,,0,0101,,,0,,0000,0,,,11,000,,,011,110,000]

        6360: ;(FREE)
        GOBUT534A:
                SETUP,  RETURN/TEST534B,                        ;RETURN TO START OF NEXT SUBTEST
                NEXT,   GOTO-PAGE(7);                           ;BUT TABLE IS ON PAGE 7
                        J/BUTD-IS-ZERO                          ;GO TEST D IS ALL ZERO
  (6360) DCS[0,00,0,0,0,0] BM[0110,,00,11,,00,11,,100,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,100,001]




        ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        ;*** TEST 534B ***
        ;READ D THRU "D[LO]!D[HI]" PORT OF AMUX[HI,LO], BMUX=CMUX/DIRECT
        ;IN(0)(125125), OUT(052652)
        6634:
        TEST534B:
                P0,     LOAD-ENUA(ZTARGET402),                  ;SETUP FOR D=ZERO TEST
                        LOAD-ERROR(TEST534B),                   ;ERROR DIRECTORY KEY
                        DCS-CTR(C5,);                           ;COMPARE AT TARGET
                        BUMP-VERIFY,                            ;COUNT
                NEXT,   J/INITD534B                             ;
  (6634) DCS[1,00,1,0,0,1] BM[1010,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,011,110,001]

        6361: ;(FREE)
```



```
        INITD534B:
                P2-T,   D_CSPD(D16),                            ;INITIAL D=(125125)
                        D[C]_ALU07,                             ;SETUP D[C] FOR SHIFT = "0"
                NEXT,   J/COMP534B                              ;
  (6361) DCS[0,00,0,0,0,0] BM[1010,,10,00,,00,00,,000,,011,,,0,1,0,,0,,0,,,0,0001,,,0,,0000,0,,,11,000,,,011,110,010]

        6362: ;(FREE)
        COMP534B:
                P0,     BUMP-VERIFY,                            ;COUNT
                SETUP,  D-SWAB,                                 ;AMUX/SWAB, BMUX=CMUX/DIRECT
                P2-T,   D_D-SHIFTED-XOR-CSPD(B17),              ;COMPARE D-SHIFTED!EXPECTED, BITWISE
                NEXT,   J/GOBUT534B                             ;EXPECTED=(052652)
  (6362) DCS[0,00,0,0,0,1] BM[0110,,11,00,,01,01,,000,,000,,,0,1,0,,0,,0,,,0,0101,,,0,,0000,0,,,11,000,,,011,110,011]

        6363: ;(FREE)
        GOBUT534B:
                SETUP,  RETURN/SCOPE534B,                       ;RETURN TO SCOPE LOOP TEST WORD
                NEXT,   GOTO-PAGE(7);                           ;BUT TABLE IS ON PAGE 7
                        J/BUTD-IS-ZERO                          ;GO TEST D IS ALL ZERO
  (6363) DCS[0,00,0,0,0,0] BM[0110,,00,01,,11,10,,100,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,100,001]

        6364: ;(FREE)
        SCOPE534B:
                P3,     CSPD[15]_EMIT, EMIT/177400,             ;CONSTANT FOR USE BELOW
                NEXT,   BUTD[SCOPE],                            ;NO ERROR: "TEST534C" (+1,WORDS)
                        J/TEST534C                              ;   ERROR: "INITD534A" (-7,WORDS)
  (6364) DCS[0,00,0,1,0,0] BM[1111,,10,11,,11,00,,000,,000,,,0,0,0,,0,,0,,,0,0010,,,1,,0000,0,,,11,000,,,110,110,101]




        ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        ;*** TEST 534C ***
        ;READ D THRU "8-D(C)!D[HI]" PORT OF AMUX[HI!LO], BMUX=CMUX/DIRECT
        ;IN(1)(000000), OUT(177400)
        6665:
        TEST534C:
                P0,     LOAD-ENUA(ZTARGET402),                  ;SETUP FOR D=ZERO TEST
                        LOAD-ERROR(TEST534C),                   ;ERROR DIRECTORY KEY
                        DCS-CTR(C5,);                           ;COMPARE AT TARGET
                NEXT,   J/INITD534C                             ;
  (6665) DCS[1,00,1,0,0,0] BM[1010,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,110,110,010]

        6662:
        INITD534C:

                P2-T,   D_ASPHI(C000000),                       ;INITIAL D=(000000)
                        D[C]_1,                                 ;SETUP D[C] FOR SHIFT = "1"
                NEXT,   J/COMP534C                              ;
  (6662) DCS[0,00,0,0,0,0] BM[1111,,00,00,,11,01,,100,,000,,,0,1,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,011,110,101]

        6365: ;(FREE)
        COMP534C:
```

```
                                                              ;CSP(15)=(177400)="1111 1111 0000 0000"
          SETUP,  D=RIGHT=8,                                  ;AMUX/@*D[C]:D<HI>, BMUX=CMUX/DIRECT
          P2-T,   D_D-SHIFTED-XOR-CSPB(B15),                  ;COMPARE D=SHIFTED:EXPECTED, BITWISE
          NEXT,   J/GOBUT534C                                 ;EXPECTED=(177400)
   (6365) DCS[0,00,0,0,0,0]  BM[0110,,11,10,,01,01,,000,,000,,,0,1,0,,0,,0,,,0,0110,,,0,,0000,0,,,11,000,,,011,110,110]

   6366: ;(FREE)
   GOBUT534C:
          SETUP,  RETURN/TEST534D,                            ;RETURN TO START OF  NEXT SUBTEST
          NEXT,   GOTO-PAGE(7),                               ;BUT TABLE IS ON PAGE 7
                  J/BUTD-IS-ZERO                              ;GO TEST  D IS ALL ZERO
   (6366) DCS[0,00,0,0,0,0]  BM[0110,,00,11,,00,11,,101,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,100,001]




   ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

   ;*** TEST 534D ***
   ;READ D THRU "8*D[C]:D[LO]" PORT OF AMUX[HI:LO], BMUX=CMUX/DIRECT
   ;IN(0)(177777), OUT(000377)
   6635:
   TEST534D:
          P0,     LOAD-ENUA(ZTARGET402),                      ;SETUP FOR D=ZERO TEST
                  LOAD-ERROR(TEST534D),                       ;ERROR DIRECTORY KEY
                  DCS-CTR(C5,),                               ;COMPARE AT TARGET
          NEXT,   J/INITD534D                                 ;
   (6635) DCS[1,00,1,0,0,0]  BM[1010,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,011,110,111]

   6367: ;(FREE)
   INITD534D:
          P2-T,   D_BSPHI(C177777),                           ;INITIAL D=(177777)
                  D[C]_0,                                     ;SETUP D[C] FOR SHIFT = "0"
          NEXT,   J/COMP534D                                  ;
   (6367) DCS[0,00,0,0,0,0]  BM[1010,,01,11,,00,00,,101,,000,,,0,1,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,011,111,000]

   6370: ;(FREE)
   COMP534D:
                                                              ;CSP(14)=(000377)="0000 0000 1111 1111"
          SETUP,  D=SIGNEXT,                                  ;AMUX/@*D[C]:D<LO>, BMUX=CMUX/DIRECT
          P2-T,   D_D-SHIFTED-XOR-CSPB(B14),                  ;COMPARE D=SHIFTED:EXPECTED, BITWISE
          NEXT,   J/GOBUT534D                                 ;EXPECTED=(000377)
   (6370) DCS[0,00,0,0,0,0]  BM[0110,,11,11,,01,01,,000,,000,,,0,1,0,,0,,0,,,0,0010,,,0,,0000,0,,,11,000,,,011,111,001]

   6371: ;(FREE)
   GOBUT534D:
          P0,     BUMP-VERIFY,                                ;COUNT
          SETUP,  RETURN/SCOPE534D,                           ;RETURN TO SCOPE LOOP TEST WORD
          NEXT,   GOTO-PAGE(7),                               ;BUT TABLE IS ON PAGE 7
                  J/BUTD-IS-ZERO                              ;GO TEST D IS ALL ZERO
   (6371) DCS[0,00,0,0,0,1]  BM[0110,,00,01,,11,11,,010,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,100,001]
```



```
   6372: ;(FREE)
   SCOPE534D:
          NEXT,   BUTD[SCOPE],                                ;NO ERROR:  "TEST534E" (+1,WORDS)
                  J/TEST534E                                  ;   ERROR:  "EXPEC534C" (-9,WORDS)
   (6372) DCS[0,00,0,1,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,110,110,011]


          .

   ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

   ;*** TEST 534E ***
   ;READ D THRU "COUNTER:D[LO]" PORT OF AMUX(HI:LO], BMUX=CMUX/DIRECT
   ;IN(0)(000125), OUT(125125), CTR(252)
   6663:
   TEST534E:
          P0,     LOAD-ENUA(ZTARGET402),                      ;SETUP FOR D =ZERO TEST
                  LOAD-ERROR(TEST534E),                       ;ERROR DIRECTORY KEY
                  DCS-CTR(C6,),                               ;COMPARE AT TARGET
          NEXT,   J/LODCNTR534E                               ;
   (6663) DCS[1,00,1,0,0,0]  BM[1001,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,110,110,000]

   6660:
   LODCNTR534E:
          P0,     BUMP-VERIFY,                                ;COUNT
          P2,     COUNTER_BSPHI(C125252),                     ;PUT A (252) IN BM COUNTER
          NEXT,   J/INITD534E                                 ;(GETS B=BUS(7:0))
   (6660) DCS[0,00,0,0,0,1]  BM[0000,,01,11,,00,00,,110,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0010,1,,,11,000,,,011,111,011]

   6373: ;(FREE)
   INITD534E:
          P0,     BUMP-VERIFY,                                ;COUNT
          P2-T,   D_CSPD(14)-AND-ASPHI(C052525),              ;INITIAL D=(000125), CSP(14)=(000377)
                  D[C]_ALUIS,                                 ;SETUP D[C] FOR SHIFT = "0"
          NEXT,   J/COMP534E                                  ;
   (6373) DCS[0,00,0,0,0,1]  BM[1011,,10,00,,11,01,,111,,100,,,0,1,0,,0,,0,,,0,0011,,,0,,0000,0,,,11,000,,,011,111,100]

   6374: ;(FREE)
   COMP534E:
          SETUP,  COUNT:D[LO],                                ;AMUX/COUNTER:D<LO> BMUX=CMUX/DIRECT
          P2-T,   D_D-SHIFTED-XOR-CSPB(B16),                  ;COMPARE D=SHIFTED:EXPECTED, BITWISE
          NEXT,   J/GOBUT534E                                 ;EXPECTED=(125125)
   (6374) DCS[0,00,0,0,0,0]  BM[0110,,11,01,,01,01,,000,,000,,,0,1,0,,0,,0,,,0,0011,,,0,,0000,0,,,11,000,,,011,111,101]

   6375: ;(FREE)
   GOBUT534E:

          SETUP,  RETURN/TEST534F,                            ;RETURN TO START OF NEXT SUBTEST
          NEXT,   GOTO-PAGE(7),                               ;BUT TABLE IS ON PAGE 7
                  J/BUTD-IS-ZERO                              ;GO TEST D IS ALL ZERO
   (6375) DCS[0,00,0,0,0,0]  BM[0110,,00,11,,00,11,,110,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,100,001]
```

```
        ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        !*** TEST 534F ***
        !READ D THRU "COUNTER&D(HI)" PORT OF AMUX(HI&LO), BMUX=CMUX/DIRECT
        !IN(0)(175000), OUT(052652), CTR(125)
        6636:
        TEST534F:
                P0,     LOAD-ENUA(ZTARGET402),                  !SETUP FOR D=ZERO TEST
                        LOAD-ERROR(TEST534F),                   !ERROR DIRECTORY KEY
                        DCS-CTR(C6,),                           !COMPARE AT TARGET
                NEXT,   J/LODCNTR534F                           !
  (6636)  DCS[1,00,1,0,0,0]  BM[1001,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,100,000,000]

        6400:  !(FREE)
        LODCNTR534F:
                P2,     COUNTER_ASPHI(C052525),                 !PUT A (125) IN BM COUNTER
                NEXT,   J/INITD534F                             !(GETS B-BUS(7-0))
  (6400)  DCS[0,00,0,0,0,0]  BM[0000,,01,11,,00,00,,111,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0010,1,,,11,000,,,100,000,001]

        6401:  !(FREE)
        INITD534F:
                P0,     BUMP-VERIFY,                            !COUNT
                P2-7,   D_CSPD(15)-AND-ASPHI(C125252),          !INITIAL D=(125000), CSP(15)=(177400)
                        D[C]_ALU00,                             !SETUP D[C] FOR SHIFT = "0"
                NEXT,   J/COMP534F                              !
  (6401)  DCS[0,00,0,0,0,1]  BM[1011,,10,00,,11,01,,110,,010,,,0,1,0,,0,,0,,,0,0010,,,0,,0000,0,,,11,000,,,100,000,010]

        6402:  !(FREE)
        COMP534F:
                SETUP,  COUNT&D[HI],                            !AMUX/COUNTER&D<HI>, BMUX=CMUX/DIRECT
                P2-7,   D_D-SHIFTED-XOR-CSPB(B17),              !COMPARE D-SHIFTED:EXPECTED, BITWISE
                P3,     BUTA(LAST),                             !CLEAR CSR TO (000) DURING P3
                NEXT,   J/GOBUT534F                             !EXPECTED=(052652)
  (6402)  DCS[0,00,0,0,0,0]  BM[0110,,11,00,,01,01,,000,,000,,,0,1,0,,0,,0,,,0,0111,,,0,,0000,0,,,10,000,,,100,000,011]

        6403:  !(FREE)
        GOBUT534F:
                SETUP,  RETURN/TEST534G,                        !RETURN TO START OF NEXT SUBTEST
                NEXT,   GOTO-PAGE(7),                           !BUT TABLE IS ON PAGE 7
                        J/BUTD-IS-ZERO                          !GO TEST D IS ALL ZERO
  (6403)  DCS[0,00,0,0,0,0]  BM[0110,,00,11,,11,01,,010,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,100,001]



        ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        !*** TEST 534G ***
        !READ D THRU "COUNTER&D[LO]" PORT OF AMUX(HI&LO), BMUX=CMUX/DIRECT
        !COUNTER AND D SHOULD READ ALL ZERO, AFTER BEING ZAPPED IN TEST-534-F
        6752:
        TEST534G:
                P0,     LOAD-ENUA(ZTARGET402),                  !SETUP FOR D=ZERO TEST
```



```
                        LOAD-ERROR(TEST534G),                   !ERROR DIRECTORY KEY
                        DCS-CTR(C4,),                           !COMPARE AT TARGET
                NEXT,   J/COUNTER534G                           !
  (6752)  DCS[1,00,1,0,0,0]  BM[1011,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,100,000,100]

        6404:  !(FREE)
        COUNTER534G:
                P2-T,   D_COUNT&D[LO],                          !READ THE MUX INTO D
                NEXT,   J/GOBUT534G                             !
  (6404)  DCS[0,00,0,0,0,0]  BM[1111,,00,00,,01,01,,000,,000,,,0,1,0,,0,,0,,,0,0011,,,0,,0000,0,,,11,000,,,100,000,101]

        6405:  !(FREE)
        GOBUT534G:
                P0,     BUMP-VERIFY,                            !COUNT
                SETUP,  RETURN/SCOPE534G,                       !RETURN TO SCOPE LOOP TEST WORD
                NEXT,   GOTO-PAGE(7),                           !BUT TABLE
                        J/BUTD-IS-ZERO                          !GO TEST D IS ALL ZERO
  (6405)  DCS[0,00,0,0,0,1]  BM[0110,,00,10,,00,00,,110,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,100,001]




        6406:  !(FREE)
        SCOPE534G:
                NEXT,   BUTD[SCOPE],                            !NO ERROR:  "TEST535A" (+1,WORDS)
                        J/TEST535A                              !   ERROR:  "LODCNTR534F"(-11,WORDS)
  (6406)  DCS[0,00,0,1,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,110,110,001]




        ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        !*** TEST 535A ***
        !READ D THRU "4*D[C]#AMUX<15:04>" PORT OF BMUX, AMUX=CMUX/DIRECT
        !IN(1)(122645), OUT(175132)
        6661:
        TEST535A:
                P0,     LOAD-ENUA(ZTARGET402),                  !SETUP FOR D=ZERO TEST
                        LOAD-ERROR(TEST535A),                   !ERROR DIRECTORY KEY
                        DCS-CTR(C7,),                           !COMPARE AT TARGET
                        BUMP-VERIFY,                            !COUNT
                NEXT,   J/INIT535A                              !
  (6661)  DCS[1,00,1,0,0,1]  BM[1000,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,110,101,110]

        6656:

        INIT535A:
                P3,     CSPD(15)_EMIT,                          !GET INITIAL PATTERN FOR D
                        EMIT/122645,                            !"1010 0101 1010 0101"
                NEXT,   J/EXPEC535A                             !
  (6656)  DCS[0,00,0,0,0,0]  BM[1010,,10,01,,01,10,,100,,101,,,0,0,0,,0,,0,,,0,0010,,,1,,0000,0,,,11,000,,,100,000,111]

        6407:  !(FREE)
```

```
	EXPEC535A:
		P3,	CSPD[14]_EMIT,				;GET EXPECTED PATTERN AFTER SHIFT
			EMIT/175132,				;"1111 1010 0101 1010"
		NEXT,	J/INITD535A				;
 (6407)	DCS[0,00,0,0,0,0]  BM[1111,,10,10,,10,01,,011,,010,,,0,0,0,,0,,0,,,0,0011,,,1,,0000,0,,,11,000,,,100,001,000]

	6410:	;(FREE)
	INITD535A:
		P0,	BUMP-VERIFY,				;COUNT
		P2-T,	D_CSPD(D15),				;INITIAL D=(122645)
			D[C]_ALU15,				;SETUP D[C] FOR SHIFT = "1"
		NEXT,	J/COMP535A				;
 (6410)	DCS[0,00,0,0,0,1]  BM[1010,,10,00,,00,00,,000,,100,,,0,1,0,,0,,0,,,0,0010,,,0,,0000,0,,,11,000,,,100,001,001]

	6411:	;(FREE)
	COMP535A:
		SETUP,	D-RIGHT-4,				;AMUX/DIRECT, BMUX/RIGHT-4, CMUX/DIRECT
		P2-T,	D_D-SHIFTED-XOR-CSPB(B14),		;COMPARE D-SHIFTED:EXPECTED, BITWISE
		NEXT,	J/GOBUT535A				;EXPECTED=(175132)
 (6411)	DCS[0,00,0,0,0,0]  BM[0110,,11,11,,01,01,,000,,000,,,0,1,0,,0,,0,,,0,1000,,,0,,0000,0,,,11,000,,,100,001,010]

	6412:	;(FREE)
	GOBUT535A:
		SETUP,	RETURN/TEST535B,			;RETURN TO START OF NEXT SUBTEST
		NEXT,	GOTO-PAGE(7),				;BUT TABLE IS ON PAGE 7
			J/BUTD-IS-ZERO				;GO TEST D IS ALL ZERO
 (6412)	DCS[0,00,0,0,0,0]  BM[0110,,00,11,,00,11,,111,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,100,001]

	; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

	;*** TEST 535B ***
	;READ D THRU "4*D(C)!AMUX<15:04>" PORT OF BMUX, AMUX-CMUX/DIRECT
	;IN(0)(055132), OUT (002645)
	6637:
	TEST535B:
		P0,	LOAD-ENUA(ETARGET402),			;SETUP FOR D=ZERO TEST
			LOAD-ERROR(TEST535B),			;ERROR DIRECTORY KEY
			DCS-CTR(C6,),				;COMPARE AT TARGET
			BUMP-VERIFY,				;COUNT
		NEXT,	J/EXPEC535B				;
 (6637)	DCS[1,00,1,0,0,1]  BM[1001,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,100,001,011]

	6413:	;(FREE)
	EXPEC535B:
		P3,	CSPD[14]_EMIT,				;GET EXPECTED PATTERN AFTER SHIFT
			EMIT/002645,				;"0000 0101 1010 0101"
		NEXT,	J/INITD535B				;
 (6413)	DCS[0,00,0,0,0,0]  BM[0000,,10,01,,01,10,,100,,101,,,0,0,0,,0,,0,,,0,0011,,,1,,0000,0,,,11,000,,,100,001,100]

	6414:	;(FREE)
```



```
	INITD535B:
		P2-T,	D_NOT-CSPB(B15),			;INITIAL D=(055132)
			D[C]_ALU15,				;SETUP D[C] FOR SHIFT = "0"
		NEXT,	J/COMP535B				;
 (6414)	DCS[0,00,0,0,0,0]  BM[0111,,11,10,,11,01,,101,,100,,,0,1,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,100,001,101]

	6415:	;(FREE)
	COMP535B:
		SETUP,	D-RIGHT-4,				;AMUX/DIRECT, BMUX/RIGHT-4, CMUX/DIRECT
		P2-T,	D_D-SHIFTED-XOR-CSPB(B14),		;COMPARE D-SHIFTED:EXPECTED, BITWISE
		NEXT,	J/GOBUT535B				;EXPECTED=(002645)
 (6415)	DCS[0,00,0,0,0,0]  BM[0110,,11,11,,01,01,,000,,000,,,0,1,0,,0,,0,,,0,1000,,,0,,0000,0,,,11,000,,,100,001,110]

	6416:	;(FREE)
	GOBUT535B:
		P0,	BUMP-VERIFY,				;COUNT
		SETUP,	RETURN/SCOPE535B,			;RETURN TO SCOPE LOOP TEST WORD
		NEXT,	GOTO-PAGE(7),				;BUT TABLE IS ON PAGE 7
			J/BUTD-IS-ZERO				;GO TEST D IS ALL ZERO
 (6416)	DCS[0,00,0,0,0,1]  BM[0110,,00,10,,00,01,,111,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,100,001]

	6417:	;(FREE)
	SCOPE535B:
		P3,	CSPD[16]_EMIT, EMIT/114631,		;CONSTANT FOR USE BELOW
		NEXT,	BUTD[SCOPE],				;NO ERROR:  "TEST536A" (+1,WORDS)
			J/TEST536A				;   ERROR:  "INIT535A" (-10,WORDS)
 (6417)	DCS[0,00,0,1,0,0]  BM[1001,,10,10,,01,10,,011,,001,,,0,0,0,,0,,0,,,0,0001,,,1,,0000,0,,,11,000,,,110,101,111]

	; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

	;*** TEST 536A ***
	;READ D THRU "2*D[C]!BMUX<15:02>" PORT OF CMUX, AMUX-BMUX/DIRECT
	;IN(1)(016161), OUT(143434)
	6657:
	TEST536A:
		P0,	LOAD-ENUA(ETARGET402),			;SETUP FOR DD=ZERO TEST
			LOAD-ERROR(TEST536A),			;ERROR DIRECTORY KEY
			DCS-CTR(C7,),				;COMPARE AT TARGET
		NEXT,	J/INIT536A				;
 (6657)	DCS[1,00,1,0,0,0]  BM[1000,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,110,101,100]

	6654:
	INIT536A:

		P3,	CSPD[15]_EMIT,				;GET INITIAL PATTTERN FOR D
			EMIT/016161,				;"0001 1100 0111 0001"
		NEXT,	J/EXPEC536A				;

 (6654)	DCS[0,00,0,0,0,0]  BM[0001,,10,11,,00,01,,110,,001,,,0,0,0,,0,,0,,,0,0010,,,1,,0000,0,,,11,000,,,100,010,000]

	6420:	;(FREE)
	EXPEC536A:
```

```
13987            P0,     BUMP-VERIFY,                            ;COUNT
13988            P3,     CSPD[14]_EMIT,                          ;GET EXPECTED PATTERN AFTER SHIFT
13989                    EMIT/143614,                            ;"1100 0111 0001 1100"
13990            NEXT,   J/INITD536A                             ;
    (6420)  DC8[0,00,0,0,0,1]  BM[1100,,10,01,,11,00,,011,,100,,,0,0,0,,0,,0,,,0,0011,,,1,,0000,0,,,11,000,,,100,010,001]
13991
13992    6421:  ;(FREE)
13993    INITD536A:
13994            P2-T,   D_CSPD(D15),                            ;INITIAL D=(161616)
13995                    D[C]_ALU00,                             ;SETUP D[C] FOR SHIFT = "1"
13996            NEXT,   J/COMP536A                              ;
    (6421)  DC8[0,00,0,0,0,0]  BM[1010,,10,00,,00,00,,000,,010,,,0,1,0,,0,,0,,,0,0010,,,0,,0000,0,,,11,000,,,100,010,010]
13997
13998    6422:  ;(FREE)
13999    COMP536A:
14000            SETUP,  D=RIGHT+2,                              ;ANDX=BNDX/DIRECT, CNDX/RIGHT=2
14001            P2-T,   D_D-SHIFTED-XOR-CSPB(B14),              ;COMPARE D=SHIFTED;EXPECTED, BITWISE
14002            NEXT,   J/GOBUT536A                             ;EXPECTED =(143434)
    (6422)  DC8[0,00,0,0,0,0]  BM[0110,,11,11,,01,11,,000,,000,,,0,1,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,100,010,011]
14003
14004    6423:  ;(FREE)
14005    GOBUT536A:
14006            SETUP,  RETURN/TEST536B,                        ;RETURN TO START OF NEXT SUBTEST
14007            NEXT,   GOTO-PAGE(7),                           ;BUT TABLE IS ON PAGE 7
14008                    J/BUTD-IS-ZERO                          ;GO TEST D IS ALL ZERO
    (6423)  DC8[0,00,0,0,0,0]  BM[0110,,00,11,,01,00,,000,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,100,001]
14009
14010
14011
14012
14013    ;  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -
14014
14015    ;*** TEST 536B ***
14016    ;READ D THRU "2+D[C]:BNUX<15:03>" PORT OF CNUX, ANUX=BNUX/DIRECT
14017    ;IN(0)(161616), OUT(034343)
14018    6640:
14019    TEST536B:
14020            P0,     LOAD-ENUA(ZTARGET402),                  ;SETUP FOR D=ZERO TEST
14021                    LOAD-ERROR(TEST536B),                   ;ERROR DIRECTORY KEY
14022                    DC8-CTR(C8,),                           ;COMPARE AT TARGET
14023            NEXT,   J/EXPEC536B                             ;
    (6640)  DC8[1,00,1,0,0,0]  BM[1001,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,100,010,100]
14024
14025    6424:  ;(FREE)
14026    EXPEC536B:
14027            P0,     BUMP-VERIFY,                            ;COUNT
14028            P3,     CSPD[14]_EMIT,                          ;GET EXPECTED PATTERN AFTER SHIFT
14029                    EMIT/034343,                            ;"1110 0011 1000 1110"
14030            NEXT,   J/INITD536B                             ;
    (6424)  DC8[0,00,0,0,0,1]  BM[0011,,10,10,,00,11,,100,,011,,,0,0,0,,0,,0,,,0,0011,,,1,,0000,0,,,11,000,,,100,010,101]
14031
14032    6425:  ;(FREE)
14033    INITD536B:
```



```
14034            P2-T,   D_NOT-CSPB(B15),                        ;INITIAL D=(161616)
14035                    D[C]_ALU00,                             ;SETUP D[C] FOR SHIFT = "0"
14036            NEXT,   J/COMP536B                              ;
    (6425)  DC8[0,00,0,0,0,0]  BM[0111,,11,10,,11,01,,101,,010,,,0,1,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,100,010,110]
14037
14038    6426:  ;(FREE)
14039    COMP536B:
14040            P0,     BUMP-VERIFY,                            ;COUNT
14041            SETUP,  D=RIGHT+2,                              ;ANDX=BNUX/DIRECT, CNUX/RIGHT=2
14042            P2-T,   D_D-SHIFTED-XOR-CSPB(B14),              ;COMPARE D=SHIFTED;EXPECTED, BITWISE
14043            NEXT,   J/GOBUT536B                             ;EXPECTED=(034343)
    (6426)  DC8[0,00,0,0,0,1]  BM[0110,,11,11,,01,11,,000,,000,,,0,1,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,100,010,111]
14044
14045    6427:  ;(FREE)
14046    GOBUT536B:
14047            SETUP,  RETURN/SCOPE536B,                       ;RETURN TO SCOPE LOOP TEST WORD
14048            NEXT,   GOTO-PAGE(7),                           ;BUT TABLE IS ON PAGE 7
14049                    J/BUTD-IS-ZERO                          ;GO TEST D IS ALL ZERO
    (6427)  DC8[0,00,0,0,0,0]  BM[0110,,00,10,,00,11,,000,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,100,001]
14050
14051    6430:  ;(FREE)
14052    SCOPE536B:
14053            P0,     BUMP-VERIFY,                            ;COUNT
14054            P3,     CSPD[17]_EMIT, EMIT/031463,             ;CONSTANT FOR USE BELOW
14055            NEXT,   BUTD(SCOPE),                            ;NO ERROR:  "TEST536C" (+1,WORD5)
14056                    J/TEST536C                              ;   ERROR:  "INIT536A"(-10,WORD5)
    (6430)  DC8[0,00,0,1,0,1]  BM[0011,,10,00,,11,00,,110,,011,,,0,0,0,,0,,0,,,0,0000,,,1,,0000,0,,,11,000,,,110,101,101]
14057
14058
14059
14060
14061    ;  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -
14062
14063    ;*** TEST 536C ***
14064    ;READ D THRU "D[C]=BNUX<15:01>" PORT OF CNUX, ANUX=BNUX/DIRECT
14065    ;IN(1)(031463), OUT(146631)
14066    6655:
14067    TEST536C:
14068            P0,     LOAD-ENUA(ZTARGET402),                  ;SETUP FOR D=ZERO TEST
14069                    LOAD-EPROR(TEST536C),                   ;ERROR DIRECTORY KEY
14070                    DC8-CTR(C8,),                           ;COMPARE AT TARGET
14071                    BUMP-VERIFY,                            ;COUNT
14072            NEXT,   J/INITD536C                             ;
    (6655)  DC8[1,00,1,0,0,1]  BM[1010,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,110,101,010]
14073
14074    6652:

14075    INITD536C:
14076            P0,     BUMP-VERIFY,                            ;COUNT
14077            P2-T,   D_CSPD(D17),                            ;INITIAL D=(031463)
14078                    D[C]_ALU00,                             ;SETUP D[C] FOR SHIFT = "1"
14079            NEXT,   J/COMP536C                              ;"0011 0011 0011 0011"
    (6652)  DC8[0,00,0,0,0,1]  BM[1010,,10,00,,00,00,,000,,010,,,0,1,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,100,011,001]
14080
```

```
        6431:  ;(FREE)
        COMP536C:
                SETUP,  D-RIGHT-1,                                      ;AMUX-BMUX/DIRECT, CMUX/RIGHT-1
                P2-T,   D_D-SHIFTED-XOR-CSPS(B16),                      ;COMPARE D-SHIFTED;EXPECTED, BITWISE
                NEXT,   J/GOBUT536C                                     ;EXPECTED=(114631)
 (6431)  DCS[0,00,0,0,0,0]  BM[0110,,11,01,,01,10,,000,,000,,,0,1,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,100,011,010]
                                                                        ;"1001 1001 1001 1001"

        6432:  ;(FREE)
        GOBUT536C:
                SETUP,  RETURN/TEST536D,                                ;RETURN TO START OF NEXT SUBTEST
                NEXT,   GOTO-PAGE(7),                                   ;BUT TABLE IS ON PAGE 7
                        J/BUTD-IS-ZERO                                  ;GO TEST D IS ALL ZERO
 (6432)  DCS[0,00,0,0,0,0]  BM[0110,,00,11,,01,00,,001,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,100,001]




        ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        ;*** TEST 536D ***
        ;READ 0 THRU "D[C]#BMUX<15:01>" PORT OF CMUX, AMUX-BMUX/DIRECT
        ;IN(0)(146314), OUT(063146)
        6641:
        TEST536D:
                P0,     LOAD-ENUA(ZTARGET402),                          ;SETUP FOR D=ZERO TEST
                        LOAD-ERROR(TEST536D),                           ;ERROR DIRECTORY KEY
                        DCS-CTR(C7,),                                   ;COMPARE AT TARGET
                NEXT,   J/INIT536D                                      ;
 (6641)  DCS[1,00,1,0,0,0]  BM[1000,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,100,011,011]

        6433:  ;(FREE)
        INIT536D:
                P3,     CSPD[15]_EMIT,                                  ;GET INITIAL PATTERN FOR D
                        EMIT/146314,                                    ;"1100 1100 1100 1100"
                NEXT,   J/EXPEC536D                                     ;
 (6433)  DCS[0,00,0,0,0,0]  BM[1100,,10,11,,00,11,,001,,100,,,0,0,0,,0,,0,,,0,0010,,,1,,0000,0,,,11,000,,,100,011,100]

        6434:  ;(FREE)
        EXPEC536D:
                P3,     CSPD[14]_EMIT,                                  ;GET EXPECTED PATTERN AFTER SHIFT
                        EMIT/063146,                                    ;"0110 0110 0110 0110"
                NEXT,   J/INITD536D                                     ;
 (6434)  DCS[0,00,0,0,0,0]  BM[0110,,10,01,,10,01,,100,,110,,,0,0,0,,0,,0,,,0,0011,,,1,,0000,0,,,11,000,,,100,011,101]

        6435:  ;(FREE)
        INITD536D:
                P0,     BUMP-VERIFY,                                    ;COUNT
                P2-T,   D_CSPD(D15),                                    ;INITIAL D=(146314)
                        D[C]_0,                                         ;SETUP D[C] FOR SHIFT = "0"
                NEXT,   J/COMP536D                                      ;
 (6435)  DCS[0,00,0,0,0,1]  BM[1010,,10,00,,00,00,,000,,000,,,0,1,0,,0,,0,,,0,0010,,,0,,0000,0,,,11,000,,,100,011,110]
```



```
        6436:  ;(FREE)
        COMP536D:
                SETUP,  D-RIGHT-1,                                      ;AMUX-BMUX/DIRECT, CMUX/RIGHT-1
                P2-T,   D_D-SHIFTED-XOR-CSPS(B14),                      ;COMPARE D-SHIFTED;EXPECTED, BITWISE
                NEXT,   J/GOBUT536D                                     ;EXPECTED=(063146)
 (6436)  DCS[0,00,0,0,0,0]  BM[0110,,11,11,,01,10,,000,,000,,,0,1,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,100,011,111]

        6437:  ;(FREE)
        GOBUT536D:
                SETUP,  RETURN/SCOPE536D,                               ;RETURN TO SCOPE LOOP TEST WORD
                NEXT,   GOTO-PAGE(7),                                   ;BUT TABLE IS ON PAGE 7
                        J/BUTD-IS-ZERO                                  ;GO TEST D IS ALL ZERO
 (6437)  DCS[0,00,0,0,0,0]  BM[0110,,00,10,,01,00,,000,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,100,001]

        6440:  ;(FREE)
        SCOPE536D:
                NEXT,   BUT0[SCOPE],                                    ;NO ERROR:  "TEST536E" (+1,WORDS)
                        J/TEST536E                                      ;   ERROR:  "INIT536C" (-11,WORDS)
 (6440)  DCS[0,00,0,1,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,110,101,011]




        ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        ;*** TEST 536E ***
        ;READ 0 THRU BMUX<14:00>#SENDMUX(=SR15) PORT OF CMUX, AMUX-BMUX/DIRECT
        ;IN(0)(146314), OUT(114631), SR=(100000)
        6653:
        TEST536E:
                P0,     LOAD-ENUA(ZTARGET402),                          ;SETUP FOR D=ZERO TEST
                        LOAD-ERROR(TEST536E),                           ;ERROR DIRECTORY KEY
                        DCS-CTR(C6,),                                   ;COMPARE AT TARGET
                P3,     ROTA(CLR-FLAG-RES-UCON),                        ;SELECT SR-LOAD, SENDMUX PORTS 0123, BUSDIN_EMIT
                NEXT,   J/LOADSR536E                                    ;
 (6653)  DCS[1,00,1,0,0,0]  BM[1001,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,010,,,110,101,000]

        6650:
        LOADSR536E:
                P0,     BUMP-VERIFY,                                    ;COUNT
                P2-T,   SR_BSPHI(C100000),                              ;(100000) IN SR<15:00>
                NEXT,   J/INITD536E                                     ;
 (6650)  DCS[0,00,0,0,0,1]  BM[1010,,01,11,,00,00,,001,,000,,,0,0,1,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,100,100,001]

        6441:  ;(FREE)
        INITD536E:

                P0,     BUMP-VERIFY,                                    ;COUNT
                P2-T,   D_CSPD(D15),                                    ;INITIAL D=(146314)
                        D[C]_0,                                         ;SETUP D[C] FOR SHIFT = "0"
                NEXT,   J/COMP536E                                      ;
 (6441)  DCS[0,00,0,0,0,1]  BM[1010,,10,00,,00,00,,000,,000,,,0,1,0,,0,,0,,,0,0010,,,0,,0000,0,,,11,000,,,100,100,010]

        6442:  ;(FREE)
```

```
14175    ;COMP536E:
14176            SETUP,  D=LEFT=1,                               ;AMUX=BMUX/DIRECT, CMUX/LEFT=1
14177                                                            ;BIT<00> = SR<15> = SENDMUX PORT 0 = (1)
14178            P2=T,   D_D=SHIFTED=XOR=CSPB(B14),              ;COMPARE D=SHIFTED;EXPECTED, BITWISE
14179            NEXT,   J/GOBUT536E                             ;EXPECTED=(114631)
     (6442)  DCS[0,00,0,0,0,0]  BM[0110..11.01..01.00..000..000...0.1.0..0..0...0.0000...0..0000.0...11.000...100.100.011]
14180
14181    6443:  ;(FREE)
14182    GOBUT536E:
14183            SETUP,  RETURN/TEST536F,                        ;RETURN TO START OF NEXT SUBTEST
14184            NEXT,   GOTO=PAGE(7),                           ;BUT TABLE IS ON PAGE 7
14185                    J/BUTD=IS=ZERO                          ;GO TEST D IS ALL ZERO
     (6443)  DCS[0,00,0,0,0,0]  BM[0110..00.11..01.00..010..111...0.0..0..0...0.0000...0..0000.0...11.100...011.100.001]
14186
14187
14188
14189
14190    ;  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -
14191
14192    ;*** TEST 536F ***
14193    ;READ D THRU BMUX<14:00>=SENDMUX(=SR<15) PORT OF CMUX, AMUX=BMUX/DIRECT
14194    ;IN(0)(031463), (063146), SR=(077777)
14195    6642:
14196    TEST536F:
14197            P0,     LOAD=ENUA(ETARGET402),                  ;SETUP FOR 0=ZERO TEST
14198                    LOAD=ERROR(TEST536F),                   ;ERROR DIRECTORY KEY
14199                    DCS=CTR(C6,),                           ;COMPARE AT TARGET
14200            NEXT,   J/LOADSR536F                            ;
     (6642)  DCS[1,00,1,0,0,0]  BM[1001..00.11..11.00..000..010...0.0.0..0..0...0.0000...0..0000.0...11.000...100.100.100]
14201
14202    6644:  ;(FREE)
14203    LOADSR536F:
14204            P0,     BUMP=VERIFY,                            ;COUNT
14205            P2=T,   SR_NOT=ASPHI(C100000),                  ;(077777) IN SR 15:00>
14206            NEXT,   J/INITD536F                             ;
     (6644)  DCS[0,00,0,0,0,1]  BM[0000..00.00..11.01..001..000...0.0.1..0..0...0.0000...0..0000.0...11.000...100.100.101]
14207
14208    6645:  ;(FREE)
14209    INITD536F:
14210            P2=T,   D_CSPD(D17),                            ;INITIAL D=(031463)
14211                    D[C]_0,                                 ;SETUP D[C] FOR SHIFT = "0"
14212            NEXT,   J/COMP536F                              ;
     (6645)  DCS[0,00,0,0,0,0]  BM[1010..10.00..00.00..000..000...0.1.0..0..0...0.0000...0..0000.0...11.000...100.100.110]
14213
14214    6646:  ;(FREE)
14215    COMP536F:
14216            SETUP.  D=LEFT=1,                               ;AMUX=BMUX/DIRECT, CMUX/LEFT=1
14217                                                            ;BIT<00> = SR<15> = SENDMUX PORT 0 = (0)
14218            P2=T,   D_D=SHIFTED=XOR=CSPB(B14),              ;COMPARE D=SHIFTED;EXPECTED, BITWISE
14219            NEXT,   J/GOBUT536F                             ;EXPECTED =(063146)
     (6646)  DCS[0,00,0,0,0,0]  BM[0110..11.11..01.00..000..000...0.1.0..0..0...0.0000...0..0000.0...11.000...100.100.111]
14220
14221    6647:  ;(FREE)
```



```
14222    GOBUT536F:
14223            SETUP,  RETURN/SCOPE536F,                       ;RETURN TO SCOPE LOOP TEST WORD
14224            NEXT,   GOTO=PAGE(7),                           ;BUT TABLE IS ON PAGE 7
14225                    J/BUTD=IS=ZERO                          ;GO TEST D IS ALL ZERO
     (6647)  DCS[0,00,0,0,0,0]  BM[0110..00.10..01.01..000..111...0.0.0..0..0...0.0000...0..0000.0...11.100...011.100.001]
14226
14227    6650:  ;(FREE)
14228    SCOPE536F:
14229            P0,     BUMP=VERIFY,                            ;COUNT
14230            P3,     CSPD[02]_EMIT,                          ;REG=CON #1:   (FOR USE BELOW)
14231            EMITC,  SENDMUX=0:3]=SEL,                       ;  SELECT SENDMUX PORTS 0=3
14232                    SR=LEFT, GUARD=DIS,                     ;  SR GOES LEFT, SR<00>_D[C]
14233            NEXT,   BUTD(SCOPE),                            ;NO ERROR:  "SETEMIT537A" (+1,WORDS)
14234                    J/SETEMIT537A                           ;   ERROR:  "LOADSR536E" (+0,WORDS)
     (6650)  DCS[0,00,0,1,0,1]  BM[0101..10.00..00.00..000..000...0.0.0..0..0...0.1101...1..0000.0...11.000...110.101.001]
14235
14236
14237
14238
14239    ;  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -
14240
14241    ;*** TEST 537A ***
14242    ;THIS TEST VALIDATES THE "SENDMUX" INPUTS TO THE SHIFT-TREE,
14243    ;EACH "SENDMUX" OUTPUT IS SET TO 1/0 VALUES, AND THEN READ OUT INTO THE "SR" WHERE THEY
14244    ;ARE ALL SAVED TO CHECK AT ONCE.
14245    6651:
14246    SETEMIT537A:
14247            P0,     DCS=CTR(C15,),                          ;HOLD UP FOR NOW
14248            NEXT,   GOTO=PAGE(4),                           ;XFER
14249                    J/LOAD16537A                            ;
     (6651)  DCS[0,00,1,0,0,0]  BM[0000..00.00..00.00..000..100...0.0.0..0..0...0.0000...0..0000.0...11.100...101.111.100]
14250
14251    4574:
14252    LOAD16537A:
14253            P3,     CSPD[16]_EMIT,                          ;BIT10 ONLY SET
14254                    EMIT/000010,                            ;
14255                    BUTA(CLR=FLAG=REG=UCON),                ;BUSDIN <= EMIT, SR LOAD
14256            NEXT,   J/LOAD14537A                            ;
     (4574)  DCS[0,00,0,0,0,0]  BM[0000..10.00..00.00..001..000...0.0.0..0..0...0.0001...1..0000.0...11.010...010.111.110]
14257
14258    4276:  ;(FREE)
14259    LOAD14537A:
14260
14261            P3,     CSPD(14]_EMIT,                          ;BIT11 ONLY SET
14262                    EMIT/004000,                            ;
14263            NEXT,   J/LOAD03537A                            ;

     (4276)  DCS[0,00,0,0,0,0]  BM[0000..10.10..00.00..000..000...0.0.0..0..0...0.0001...1..0000.0...11.000...010.111.111]
14264
14265    4277:  ;(FREE)
14266    LOAD03537A:
14267            P3,     CSPD[03]_EMIT,                          ;WHAT THE "SR" SHOULD BE
14268                    EMIT/177242,                            ;AFTER THIS TEST
```

```
14269            NEXT,   J/LOAD01537A                            ;
     (4277)  DCB[0,00,0,0,0,0]  BM[1111,,10,11,,10,10,,101,,010,,,0,0,0,,0,,0,,,0,1100,,,1,,0000,0,,,11,000,,,011,000,000]
14270
14271    4300:  ;(FREE)
14272    LOAD01537A:
14273            P3,     CSPD[01]_EMIT,                          ;RES-CON #2:
14274            EMITC,  SENDMUX-4567-SEL,                       ;  SELECT SENDMUX PORTS 4-7
14275                    SR-LEFT, GUARD-DIS,                     ;  SR GOES LEFT, SR<00>_D[C]
14276            NEXT,   GOTO-PAGE(7),                           ;XFER
14277                    J/INITSR537A                            ;
     (4300)  DCB[0,00,0,0,0,0]  BM[0001,,10,00,,00,00,,000,,111,,,0,0,0,,0,,0,,,0,1110,,,1,,0000,0,,,11,100,,,001,011,001]
14278
14279    7131:  ;(FREE)
14280    INITSR537A:
14281            P2=T,   SR_ALL=ONES,                            ;START 'SR' WITH ALL ONES
14282            NEXT,   J/SETRES&537A                           ;
     (7131)  DCB[0,00,0,0,0,0]  BM[1111,,00,00,,11,01,,101,,000,,,0,0,1,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,011,101]
14283
14284    7135:  ;(FREE)
14285    SETRES&537A:
14286            P2,     RES_CSPD(D02),                          ;LOAD RES W/ RES=CON#1 (SENDMUX=0123)
14287            NEXT,   GOTO=PAGE(6),                           ;XFER
14288                    J/NEWCTR537A                            ;
     (7135)  DCB[0,00,0,0,0,0]  BM[0000,,10,00,,00,00,,000,,110,,,0,0,0,,0,,0,,,0,1101,,,0,,1000,1,,,11,100,,,110,100,011]
14289
14290    6643:
14291    NEWCTR537A:
14292            P0,     LOAD=ENVA(SETRES&537A),                 ;COMPARE POINT #2
14293                    LOAD=ERROR(NEWCTR537A),                 ;ERROR DIRECTORY KEY
14294                    DCB=CTR(C14,),                          ;COMPARE BELOW
14295            NEXT,   J/AR3-537A                              ;
     (6643)  DCB[1,00,1,0,0,0]  BM[0001,,00,11,,10,11,,111,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,100,101,001]
14296
14297    6451:  ;(FREE)
14298    AR3-537A:
14299            P0,     BUMP=VERIFY,                            ;COUNT
14300            P2=T,   D_NOT-CSPD(D16),                        ;D=(177767), BIT03="0"
14301            NEXT,   GOTO=PAGE(7),                           ;XFER
14302                    J/BR3-537A                              ;
     (6451)  DCB[0,00,0,0,0,1]  BM[0111,,10,00,,11,01,,101,,111,,,0,1,0,,0,,0,,,0,0001,,,0,,0000,0,,,11,100,,,001,011,100]
14303
14304    7134:  ;(FREF)
14305    BR3-537A:
14306            P2=T,   D_D=RIGHT-3,                            ;USE SENDMUX PORT 1 = AMUX03 = "0"
14307                    D[C]_ALU00,                             ;INTO D[C] FOR SR
14308            NEXT,   J/CR3-537A                              ;
     (7134)  DCB[0,00,0,0,0,0]  BM[1111,,00,00,,01,00,,000,,010,,,0,1,0,,0,,0,,,0,1000,,,0,,0000,0,,,11,000,,,001,011,111]
14309
14310    7137:  ;(FREE)
14311    CR3-537A:
14312            P2=T,   SR_SR=LEFT-1,                           ;SENDMUX OUTPUT INTO SR<00>
```



```
14313                    D_CSPB(B16), SAVE=D[C],                 ;D=(000010), BIT03="1"
14314            NEXT,   J/DR3-537A                              ;
     (7137)  DCB[0,00,0,0,0,0]  BM[1010,,11,01,,00,00,,000,,111,,,0,1,1,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,100,000]
14315
14316    7140:  ;(FREE)
14317    DR3-537A:
14318            P2=T,   D_D=RIGHT-3,                            ;USE SENDMUX PORT 1 = AMUX03 = "1"
14319                    D[C]_ALU00,                             ;INTO D[C] FOR SR
14320            NEXT,   J/AR7-537A                              ;
     (7140)  DCB[0,00,0,0,0,0]  BM[1111,,00,00,,01,00,,000,,010,,,0,1,0,,0,,0,,,0,1000,,,0,,0000,0,,,11,000,,,001,100,001]
14321
14322    7141:  ;(FREF)
14323    AR7-537A:
14324            P2=T,   SR_SR=LEFT-1,                           ;SENDMUX OUTPUT INTO SR<00>
14325                    D_NOT-ASPHI(C000200), SAVE=D[C],        ;D=(177577), BIT07="0"
14326            NEXT,   J/BR7-537A                              ;
     (7141)  DCB[0,00,0,0,0,0]  BM[0000,,00,00,,11,01,,010,,111,,,0,1,1,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,100,010]
14327
14328    7142:  ;(FREE)
14329    BR7-537A:
14330            P2=T,   D_D=RIGHT-7,                            ;USE SENDMUX PORT 2 = D07 = "0"
14331                    D[C]_ALU00,                             ;INTO D[C] FOR SR
14332            NEXT,   J/CR7-537A                              ;
     (7142)  DCB[0,00,0,0,0,0]  BM[1111,,00,00,,01,00,,000,,010,,,0,1,0,,0,,0,,,0,0110,,,0,,0000,0,,,11,000,,,001,100,011]
14333
14334    7143:  ;(FREE)
14335    CR7-537A:
14336            P2=T,   SR_SR=LEFT-1,                           ;SENDMUX OUTPUT INTO SR<00>
14337                    D_ASPHI(C000200), SAVE=D[C],            ;D=(000200), BIT07="1"
14338            NEXT,   J/DR7-537A                              ;
     (7143)  DCB[0,00,0,0,0,0]  BM[1111,,00,00,,11,01,,010,,111,,,0,1,1,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,100,100]
14339
14340    7144:  ;(FREE)
14341    DR7-537A:
14342            P2=T,   D_D=RIGHT-7,                            ;USE SENDMUX PORT 2 = D07 = "1"
14343                    D[C]_ALU00,                             ;INTO D[C] FOR SR
14344            NEXT,   J/AR11-537A                             ;
     (7144)  DCB[0,00,0,0,0,0]  BM[1111,,00,00,,01,00,,000,,010,,,0,1,0,,0,,0,,,0,0110,,,0,,0000,0,,,11,000,,,001,100,101]
14345
14346    7145:  ;(FREE)
14347    AR11-537A:
14348            P2=T,   SR_SR=LEFT-1,                           ;SENDMUX OUTPUT INTO SR<00>
14349                    D_NOT-CSPB(B16), SAVE=D[C],             ;D=(177777), BIT11="0"
14350            NEXT,   J/BR11-537A                             ;
     (7145)  DCB[0,00,0,0,0,0]  BM[0111,,11,11,,11,01,,101,,111,,,0,1,1,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,100,110]
14351

14352    7146:  ;(FREF)
14353    BR11-537A:
14354            P2=T,   D_D=RIGHT-11,                           ;USE SENDMUX PORT 3 = AMUX03 = "0"
14355                    D[C]_ALU00,                             ;INTO D[C] FOR SR
14356            NEXT,   J/CR11-537A                             ;
     (7146)  DCB[0,00,0,0,0,0]  BM[1111,,00,00,,01,00,,000,,010,,,0,1,0,,0,,0,,,0,1110,,,0,,0000,0,,,11,000,,,001,100,111]
14357
14358    7147:  ;(FREF)
```

```
	CR11-537A:
		P2-T,	SR_SR-LEFT-1,			;SENDMUX OUTPUT INTO SR<00>
			D_CSPB(B14), SAVE-D[C],		;D=(004000), BIT11="1"
		NEXT,	J/DR11-537A			;
    (7147)  DCS[0,00,0,0,0,0]  BM[1010,,11,11,,00,00,,000,,111,,,0,1,1,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,101,000]

	7150:	;(FREE)
	DR11-537A:
		P2-T,	D_D-RIGHT-11,			;USE SENDMUX PORT 3 = AMUX03 = "1"
			D[C]_ALU00,			;INTO D[C] FOR SR
		NEXT,	J/ER3-537A			;
    (7150)  DCS[0,00,0,0,0,0]  BM[1111,,00,00,,01,00,,000,,010,,,0,1,0,,0,,0,,,0,1110,,,0,,0000,0,,,11,000,,,001,101,001]

	7151:	;(FREE)
	ER3-537A:
		P2-T,	SR_SR-LEFT-1,			;SENDMUX OUTPUT INTO SR<00>
			D_NOT-CSPB(B16), SAVE-D[C],	;D=(177767), BIT03="0"
		NEXT,	J/SETRESB537A			;
    (7151)  DCS[0,00,0,0,0,0]  BM[0111,,11,01,,11,01,,101,,111,,,0,1,1,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,011,111,000]

	7370:
	SETRESB537A:
		P2,	RES_CSPD(D01),			;LOAD RES W/RES-CON#2 (SENDMUX-4567)
		NEXT,	J/FR3-537A			;
    (7370)  DCS[0,00,0,0,0,0]  BM[0000,,10,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,1110,,,0,,1000,1,,,11,000,,,001,101,010]

	7152:	;(FREE)
	FR3-537A:
		P2-T,	D_D-RIGHT-3,			;USE SENDMUX PORT 5 = AMUX03 = "0"
			D[C]_ALU00,			;INTO D[C] FOR SR
		NEXT,	J/GR3-537A			;
    (7152)  DCS[0,00,0,0,0,0]  BM[1111,,00,00,,01,00,,000,,010,,,0,1,0,,0,,0,,,0,1000,,,0,,0000,0,,,11,000,,,001,101,011]

	7153:	;(FREE)
	GR3-537A:
		P2-T,	SR_SR-LEFT-1,			;SENDMUX OUTPUT INTO SR<00>
			D_CSPB(B16), SAVE-D[C],		;D=(000010), BIT03="1"
		NEXT,	J/HR3-537A			;
    (7153)  DCS[0,00,0,0,0,0]  BM[1010,,11,01,,00,00,,000,,111,,,0,1,1,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,101,100]

	7154:	;(FREE)
	HR3-537A:
		P2-T,	D_D-RIGHT-3,			;USE SENDMUX PORT 5 = AMUX03 = "1"
			D[C]_ALU00,			;INTO D[C] FOR SR
		NEXT,	J/AL1-537A			;
    (7154)  DCS[0,00,0,0,0,0]  BM[1111,,00,00,,01,00,,000,,010,,,0,1,0,,0,,0,,,0,1000,,,0,,0000,0,,,11,000,,,001,101,101]

	7155:	;(FREE)
	AL1-537A:
		P2-T,	SR_SR-LEFT-1,			;SENDMUX OUTPUT INTO SR<00>
			D_ALL-ONES,			;D=(177777)
		NEXT,	GOTO-PAGE(4),			;XFER
			J/BL1-537A			;
```



```
    (7155)  DCS[0,00,0,0,0,0]  BM[1111,,00,00,,11,01,,101,,100,,,0,1,1,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,100,010]

	4242:	;(FREE)
	BL1-537A:
		P2-T,	D_D-LEFT-1,			;USE SENDMUX PORT 4 = HARD "0"
			D[C]_ALU00,			;INTO D[C] FOR SR
		NEXT,	J/TEST537A			;
    (4242)  DCS[0,00,0,0,0,0]  BM[1111,,00,00,,01,00,,000,,010,,,0,1,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,101,111,010]

	4572:
	TEST537A:
		P0,	LOAD-ENUA(ZTARGET402),		;SETUP FOR D-IS-ZERO COMPARE
			LOAD-ERROR(TEST537A),		;ERROR DIRECTORY KEY
			DCS-CTR(C4,),			;COMPARE AT TARGET
			BUMP-VERIFY,			;COUNT
		P2-T,	SR_SR-LEFT-1,			;SENDMUX OUTPUT INTO SR<00>
		NEXT,	J/COMP537A			;
    (4572)  DCS[1,00,1,0,0,1]  BM[1011,,00,11,,11,00,,000,,010,,,0,0,1,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,011,000,010]

	4302:	;(FREE)
	COMP537A:
		P2-T,	D_SR-XOR-CSPD(D03),		;COMPARE RECEIVED:EXPECTED BITWISE
		P3,	PUTA(CLR-FLAG-RES-UCON),	;RESET SR TO LOAD/GUARD-DIS
		NEXT,	J/GOBUT537A			;
    (4302)  DCS[0,00,0,0,0,0]  BM[0110,,10,00,,00,00,,000,,000,,,0,1,0,,0,,0,,,0,1100,,,0,,0000,0,,,11,010,,,011,000,011]

	4303:	;(FREE)
	GOBUT537A:
		SETUP,	RETURN/SCOPE537A,		;RETURN TO SCOPE LOOP TEST WORD
		NEXT,	GOTO-PAGE(7),			;BUT TABLE IS ON PAGE 7
			J/BUTD-IS-ZERO			;GO TEST D IS ALL ZERO
    (4303)  DCS[0,00,0,0,0,0]  BM[0100,,00,01,,10,00,,100,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,100,001]

	4304:	;(FREE)
	SCOPE537A:
		P0,	BUMP-VERIFY,			;COUNT
			PUSDIN_EMIT-[I],		;RESET PROC UCON
			EN-CLK-IR(15-00),		;
		NEXT,	BUTD[SCOPE],			;NO ERROR:  "TEST551A" (+1,WORD)
			J/TEST551A			;    ERROR:  "LOAD16537A" (-30,WORDS)
    (4304)  DCS[0,00,0,1,0,1]  BM[0000,,00,00,,00,01,,000,,100,,,0,0,0,,0,,0,,,1,1001,,,0,,0000,0,,,11,000,,,101,111,101]





	;.PAGE=================================================================================================


	.TOC	*	TEST551:  BASE MACHINE DATAPATH COUNTER CAN COUNT


	;*****************************************************************************************
```

```
14452     !*
14453     !*      TEST#:  551 A = C                               UWORD#:  018 + 035
14454     !*
14455     !*      FUNCTIONS:
14456     !*
14457     !*      THE FOLLOWING THREE TESTS USE A COMMON SUBROUTINE TO TEST THE COUNTING
14458     !*      ABILITY OF THE BASE MACHINE DATAPATH COUNTER;   USING THE THREE ACTIVE
14459     !*      BUTS THAT ENABLE COUNTING.   ADMITTEDLY THIS SEEMS LIKE OVERKILL,  BUT
14460     !*      THIS METHOD WAS  THE LEAST EXPENSIVE IN  TERMS OF NUMBER OF MICROWORDS
14461     !*      USED FOR THE TESTING.
14462     !*
14463     !**************************************************************************
14464
14465
14466
14467     !THE FIRST TEST USES THE ACTIVE BUT(#13) "SR<1:0>#COUNT-IS-377" TO CHECK THE COUNTER
14468     4575:
14469     TEST551A:
14470             P0,     LOAD=ERROR(TEST551A),                   !ERROR DIRECTORY KEY
14471                     DC$=CTR(C15.),                          !HOLD UP FOR NOW
14472                     BUMP=VERIFY,                            !BUMP DC$ COUNTER
14473             NEXT,   J/LOADIN551A                            !
     (4575)  DC$(1,00,1,0,0,1)  BM(0000..00,00..00,00..000..000...0,0,0..0..0...0,0000...0..0000,0...11,000...011,000,101)
14474
14475     4305:  !(FREE)
14476     LOADIN551A:
14477             P3,     CSPD(17)_EMIT, EMIT/400,                !INCREMENT FOR D, TO MATCH COUNTER
14478             NEXT,   J/SET$R551A                             !
     (4305)  DC$(0,00,0,0,0,0)  BM(0000..10,00..01,00..000..000...0,0,0..0...0,0000...1..0000,0...11,000...011,000,110)
14479
14480     4306:  !(FREE)
14481     SET$R551A:
14482             P0,     BUMP-VERIFY,                            !COUNT
14483             P2-T,   $R_ZERO,                                !KEY IN $R<1:0>=(00) FOR SELECT THIS BUT
14484             NEXT,   J/GOTEST551A                            !
     (4306)  DC$(0,00,0,0,0,1)  BM(0011..00,00..00,00..008..000...0,0,1...0...0...0,0000...0...0000,0...11,000...101,110,100)
14485
14486     4564:
14487     GOTEST551A:
14488             SETUP,  RETURN/TEST551B,                        !GO TO TEST SUBROUTINE
14489             NEXT,   CALL(COUNT-TEST)                        !  (SEE DESCRIP, FOLLOWING)
     (4564)  DC$(0,00,0,0,0,0)  BM(0100..00,10..11,11..110..100...0,0,0...0...0...0,0000...0...0000,0...11,100...011,000,001)
14490
14491
14492
14493
14494     !  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -
14495
14496     !THE SECOND TEST USES THE ACTIVE BUT(#25) "COUNT-IS-377" TO CHECK THE COUNTER
14497     4576:
14498     TEST551B:
14499             P0,     LOAD=ENUA(ZTARGET400),                  !BIT<1:0> CLEAR
14500                     LOAD=ERROR(TEST551B),                   !ERROR DIRECTORY KEY
14501                     DC$=CTR(C4,),                           !COMPARE AT TARGET
```



```
14502             NEXT,   BUTD(SCOPE),                            !NO ERROR:  "SET$R551B" (+1, WORD$)
14503                     J/SET$R551B                             !   ERROR:  "GOTEST551A" (-1, WORD$) REPEAT PREV TEST
     (4576)  DC$(1,00,1,1,0,0)  BM(1011..00,11..11,00..500..000...0,0,0...0...0...0,0000...0...0000,0...11,000...101,110,101)
14504
14505     4565:
14506     SET$R551B:
14507             P2-T,   $R_$R=PLUS=1;                           !KEY IN $R<1:0>=(01) FOR SELECT THIS BUT
14508             NEXT,   J/GOTEST5518                            !
     (4565)  DC$(0,00,0,0,0,0)  BM(1000..01,11..00,00..000..000...0,0,1...0...0...0,0000...0...0000,0...11,000...101,110,000)
14509
14510     4560:
14511     GOTEST551B:
14512             SETUP,  RETURN/GOTEST551B1,                     !RETURN TO START OF NEXT SUATEST
14513             NEXT,   GOTO=PAGE(7);                           !BUT TABLE
14514                     J/BUTCOUNT-IS-377                       !FAKE A TEST
     (4560)  DC$(0,00,0,0,0,0)  BM(0111..00,00..11,01..111..111...0,0,0...0...0...0,0000...0...0000,0...11,100...001,011,110)
14515
14516     7136:  !(FREE)
14517     BUTCOUNT-IS-377:
14518             NEXT,   BUT(COUNT-IS-377),                      !SHOULD TARGET INTO *** *** #00
14519                     J/ZTARGET400                            !BIT<1:0> BOTH CLEAR
     (7136)  DC$(0,00,0,0,0,0)  BM(0000..00,00..00,00..000..000...0,0,0...0...0...0,0000...0...0000,0...10,101...100,000,000)
14520
14521
14522     !     -     -     -     -     -     -     -     -     -     -     -     -     -     -
14523
14524
14525     !   AND NOW TEST THAT THE ACTUAL BUT/COUNTER WORKS
14526
14527     7137:  !(FREE)
14528     GOTEST551B1:
14529             SETUP,  RETURN/TEST55(C,                        !GO TO TEST SUBROUTINE
14530             NEXT,   CALL(COUNT-TEST)                        !  (SEE DESCRIP, FOLLOWING)
     (7137)  DC$(0,00,0,0,0,0)  BM(0100..00,10..11,10..110..100...0,0,0...0...0...0,0000...0...0000,0...11,100...011,000,001)
14531
14532
14533
14534
14535     !  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -
14536
14537     !THE THIRD TEST USES THE ACTIVE BUT(#17) "COUNT-IS-377&D(C)" TO CHECK THE COUNTER
14538     4566:
14539     TEST551C:
14540             P0,     LOAD=ERROR(TEST551C),                   !ERROR DIRECTORY KEY
14541                     DC$=CTR(C15.),                          !HOLD UP FOR NOW
14542             NEXT,   BUTD(SCOPE),                            !NO ERROR:  "SET$R551C" (+1, WORD$)
14543                     J/SET$R551C                             !   ERROR:  "GOTEST5518" (-1, WORD$) REPEAT PREV TEST
     (4566)  DC$(1,00,1,1,0,0)  BM(0000..00,00..00,00..000..000...0,0,0...0...0...0,0000...0...0000,0...11,000...101,110,001)
14544
14545     4561:
14546     SET$R551C:
14547             P2-T,   $R_$R-PLUS=1;                           !KEY IN $R<1:0>=(10) FOR SELECT THIS BUT
```

```
              NEXT,   GOTO-PAGE(5),                           ;XFER
                      J/GOTEST551C                            ;
    (4561)  DCS[0,00,0,0,0,0]  BM[1001,,01,11,,00,00,,000,,101,,,0,0,1,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,110,010,110]

       5626:
       GOTEST551C:
              SETUP,  RETURN/SCOPE551,                        ;GO TO TEST SUBROUTINE
              NEXT,   CALL[COUNT-TEST]                        ;  (SEE DESCRIP, FOLLOWING)
    (5626)  DCS[0,00,0,0,0,0]  BM[0101,,00,01,,10,00,,101,,100,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,000,001]



       5305:  ;(FREE)
       SCOPE551:
              P0,     BUSDIN_EMIT-[1],                        ;RESET PROC UCON
                      EN-CLK-IR(15-00),                       ;
              NEXT,   BUTD[SCOPE],                            ;NO ERROR:  "TEST610A1"  (NEXT SECTION)
                      J/TEST610A1                             ;   ERROR:  "GOTEST551C" (-1, WORDS) REPEAT PREV TEST
    (5305)  DCS[0,00,0,1,0,0]  BM[0000,,00,00,,00,01,,000,,100,,,0,0,0,,0,,0,,,1,1001,,,0,,0000,0,,,11,000,,,110,010,111]



       ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

       ;***   B.M. COUNTER TESTING ROUTINE   ***

       ;      THIS ROUTINE USES THE B.M. COUNTER TO COUNT FROM 000-377, WAITING FOR IT
       ;      TO OVERFLOW AT THE RIGHT MOMENT.  AT THE SAME TIME THE D REGISTER HAS BEEN
       ;      TRACKING THE PROGRESS OF THE COUNTER, WAITING FOR IT TO OVERFLOW, COMPARING
       ;      COUNT FOR COUNT THE INCREMENTED VALUES, AND GUARANTEEING WE WILL EXIT THE
       ;      LOOP IF THE COUNTER SOMEHOW NEVER OVERFLOWS.


       4301:  ;(FREE)
       COUNTER01:
              P2,     COUNTER_ASPHI(C000000),                 ;LOAD COUNTER WITH (000) TO START
              NEXT,   J/COUNTER02                             ;
    (4301)  DCS[0,00,0,0,0,0]  BM[0000,,01,11,,00,00,,100,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0010,1,,,11,000,,,011,001,000]

       4310:  ;(FREE)
       COUNTER02:
              P2-T,   D_ZERO, D[C]_ALU15,                     ;ZERO D, D[C]
              P3,     A#BSPHI[17]_D,                          ;ASPHI[17] WILL TRACK THE COUNTER
              NEXT,   J/COUNTER03                             ;
    (4310)  DCS[0,00,0,0,0,0]  BM[0011,,00,11,,00,01,,011,,100,,,0,1,0,,0,,0,,,0,0000,,,0,,1011,0,,,11,000,,,101,010,110]


       ;***   THE LOOP FOR THE COUNTER TEST ENTERS HERE   ***
       4526:
       COUNTER03:
              P2-T,   D_A-XOR-B, SAVE-D[C],                   ;COMPARE COUNTER:TRACKER
                      BUS-A, COUNT#D[LO],                     ;=(COUNTER):(000)
```



```
                      BUS-B_BSPHI(R17),                       ;=(TRACKER):(000)
              NEXT,   BUTR(D[C]-B),                           ;D[C] HOLDS THE CARRYOUT FROM THE TRACKING REGISTER
                                                              ;IF (0), GOTO(COUNTER04), CONTINUE TESTING
                      J/COUNTER04                             ;IF (1), GOTO(COUNTER11), ERROR, D OVERFLOW BEFORE COUNTER
    (4526)  DCS[0,00,0,0,0,0]  BM[0110,,01,11,,01,01,,011,,111,,,0,1,0,,0,,0,,,0,0011,,,0,,0000,0,,,10,011,,,101,100,101]


       ;ENTER HERE FOR CONTINUE TEST
       4545:
       COUNTER04:
              P0,     DCS-CTR(CS,),                           ;HOLD UP ON DCS COMPARE
              NEXT,   BUTR(D14-00-EQ-0),                      ;COMPARE COUNTER:TRACKER, GENERATED ABOVE
                                                              ;IF  EQUAL, GOTO(COUNTER05), CONTINUE TEST
                      J/COUNTER12                             ;IF NOT EQUAL, GOTO(COUNTER12), COUNTER:TRACKER ERROR
    (4545)  DCS[0,00,1,0,0,0]  BM[1010,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,01,101,,,101,101,101]


       ;ENTER HERE FOR CONTINUE TEST
       4557:
       COUNTER05:
              P2-T,   D_ZERO, D[C]_ALU15,                     ;TRY TO FORCE EXIT
              NEXT,   BUTR(CASE),                             ;DETERMINE WHO TO TEST:
                                                              ;SR=0 -> COUNTER06
                                                              ;SR=1 -> COUNTER07
                      J/COUNTER06                             ;SR=2 -> COUNTER08    (SR=3 NOT USED)
    (4557)  DCS[0,00,0,0,0,0]  BM[0011,,00,00,,00,00,,000,,100,,,0,1,0,,0,,0,,,0,0000,,,0,,0000,0,,,00,000,,,101,011,100]

       ;ENTER HERE FOR SR=0
       4534:
       COUNTER06:
              P2-T,   D_A-PLUS-B, D[C]_COUT15,                ;INCREMENT TRACKING REGISTER
                      BUS-A_ASPHI(R17),                       ;
                      BUS-B_CSPD(D17),                        ;CONSTANT (400)
              P3,     A#BSPHI[17]_D,                          ;SAVE NEW
              NEXT,   BUTA(SR1-0#COUNT-IS-377),               ;IF SET, GOTO(COUNTER10), B.M. COUNTER OVERFLOWED
                      J/COUNTER03                             ;IF CLEAR, GOTO(COUNTER03), NEXT PASS THRU TEST
    (4534)  DCS[0,00,0,0,0,0]  BM[1001,,10,11,,11,01,,011,,110,,,0,1,0,,0,,0,,,0,0000,,,0,,1011,0,,,01,011,,,101,010,110]

       ;ENTER HERE FOR SR=1
       4535:
       COUNTER07:
              P2-T,   D_A-PLUS-B, D[C]_COUT15,                ;INCREMENT TRACKING REGISTER
                      BUS-A_ASPHI(R17),                       ;
                      BUS-B_CSPD(D17),                        ;CONSTANT (400)
              P3,     A#BSPHI[17]_D,                          ;SAVE NEW
              NEXT,   BUTA(COUNT-IS-377),                     ;IF SET, GOTO(COUNTER10), B.M. COUNTER OVERFLOWED

                      J/COUNTER03                             ;IF CLEAR, GOTO(COUNTER03), NEXT PASS THRU TEST
    (4535)  DCS[0,00,0,0,0,0]  BM[1001,,10,11,,11,01,,011,,110,,,0,1,0,,0,,0,,,0,0000,,,0,,1011,0,,,10,101,,,101,010,110]

       ;ENTER HERE FOR SR=2
       4536:
       COUNTER08:
              P2-T,   D_A-PLUS-B, D[C]_COUT15,                ;INCREMENT TRACKING REGISTER
                      BUS-A_ASPHI(R17),                       ;
```

```
14644                   BUS-S_CSPD(017),                        ;CONSTANT (400)
14645           P3,     ASRSPHI[17]_D,                          ;SAVE REV
14646           NEXT,   BUTA(COUNT-IS-377?D(C)),                ;IF SET, GOTO(COUNTER10), B.M, COUNTER OVERFLOWED
14647                   J/COUNTER09                             ;IF CLEAR, GOTO(COUNTER09), NEXT PASS THRU TEST
    (4536)  DCS[0,00,0,0,0,0]  BM[1001..10.11..11.01..011..110...0.1.0..0..0...0.0000...0..1011.0...01.111...101.010.101]
14648
14649
14650   ;INTERMEDIATE WORD FOR NEXT PASS
14651   4525:
14652   COUNTER09:
14653           NEXT,   J/COUNTER0]                             ;NEXT PASS
    (4525)  DCS[0,00,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0..0..0...0.0000...0..0000.0...11.000...101.010.110]
14654
14655
14656
14657   ;       *       -       -       -       -       -       -       -       -       -       -
14658
14659   ;ENTER THIS SECTION WHEN HIT AN END CONDITION:
14660
14661   ;ENTER HERE WHEN COUNTER SIGNALS IT HAS OVERFLOWN:
14662   4527:
14663   COUNTER10:
14664           P0,     DCS-CTR(C15.),                          ;HOLD UP
14665           NEXT,   BUTR(D(C)-B),                           ;NOW TEST THAT D HAS ALSO OVERFLOWED, AT THE SAME TIME
14666                                                           ;IF TRUE, GOTO(COUNTER10A),  ALL IS OK, END THE TEST
14667                   J/COUNTER10B                            ;IF FALSE, GOTO(COUNTER10B), ERROR, COUNTER OVERFLOWED TOO SOON
    (4527)  DCS[0,00,1,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0..0..0...0.0000...0..0000.0...10.011...101.011.001]
14668
14669   4533:
14670   COUNTER10A:
14671           P0,     DCS-CTR(C15.),                          ;ALL, END THE TEST WITH NO ERRORS
14672           NEXT,   BUTA(RETURN),                           ;
14673                   J/BUTERROR4                             ;
    (4533)  DCS[0,00,1,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0..0..0...0.0000...0..0000.0...11.111...011.111.110]
14674
14675   4531:
14676   COUNTER10B:
14677           P0,     DCS-CTR(C0.),                           ;FORCE ERROR NOW,
14678           NEXT,   BUTA(RETURN),                           ;  B.M, COUNTER OVERFLOWED TOO SOON
14679                   J/BUTERROR4                             ;
    (4531)  DCS[0,00,1,0,0,0]  BM[1111..00.00..00.00..000..000...0.0.0..0..0...0.0000...0..0000.0...11.111...011.111.110]
14680
14681   ;ENTER HERE WHEN WE GET A "D OVERFLOWED BEFORE COUNTER" CONDITION
14682   4547:
14683   COUNTER11:
14684           P0,     DCS-CTR(C0.),                           ;FORCE ERROR NOW,
14685           NEXT,   BUTA(RETURN),                           ;  B.M, COUNTER DIDN'T OVERFLOW ON TIME
14686                   J/BUTERROR4                             ;
    (4547)  DCS[0,00,1,0,0,0]  BM[1111..00.00..00.00..000..000...0.0.0..0..0...0.0000...0..0000.0...11.111...011.111.110]
14687
14688   ;ENTER HERE WHEN WE GET A "TRACKER ( COUNTER" MISCOMPARE
14689   4555:
14690   COUNTER12:
14691           P0,     DCS-CTR(C0.),                           ;FORCE ERROR NOW,
```



```
14692           NEXT,   BUTA(RETURN),                           ;  B.M, COUNTER DIDN'T INCREMENT IN STEP WITH TRACKER
14693                   J/BUTERROR4                             ;
    (4555)  DCS[0,00,1,0,0,0]  BM[1111..00.00..00.00..000..000...0.0.0..0..0...0.0000...0..0000.0...11.111...011.111.110]
14694
14695
14696
14697
14698
14699   1.PAGE********************************************************************************************************************
14700
14701
14702   .TOC    *   TEST610:  CONDITION CODE LOGIC
14703
14704
14705
14706   ;*****************************************************************************************
14707   ;*
14708   ;*      TESTS:  610 A - D                               UWORDS:  074 + 070
14709   ;*
14710   ;*      FUNCTIONS:
14711   ;*
14712   ;*      THE FOLLOWING EIGHT TESTS EXERCISE THE PSW CC LOGIC (THE NZVC BITS)
14713   ;*      AND THE ASSOCIATED ROMS/MUXES ETC, TO VERIFY THE ABSENCE OF STUCK
14714   ;*      ONE/ZERO CONDITIONS ON ALL LOGIC LINES.
14715   ;*
14716   ;*****************************************************************************************
14717   ;
14718   ;       SUMMARY OF CC-LOGIC TESTS:
14719   ;
14720   ;       TEST  ROM     IR     CC                                      MODIFY  PREV  GENERATED
14721   ;       NUMB  ADR     DATA   CLASS BYTE-H  D(C)/D-REG A-SIDE B-SIDE  VBIT-H NZVC  NZVC
14722   ;       ----  ---     ------ ----- ------  ---------- ------ ------  ------ ----  ----
14723   ;
14724   ;        A1   665   105200  010     1      1-100000  177700 100100     1    1010  0101
14725   ;        A2   145   105200  010     1      0-100000  077600 000200     0    1001  0110
14726   ;
14727   ;        B1   133   105300  101     1      1-000200  100000 000200     0    0100  1011
14728   ;        B2   253   005300  101     0      0-000000  100200 000200     1    1001  0110
14729   ;
14730   ;        C1   437   072000  111     0      1-000000  000200 000200     0    1010  0111
14731   ;        C2   037   072000  111     0      1-077776  037777 037777     0    1100  0001
14732   ;
14733   ;        D1   216   072000  111     0      0-177400  077600 077600     1    0101  1000
14734   ;        D2   116   072000  111     0      0-100000  000200 100200     0    0101  1000
14735   ;
14736   ;
14737
14738
14739   ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
14740
14741   ;*** TEST 610A1 ***
14742
14743   5627:
14744   TEST610A1:
```

```
14745            P0,     LOAD-ENUA(ZTARGET405),                  ;NZVC AFTER = "0101"
14746                    LOAD-ERROR(TEST610A1),                  ;ERROR DIRECTORY KEY
14747                    DCS-CTR(C15.),                          ;COMPARE AT TARGET
14748                    BUMP-VERIFY,                            ;COUNT
14749            NEXT,   J/LOADIR610A1                           !
     (5527)  DCS[1,00,1,0,0,1]  BM[0000,,00,11,,11,00,,000,,101,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,101,011,010]
14750
14751    5532:
14752    LOADIR610A1:
14753            P0,     BUMP-VERIFY,                            ;COUNT
14754            P2=0,   IR_EMIT, EMIT/105200,                   ;(105200)=INCB, CC-CLASS="010", BYTE=H
14755            NEXT,   J/LOAD01-610A1                          !
     (5532)  DCS[0,00,0,0,0,1]  BM[1000,,00,10,,10,10,,000,,000,,,0,0,0,,0,,0,,,1,1010,,,0,,0000,0,,,11,000,,,011,000,110]
14756
14757    5106:  1(FREE)
14758    LOAD01-610A1:
14759            P0,     BUMP-VERIFY,                            ;COUNT
14760            P3,     CSPD[01]_EMIT, EMIT/100012,             ;FOR D[C]=(1), PS[NZVC]="1010" PREVIOUSLY
14761            NEXT,   J/LOAD05-610A1                          !
     (5306)  DCS[0,00,0,0,0,1]  BM[1000,,10,00,,00,00,,001,,010,,,0,0,0,,0,,0,,,0,1110,,,1,,0000,0,,,11,000,,,011,000,111]
14762
14763    5107:  1(FREE)
14764    LOAD05-610A1:
14765            P3,     CSPD[05]_EMIT, EMIT/177700,             ;A-SIDE DATA
14766            NEXT,   J/LOAD06-610A1                          !
     (5307)  DCS[0,00,0,0,0,0]  BM[1111,,10,11,,11,11,,000,,000,,,0,0,0,,0,,0,,,0,1010,,,1,,0000,0,,,11,000,,,011,001,000]
14767
14768    5110:  1(FREE)
14769    LOAD06-610A1:
14770            P3,     CSPD[06]_EMIT, EMIT/100100,             ;B-SIDE DATA
14771            NEXT,   J/PSCC-DC610A1                          !
     (5310)  DCS[0,00,0,0,0,0]  BM[1000,,10,00,,00,01,,000,,000,,,0,0,0,,0,,0,,,0,1001,,,1,,0000,0,,,11,000,,,011,001,001]
14772
14773    5111:  1(FREE)
14774    PSCC-DC610A1:
14775            SETUP,  RETURN/SETBUSA610A1,                    ;EXEC SUBR WHICH:
14776                                                            ;(1) CSP(10)<3:0> -> PS[NZVC]
14777            NEXT,   CALL(SETUPPSCC#DC)                      ;(2) CSP(10)<15> -> D[C]
     (5311)  DCS[0,00,0,0,0,0]  BM[0101,,00,01,,10,01,,010,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,001,101,110]
14778
14779    5112:  1(FREE)
14780    SETBUSA610A1:
14781            P2=T,   SR_CSPD(D05),                           ;GET CONSTANT FOR A-SIDE WHEN CC SET
14782            NEXT,   J/DOIT610A1                             !
     (5312)  DCS[0,00,0,0,0,0]  BM[1010,,10,00,,00,00,,000,,000,,,0,0,1,,0,,0,,,0,1010,,,0,,0000,0,,,11,000,,,011,001,011]
14783
14784    5113:  1(FREE)
14785    DOIT610A1:
14786            SETUP,  SET-CC,                                 ;FOR CLOCKING CC-S IN NEXT UWORD
14787                    MODIFY-YBIT,                            ;EXTRA CC ROM INPUT
14788            P2=T,   D_A-PLUS-B, SAVE=D[C],                  ;D=(100000), D[C]=(1)
14789                    BUS=A_SR,                               ;A=(177700)
14790                    BUS=B_CSPD(D06),                        ;B=(100100)
```



```
14791            NEXT,   J/GETIT610A1                            !
     (5313)  DCS[0,00,0,0,0,0]  BM[1001,,10,00,,00,10,,000,,111,,,0,1,0,,0,,1,,,0,1001,,,0,,0000,0,,,11,000,,,011,001,100]
14792
14793    5114:  1(FREE)
14794    GETIT610A1:
14795            P2=T,   CLK-CC,                                 ;PS[NZVC] GENERATED ABOVE LATCHED HERE
14796            SETUP,  RETURN/TEST610A2,                       ;EXEC SUBR WHICH:
14797            NEXT,   CALL(PSCCTOSR3=0)                       ;  PS<3:0> -> SR<3:0>, J/BUT(SR3=0)
     (5314)  DCS[0,00,0,0,0,0]  BM[0101,,00,11,,00,10,,010,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,001,110,010]
14798
14799
14800
14801    ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
14802
14803    ;*** TEST 610A2 ***
14804
14805    5622:
14806    TEST610A2:
14807            P0,     LOAD-ENUA(ZTARGET406),                  ;NZVC AFTER = "0110"
14808                    LOAD-ERROR(TEST610A2),                  ;ERROR DIRECTORY KEY
14809                    DCS-CTR(C13.),                          ;COMPARE AT TARGET
14810            NEXT,   J/LOAD06-610A2                          !
     (5627)  DCS[1,00,1,0,0,0]  BM[0010,,00,11,,11,00,,000,,110,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,011,001,101]
14811
14812    5115:  1(FREE)
14813    LOAD06-610A2:
14814            P0,     BUMP-VERIFY,                            ;COUNT
14815            P3,     CSPD[06]_EMIT, EMIT/077600,             ;A-SIDE DATA
14816            NEXT,   J/LOAD01-610A2                          !
     (5315)  DCS[0,00,0,0,0,1]  BM[0111,,10,11,,11,10,,000,,000,,,0,0,0,,0,,0,,,0,1001,,,1,,0000,0,,,11,000,,,011,001,110]
14817
14818    5116:  1(FREE)
14819    LOAD01-610A2:
14820            P3,     CSPD[01]_EMIT, EMIT/000011,             ;FOR D[C]=(0), PS[NZVC]="1001" PREVIOUSLY
14821            NEXT,   J/PSCC-DC610A2                          !
     (5316)  DCS[0,00,0,0,0,0]  BM[0000,,10,00,,00,00,,001,,001,,,0,0,0,,0,,0,,,0,1110,,,1,,0000,0,,,11,000,,,011,001,111]
14822
14823    5117:  1(FREE)
14824    PSCC-DC610A2:
14825            SETUP,  RETURN/SETBUSA610A2,                    ;EXEC SUBR WHICH:
14826                                                            ;(1) CSP(10)<3:0> -> PS[NZVC]
14827            NEXT,   CALL(SETUPPSCC#DC)                      ;(2) CSP(10)<15> -> D[C]
     (5317)  DCS[0,00,0,0,0,0]  BM[0101,,00,01,,10,10,,000,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,001,101,110]
14828
14829    5320:  1(FREE)
14830    SETBUSA610A2:

14831            P2=T,   SR_CSPD(D06),                           ;GET CONSTANT FOR A-SIDE WHEN CC SET
14832            NEXT,   J/DOIT610A2                             !
     (5320)  DCS[0,00,0,0,0,0]  BM[1010,,10,00,,00,00,,000,,000,,,0,0,1,,0,,0,,,0,1001,,,0,,0000,0,,,11,000,,,011,010,001]
14833
14834    5321:  1(FREE)
14835    DOIT610A2:
```

```
         SETUP,  SET-CC,                                 !FOR CLOCKING CC-S IN NEXT UWORD
                 NOT-MODIFY-VBIT,                        !EXTRA CC ROM INPUT
         P2-T,   D_A-PLUS-B, SAVE-D[C],                  !D=(100000), D[C]=(0)
                 BUS-A_SR,                               !A=(077600)
                 BUS-B_BSPHI(C000200),                   !B=(000200)
         NEXT,   J/GETIT610A2                            !
 (5321)  DCS[0,00,0,0,0,0]  BM[1001,,01,11,,00,00,,010,,111,,,0,1,0,,0,,1,,,0,0000,,,0,,0000,0,,,11,000,,,011,010,010]

    5322:  !(FREE)
    GETIT610A2:
         P2-T,   CLK-CC,                                 !PS[NZVC] GENERATED ABOVE LATCHED HERE
         SETUP,  RETURN/SCOPE610A,                       !EXEC SUBR WHICH:
         NEXT,   CALL[PSCCTOSR3-0]                       !  PS<3:0> => SR<3:0>, J/BUT(SR3-0)
 (5322)  DCS[0,00,0,0,0,0]  BM[0101,,00,01,,10,10,,011,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,001,110,010]



    5323:  !(FREE)
    SCOPE610A:
         P0,     BUMP-VERIFY,                            !COUNT
         NEXT,   BUTO[SCOPE],                            !NO ERROR:  "TEST610B1" (+1, WORDS)
                 J/TEST610B1                             !    ERROR:  "LOADIR610A1" (-18, WORDS)
 (5323)  DCS[0,00,0,1,0,1]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,101,011,011]








    ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

    !*** TEST 610B1 ***

    5533:
    TEST610B1:
         P0,     LOAD-ENUA[ZTARGET413],                  !NZVC AFTER = "1011"
                 LOAD-ERROR(TEST610B1),                  !ERROR DIRECTORY KEY
                 DCS-CTR(C13,),                          !COMPARE AT TARGET
         NEXT,   J/LOADIR610B1                           !
 (5533)  DCS[1,00,1,0,0,0]  BM[0010,,00,11,,11,00,,001,,011,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,101,011,100]

    5534:
    LOADIR610B1:
         P0,     BUMP-VERIFY,                            !COUNT
         P2-U,   IR_EMIT, EMIT/105300,                   !(105300)=DECB, CC-CLASS="101", BYTE-M
         NEXT,   J/LOAD01-610B1                          !
 (5534)  DCS[0,00,0,0,0,1]  BM[1000,,00,10,,10,11,,000,,000,,,0,0,0,,0,,0,,,1,1010,,,0,,0000,0,,,11,000,,,011,010,100]

    5324:  !(FREE)
    LOAD01-610B1:
         P3,     CSPR[01]_EMIT, EMIT/100004,             !FOR D[C]=(1), PS[NZVC]="0100" PREVIOUSLY
```



```
         NEXT,   J/PSCC-DC610B1                          !
 (5324)  DCS[0,00,0,0,0,0]  BM[1000,,10,00,,00,00,,000,,100,,,0,0,0,,0,,0,,,0,1110,,,1,,0000,0,,,11,000,,,011,010,101]

    5325:  !(FREE)
    PSCC-DC610B1:
         SETUP,  RETURN/SETBUSA610B1,                    !EXEC SUBR WHICH:
                                                         !(1) CSP(10)<3:0> => PS(NZVC)
         NEXT,   CALL[SETUPPSCC&DC]                      !(2) CSP(10)<15> => D(C)
 (5325)  DCS(0,00,0,0,0,0]  BM[0101,,00,01,,10,10,,110,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,001,101,110]

    5326:  !(FREE)
    SETBUSA610B1:
         P2-T,   SR_BSPHI(C100000),                      !GET CONSTANT FOR A-SIDE WHEN CC SET
         NEXT,   J/DOIT610B1                             !
 (5326)  DCS[0,00,0,0,0,0]  BM[1010,,01,11,,00,00,,001,,000,,,0,0,1,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,011,010,111]

    5327:  !(FREE)
    DOIT610B1:
         SETUP,  SET-CC,                                 !FOR CLOCKING CC-S IN NEXT UWORD
                 NOT-MODIFY-VBIT,                        !EXTRA CC ROM INPUT
         P2-T,   D_NOT-A-AND-B, SAVE-D[C],               !D=(000200), D[C]=(1)
                 BUS-A_SR,                               !A=(100000)
                 BUS-B_BSPHI(C000200),                   !B=(000200)
         NEXT,   J/GETIT610B1                            !
 (5327)  DCS[0,00,0,0,0,0]  BM[0010,,01,11,,00,00,,010,,111,,,0,1,0,,0,,1,,,0,0000,,,0,,0000,0,,,11,000,,,011,011,000]

    5330:  !(FREE)
    GETIT610B1:
         P2-T,   CLK-CC,                                 !PS[NZVC] GENERATED ABOVE LATCHED HERE
         SETUP,  RETURN/TEST610B2,                       !EXEC SUBR WHICH:
         NEXT,   CALL[PSCCTOSR3-0]                       !  PS<3:0> => SR<3:0>, J/BUT(SR3-0)
 (5330)  DCS[0,00,0,0,0,0]  BM[0101,,00,11,,00,01,,010,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,001,110,010]



    ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

    !*** TEST 610B2 ***

    5612:
    TEST610B2:
         P0,     LOAD-ENUA(ZTARGET406),                  !NZVC AFTER = "0110"
                 LOAD-ERROR(TEST610B2),                  !ERROR DIRECTORY KEY
                 DCS-CTR(C14,),                          !COMPARE AT TARGET
         NEXT,   J/LOADIR610B2                           !
 (5612)  DCS[1,00,1,0,0,0]  BM[0001,,00,11,,11,00,,000,,110,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,011,011,001]


    5331:  !(FREE)
    LOADIR610B2:

         P0,     BUMP-VERIFY,                            !COUNT
         P2-U,   IR_EMIT, EMIT/005300,                   !(005300)=DEC, CC-CLASS="101", NOT-BYTE-M
         NEXT,   J/LOAD07-610B2                          !
```

```
      (5331)  DCS[0,00,0,0,0,1]  BM[0000,,00,10,,10,11,,000,,000,,,0,0,0,,0,,0,,,1,1010,,,0,,0000,0,,,11,000,,,011,011,010]

        5332:  !(FREE)
        LOAD07-610B2:
                P3,     CSPD[07]_EMIT, EMIT/100200,                   !A-SIDE DATA
                NEXT,   J/LOAD01-610B2                                !
      (5332)  DCS[0,00,0,0,0,0]  BM[1000,,10,00,,00,10,,000,,000,,,0,0,0,,0,,0,,,0,1000,,,1,,0000,0,,,11,000,,,011,011,011]

        5333:  !(FREE)
        LOAD01-610B2:
                P3,     CSPD[01]_EMIT, EMIT/000011,                   !FOR D[C]=(0), PS[NZVC]="1001" PREVIOUSLY
                NEXT,   J/PSCC-DC610B2                                !
      (5333)  DCS[0,00,0,0,0,0]  BM[0000,,10,00,,00,00,,001,,001,,,0,0,0,,0,,0,,,0,1110,,,1,,0000,0,,,11,000,,,011,011,100]

        5334:  !(FREE)
        PSCC-DC610B2:
                SETUP,  RETURN/SETBUSA610B2,                          !EXEC SUBR WHICH:
                                                                      !(1) CSP(10)<3:0> => PS[NZVC]
                NEXT,   CALL[SETUPPSCC#DC]                            !(2) CSP(10)<15> => D[C]
      (5334)  DCS[0,00,0,0,0,0]  BM[0101,,00,01,,10,11,,101,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,001,101,110]

        5335:  !(FREE)
        SETBUSA610B2:
                P2-T,   SP_CSPD(D07),                                 !GET CONSTANT FOR A-SIDE WHEN CC SET
                NEXT,   J/DOIT610B2                                   !
      (5335)  DCS[0,00,0,0,0,0]  BM[1010,,10,00,,00,00,,000,,000,,,0,0,1,,0,,0,,,0,1000,,,0,,0000,0,,,11,000,,,011,011,110]

        5336:  !(FREE)
        DOIT610B2:
                SETUP,  SET-CC,                                       !FOR CLOCKING CC-S IN NEXT UWORD
                        MODIFY-VBIT,                                  !EXTRA CC ROM INPUT
                P2-T,   D_NOT-A-AND-B, SAVE-D[C],                     !D=(000000), D[C]=(0)
                        BUS-A_SP,                                     !A=(100200)
                        BUS-B_BSPHI(C000200),                         !B=(000200)
                NEXT,   J/GETIT610B2                                  !
      (5336)  DCS[0,00,0,0,0,0]  BM[0010,,01,11,,00,10,,010,,111,,,0,1,0,,0,,1,,,0,0000,,,0,,0000,0,,,11,000,,,011,011,111]

        5337:  !(FREE)
        GETIT610B2:
                P2-T,   CLK-CC,                                       !PS[NZVC] GENERATED ABOVE LATCHED HERE
                SETUP,  RETURN/SCOPE610B,                             !EXEC SUBR WHICH:
                NEXT,   CALL[PSCCTOSR3-0]                             !   PS<3:0> => SR<3:0>, J/BUT(SR3-0)
      (5337)  DCS[0,00,0,0,0,0]  BM[0101,,00,01,,11,00,,000,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,001,110,010]



        5340:  !(FREE)
        SCOPE610B:
                P0,     BUMP-VERIFY,                                  !COUNT
                NEXT,   BUTD(SCOPE),                                  !NO ERROR:  "TEST610C1" (+1, WORDS)
                        J/TEST610C1                                   !   ERROR:  "LOADIR610B1" (-14, WORDS)
      (5340)  DCS[0,00,0,1,0,1]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,101,011,101]
```



```









        ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        !*** TEST 610C1 ***

        5535:
        TEST610C1:
                P0,     LOAD-ENUA(ZTARGET407),                        !NZVC AFTER = "0111"
                        LOAD-ERROR(TEST610C1),                        !ERROR DIRECTORY KEY
                        DCS-CTP(C13,),                                !COMPARE AT TARGET
                NEXT,   J/LOADIR610C1                                 !
      (5535)  DCS[1,00,1,0,0,0]  BM[0010,,00,11,,11,00,,000,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,101,100,100]

        5544:
        LOADIR610C1:
                P0,     BUMP-VERIFY,                                  !COUNT
                P2-U,   IR_EMIT, EMIT/072000,                         !(072000)=ASH, CC-CLASS="111", NOT-BYTE-K
                NEXT,   J/LOAD01-610C1                                !
      (5544)  DCS[0,00,0,0,0,1]  BM[0111,,00,01,,00,00,,000,,000,,,0,0,0,,0,,0,,,1,1010,,,0,,0000,0,,,11,000,,,011,100,001]

        5341:  !(FREE)
        LOAD01-610C1:
                P3,     CSPD[01]_EMIT, EMIT/100012,                   !FOR D[C]=(1), PS[NZVC]="1010" PREVIOUSLY
                NEXT,   J/PSCC-DC610C1                                !
      (5341)  DCS[0,00,0,0,0,0]  BM[1000,,10,00,,00,00,,001,,010,,,0,0,0,,0,,0,,,0,1110,,,1,,0000,0,,,11,000,,,011,100,010]

        5342:  !(FREE)
        PSCC-DC610C1:
                SETUP,  RETURN/SETBUSA610C1,                          !EXEC SUBR WHICH:
                                                                      !(1) CSP(10)<3:0> => PS[NZVC]
                NEXT,   CALL[SETUPPSCC#DC]                            !(2) CSP(10)<15> => D[C]
      (5342)  DCS[0,00,0,0,0,0]  BM[0101,,00,01,,11,00,,011,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,001,101,110]

        5343:  !(FREE)
        SETBUSA610C1:
                P2-T,   SP_BSPHI(C000200),                            !GET CONSTANT FOR A-SIDE WHEN CC SET
                NEXT,   J/DOIT610C1                                   !
      (5343)  DCS[0,00,0,0,0,0]  BM[1010,,01,11,,00,00,,010,,000,,,0,0,1,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,011,100,100]

        5344:  !(FREE)
        DOIT610C1:
                SETUP,  SET-CC,                                       !FOR CLOCKING CC-S IN NEXT UWORD
                        NOT-MODIFY-VBIT,                              !EXTRA CC ROM INPUT
                P2-T,   D_A-XOR-B, SAVE-D[C],                         !D=(000000), D[C]=(1)
                        BUS-A_SP,                                     !A=(000200)
                        BUS-B_BSPHI(C000200),                         !B=(000200)
```

```
            NEXT,   J/GETIT610C1                                    ;
   (5344)  DCS[0,00,0,0,0,0]  BM[0110,,01,11,,00,00,,010,,111,,,0,1,0,,0,,1,,,0.0000,,,0,,0000,0,,,11,000,,,011,100,101]

    5345:  ;(FREE)
    GETIT610C1:
            P2-T,   CLK-CC,                                         ;PS[NZVC] GENERATED ABOVE LATCHED HERE
            SETUP,  RETURN/TEST610C2,                               ;EXEC SUBR WHICH:
            NEXT,   CALL[PSCCTOSR3-0]                               ;  PS<3:0> -> SR<3:0>, J/BUT(SR3=0)
   (5345)  DCS[0,00,0,0,0,0]  BM[0101,,00,11,,00,00,,010,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,001,110,010]




    ;  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -

    ;*** TEST 610C2 ***

    5602:
    TEST610C2:
            P0,     LOAD-ENUA(ZTARGET401),                          ;NZVC AFTER = "0001"
                    LOAD-ERROR(TEST610C2),                          ;ERROR DIRECTORY KEY
                    DCS-CTR(C13,),                                  ;COMPARE AT TARGET
            NEXT,   J/LOAD05-610C2                                  ;
   (5602)  DCS[1,00,1,0,0,0]  BM[0010,,00,11,,11,00,,000,,001,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,011,100,110]

    5346:  ;(FREE)
    LOAD05-610C2:
            P0,     BUMP-VERIFY,                                    ;COUNT
            P3,     CSPD[05]_EMIT, EMIT/037777,                     ;A AND B SIDE DATA
            NEXT,   J/LOAD01-610C2                                  ;
   (5346)  DCS[0,00,0,0,0,1]  BM[0011,,10,11,,11,11,,111,,111,,,0,0,0,,0,,0,,,0,1010,,,1,,0000,0,,,11,000,,,011,100,111]

    5347:  ;(FREE)
    LOAD01-610C2:
            P3,     CSPD[01]_EMIT, EMIT/100014,                     ;FOR D[C]=(1), PS[NZVC]="1100" PREVIOUSLY
            NEXT,   J/PSCC-DC610C2                                  ;
   (5347)  DCS[0,00,0,0,0,0]  BM[1000,,10,00,,00,00,,001,,100,,,0,0,0,,0,,0,,,0,1110,,,1,,0000,0,,,11,000,,,011,101,000]

    5350:  ;(FREE)
    PSCC-DC610C2:
            SETUP,  RETURN/SETBUSA610C2,                            ;EXEC SUBR WHICH:
                                                                    ;(1) CSP[10]<3:0> -> PS[NZVC]
            NEXT,   CALL[SETUPPSCC&DC]                              ;(2) CSP[10]<15> -> D[C]
   (5350)  DCS[0,00,0,0,0,0]  BM[0101,,00,01,,11,01,,001,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,001,101,110]

    5351:  ;(FREE)
    SETBUSA610C2:
            P2-T,   SR_CSPD(D05),                                   ;GET CONSTANT FOR A-SIDE WHEN CC SET
            NEXT,   J/DOIT610C2                                     ;
   (5351)  DCS[0,00,0,0,0,0]  BM[1010,,10,00,,00,00,,000,,000,,,0,0,1,,0,,0,,,0,1010,,,0,,0000,0,,,11,000,,,011,101,010]

    5352:  ;(FREE)
```



```
    DOIT610C2:
            SETUP,  SET-CC,                                         ;FOR CLOCKING CC-S IN NEXT UWORD
                    NOT-MODIFY-VBIT,                                ;EXTRA CO ROM INPUT
            P2-T,   D_A-PLUS-B, SAVE-D(C),                          ;D=(077776), D[C]=(1)
                    BUS-A_SR,                                       ;A=(037777)
                    BUS-B_CSPD(D05),                                ;B=(037777)
            NEXT,   J/GETIT610C2                                    ;
   (5352)  DCS[0,00,0,0,0,0]  BM[1001,,10,00,,00,00,,000,,111,,,0,1,0,,0,,1,,,0,1010,,,0,,0000,0,,,11,000,,,011,101,011]

    5353:  ;(FREE)
    GETIT610C2:
            P2-T,   CLK-CC,                                         ;PS[NZVC] GENERATED ABOVE LATCHED HERE
            SETUP,  RETURN/SCOPE610C,                               ;EXEC SUBR WHICH:
            NEXT,   CALL[PSCCTOSR3-0]                               ;  PS<3:0> -> SR<3:0>, J/BUT(SR3=0)
   (5353)  DCS[0,00,0,0,0,0]  BM[0101,,00,01,,11,01,,100,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,001,110,010]



    5354:  ;(FREE)
    SCOPE610C:
            P0,     BUMP-VERIFY,                                    ;COUNT
            NEXT,   BUTD[SCOPE],                                    ;NO ERROR:  "TEST610D1" (+1, WORDS)
                    J/TEST610D1                                     ;   ERROR:  "LOADIN610C1" (-13, WORDS)
   (5354)  DCS[0,00,0,1,0,1]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,101,100,101]









    ;  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -

    ;*** TEST 610D1 ***

    5545:
    TEST610D1:
            P0,     LOAD-ENUA(ZTARGET410),                          ;NZVC AFTER = "1000"
                    LOAD-ERROR(TEST610D1),                          ;ERROR DIRECTORY KEY
                    DCS-CTR(C12,),                                  ;COMPARE AT TARGET
            NEXT,   J/LOAD01-610D1                                  ;
   (5545)  DCS[1,00,1,0,0,0]  BM[0011,,00,11,,11,00,,001,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,101,001,000]

    5510:
    LOAD01-610D1:

            P3,     CSPD(01)_EMIT, EMIT/000005,                     ;FOR D[C]=(0), PS[NZVC]="0101" PREVIOUSLY
            NEXT,   J/PSCC-DC610D1                                  ;
   (5510)  DCS[0,00,0,0,0,0]  BM[0000,,10,00,,00,00,,000,,101,,,0,0,0,,0,,0,,,0,1110,,,1,,0000,0,,,11,000,,,011,101,101]

    5355:  ;(FREE)
    PSCC-DC610D1:
            SETUP,  RETURN/SETBUSA610D1,                            ;EXEC SUBR WHICH:
```

```
                                                            ;(1) CSP(10)<3:0> => PS[NZVC]
            NEXT,   CALL[SETUPPSCC#DC]                      ;(2) CSP(10)<15> => D[C]
    (5355)  DCS[0,00,0,0,0,0]  BM[0101,,00,01,,11,01,,110,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,001,101,110]

    5356:   ;(FREE)
    SETBUSA610D1:
            P2-T,   SR_CSPD(D06),                           ;GET CONSTANT FOR A-SIDE WHEN CC SET
            NEXT,   J/DOIT610D1                             ;
    (5356)  DCS[0,00,0,0,0,0]  BM[1010,,10,00,,00,00,,000,,000,,,0,0,1,,0,,0,,,0,1001,,,0,,0000,0,,,11,000,,,011,101,111]

    5357:   ;(FREE)
    DOIT610D1:
            SETUP,  SET-CC,                                 ;FOR CLOCKING CC-S IN NEXT UWORD
                    MODIFY-VBIT,                            ;EXTRA CC ROM INPUT
            P2-T,   D_A-PLUS-B, SAVE-D[C],                  ;D=(177400), D[C]=(0)
                    BUS-A_SR,                               ;A=(077400)
                    BUS-B_CSPD(D06),                        ;B=(077400)
            NEXT,   J/GETIT610D1                            ;
    (5357)  DCS[0,00,0,0,0,0]  BM[1001,,10,00,,00,10,,000,,111,,,0,1,0,,0,,1,,,0,1001,,,0,,0000,0,,,11,000,,,011,110,000]

    5360:   ;(FREE)
    GETIT610D1:
            P2-T,   CLK-CC,                                 ;PS[NZVC] GENERATED ABOVE LATCHED HERE
            SETUP,  RETURN/TEST610D2,                       ;EXEC SUBR WHICH:
            NEXT,   CALL[PSCCTOSR3-0]                       ;  PS<3:0> => SR<3:0>, J/BUT(SR3-0)
    (5360)  DCS[0,00,0,0,0,0]  BM[0101,,00,11,,00,10,,100,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,001,110,010]



    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

    ;*** TEST 610D2 ***

    5624:
    TEST610D2:
            P0,     LOAD-ENUA(ZTARGET410),                  ;NZVC AFTER = "1000"
                    LOAD-ERROR(TEST610D2),                  ;ERROR DIRECTORY KEY
                    DCS-CTR(C11,),                          ;COMPARE AT TARGET
            NEXT,   J/PSCC-DC610D2                          ;
    (5624)  DCS[1,00,1,0,0,0]  BM[0100,,00,11,,11,00,,001,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,011,110,001]

    5361:   ;(FREE)
    PSCC-DC610D2:
            SETUP,  RETURN/SETBUSA610D2,                    ;EXEC SUBR WHICH:
                                                            ;(1) CSP(10)<3:0> => PS[NZVC]
            NEXT,   CALL(SETUPPSCC#DC]                      ;(2) CSP(10)<15> => D[C]
    (5361)  DCS[0,00,0,0,0,0]  BM[0101,,00,01,,11,10,,010,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,001,101,110]

    5362:   ;(FREE)
    SETBUSA610D2:
            P2-T,   SR_BSPHI(C000200),                      ;GET CONSTANT FOR A-SIDE WHEN CC SET
```



```
            NEXT,   J/DOIT610D2                             ;
    (5362)  DCS[0,00,0,0,0,0]  BM[1010,,01,11,,00,00,,010,,000,,,0,0,1,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,011,110,011]

    5363:   ;(FREE)
    DOIT610D2:
            SETUP,  SET-CC,                                 ;FOR CLOCKING CC-S IN NEXT UWORD
                    NOT-MODIFY-VBIT,                        ;EXTRA CC ROM INPUT
            P2-T,   D_NOT-A-AND-B, SAVE-D(C),               ;D=(100000), D[C]=(0)
                    BUS-A_SR,                               ;A=(000200)
                    BUS-B_CSPD(D07],                        ;B=(100200)
            NEXT,   J/GETIT610D2                            ;
    (5363)  DCS[0,00,0,0,0,0]  BM[0010,,10,00,,00,00,,000,,111,,,0,1,0,,0,,1,,,0,1000,,,0,,0000,0,,,11,000,,,011,110,100]

    5364:   ;(FREE)
    GETIT610D2:
            P2-T,   CLK-CC,                                 ;PS[NZVC] GENERATED ABOVE LATCHED HERE
            SETUP,  RETURN/SCOPE610D,                       ;EXEC SUBR WHICH:
            NEXT,   CALL[PSCCTOSR3-0]                       ;  PS<3:0> => SR<3:0>, J/BUT(SR3-0)
    (5364)  DCS[0,00,0,0,0,0]  BM[0101,,00,01,,11,10,,101,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,001,110,010]



    5365:   ;(FREE)
    SCOPE610D:
            P0,     BUSDIN_EXIT-[I],                        ;RESET PROC UCONS
                    EN-CLK-IR[15-00],                       ;
            NEXT,   RUID[SCOPE],                            ;NO ERROR:  "TEST620A" (+6, WORDS)
                    J/TEST620A                              ;   ERROR:  "LOAD01-610D1" (-10, WORDS)
    (5365)  DCS[0,00,0,1,0,0]  BM[0000,,00,00,,00,01,,000,,100,,,0,0,0,,0,,0,,,1,1001,,,0,,0000,0,,,11,000,,,101,001,001]







    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

    ;       THESE TWO SUBROUTINES ARE USED IN THE ABOVE CC-LOGIC TESTS:

    7156:   ;(FREE)
    SETUPPSCC#DC:
            P2-T,   D_CSPD(D01), D[C]_ALU15,                ;SET D(C) FROM BIT<15>
            NEXT,   J/SETUPPSCC#DC02                        ;
    (7156)  DCS[0,00,0,0,0,0]  BM[1010,,10,00,,00,00,,000,,100,,,0,1,0,,0,,0,,,0,1110,,,0,,0000,0,,,11,000,,,001,110,001]

    7161:   ;(FREE)
    SETUPPSCC#DC02:
            P2,     PS(3-0)_D(3-0)-[I],                     ;SET PS[CC] FROM BIT<03:00>
                    BUSDIN_PS-[I],                          ;SETUP BUSDIN TO READ PS
            NEXT,   BUTA(RETURN),                           ;AND RETURN
                    J/BUTERROR7                             ;ERROR IF HERE
    (7161)  DCS[0,00,0,0,0,0]  BM[1000,,00,00,,10,01,,000,,000,,,0,0,0,,0,,0,,,1,1011,,,0,,0000,0,,,11,111,,,011,111,110]
```

```
15207
15208
15209
15210
15211    7162:  ;(FREE)
15212    PSCCTOSR3=0:
15213            P3,     CSPD[03]_BUSDIN,                        ;GET ENABLED P6 INTO CSP
15214            NEXT,   J/PSCCTOSR3=0AA                         ;
        (7162)  DCS[0,00,0,0,0,0]  BM[0000,,10,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,1100,,,1,,0000,0,,,11,000,,,001,110,011]
15215
15216    7163:  ;(FREE)
15217    PSCCTOSR3=0AA:
15218            P2=7,   SR_CSPD(D03);                           ;MOVE P6[CC] TO SR<310>
15219            NEXT,   J/PSCCTOSR3=0BB                         ;
        (7163)  DCS[0,00,0,0,0,0]  BM[1010,,10,00,,00,00,,000,,000,,,0,0,1,,0,,0,,,0,1100,,,0,,0000,0,,,11,000,,,001,110,100]
15220
15221    7164:  ;(FREE)
15222    PSCCTOSR3=0BB:
15223            P0,     BUSDIN_EMIT=[I],                        ;RESET EMIT FOR CONSTANTS
15224                    EN=CLK=IR[15=00],                       ;AND IR LOADING
15225            NEXT,   J/BUTSR3=0                              ;AND GO TEST THE BITS
        (7164)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,01,,000,,100,,,0,0,0,,0,,0,,,1,1001,,,0,,0000,0,,,11,000,,,010,111,110]
15226
15227
15228
15229
15230
15231    ;.PAGE**********************************************************************************************************************
15232
15233
15234    .TOC    *   TEST620-624:  TESTING UBREAK AND JAMUPP
15235
15236    ;**************************************************************************************
15237    ;*
15238    ;*      TESTS:  620 - 624                               UWORDS: 101 - 133
15239    ;*
15240    ;*      FUNCTIONS:
15241    ;*
15242    ;*      THE FOLLOWING GROUP OF TEST FORCES A JAMUPP CONDITION VIA THE UBREAK FACILITY.
15243    ;*      TESTS ARE THEN PERFORMED ON JAMUPP RELATED SIDE EFFECTS:
15244    ;*
15245    ;*      ALL JAM CONDITION BITS CAN BE RESET; ACTIVE BUT NON, CSP WRITE BIT ARE CLEARED
15246    ;*      IN WORD WHICH JAM OCCURS; NON-INTERNAL-JAM IS SET; CUA IS LOCKED, UNLOCKED BY
15247    ;*      BUTA(TRACK); PREFETCH=JAM SET/CLEARED.
15248    ;*
15249    ;**************************************************************************************
15250
15251
15252
15253    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
15254
15255    ;*** TEST 620 ***
15256
15257    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
```



```
15258
15259    ;* PART A ***
15260    ;TEST-620-A DOES A 'CLR-JAM-ERRORS' UCON-I=0 FUNCTION, THEN CHECKS 'INIT JAM'=0
15261    5511:
15262    TEST620A:
15263            P0,     LOAD-ENUA(ZTARGET406),                  ;EXPECTED ADDRESS AFTER 'BUT'
15264                    LOAD-ERROR(TEST620A),                   ;ERROR DIRECTORY KEY
15265                    DCS-CTR(CS,),                           ;COMPARE AT TARGET
15266            NEXT,   J/BCERC620A                             ;
        (5511)  DCS[1,00,1,0,0,0]  BM[1010,,00,11,,11,00,,000,,110,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,101,111,000]
15267
15268    5570:
15269    BCERC620A:
15270            P0,     BUMP-VERIFY,                            ;COUNT
15271            P7,     CLR-JAM-ERRORS=[1],                     ;CLEAR OUT ERROR REGISTERS
15272            NEXT,   J/GOBUT620A                             ;
        (5570)  DCS[0,00,0,0,0,1]  BM[0100,,00,00,,10,00,,000,,000,,,0,0,0,,0,,0,,,1,1011,,,0,,0000,0,,,11,000,,,011,110,110]
15273
15274    5636:  ;(FREE)
15275    GOBUT620A:
15276            SETUP,  RETURN/TEST620B,                        ;RETURN TO START OF NEXT SUBTEST
15277            NEXT,   GOTO-PAGE(7),                           ;BUT TABLE IS ON PAGE 7,
15278                    J/BUT=INITJAM                           ;GO DO MULTIPLE 'BUT' ON INIT JAM
        (5366)  DCS[0,00,0,0,0,0]  BM[0101,,00,11,,00,10,,011,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,011,110]
15279
15280
15281    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
15282
15283    ;* PART B *
15284    ;TEST-620-B CHECKS THAT 'OTHER-JAM-H'=0 ALSO
15285    5623:
15286    TEST620B:
15287            P0,     LOAD-ENUA(ZTARGET401),                  ;BIT<01> CLEAR
15288                    LOAD-ERROR(TEST620B),                   ;ERROR DIRECTORY KEY
15289                    DCS-CTR(C),                             ;COMPARE AT TARGET
15290            NEXT,   J/GOBUT620B                             ;
        (5623)  DCS[1,00,1,0,0,0]  BM[1100,,00,11,,11,00,,000,,001,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,011,110,111]
15291
15292    5367:  ;(FREE)
15293    GOBUT620B:
15294            SETUP,  RETURN/TEST620C,                        ;RETURN TO START OF NEXT SUBTEST
15295            NEXT,   GOTO-PAGE(7),                           ;BUT TABLE IS ON PAGE 7
15296                    J/BUT=OTHERJAM                          ;GO DO BUT ON OTHER-JAM-H SIGNAL
        (5367)  DCS[0,00,0,0,0,0]  BM[0101,,00,11,,00,10,,001,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,101,010]
15297
15298
15299    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
15300
15301    ;* PART C *
15302    ;TEST-620-C CHECKS THAT STATUS MUX PORT 2 (JAM REG) READS [001000] WHEN RESET
15303    ;       "CLEAR-JAM-ERRORS" FUNCTION CLEARS BITS<15,13,11,8-2,0> OF JAM-REG
15304    ;       "CLR-YELLOW-ZONE" FUNCTION CLEARS BIT<12> OF JAM-REG
```

```
!         BIT<9> IS ACTIVE LOW, READS AS "1"N
!         BITS<10,1> READ "0" SINCE NO WCS PRESENT
!         BIT<14> IS "0" ALWAYS
5621:
TEST620C:
        P0,     LOAD-ENUA(ZTARGET402),                !SETUP FOR D<15:00>=0 TEST RESULT
                LOAD-ERROR(TEST620C),                 !ERROR DIRECTORY KEY
                DCS-CTR(C11,),                        !COMPARE AT TARGET
        P3,     BUTA(CLR-FLAG-REG-UCON),              !PUT EMIT ONTO BUSDIN, CLEAR OUT I-O UCON
        NEXT,   J/MASK620C                            !
  (5621)  DCS[1,00,1,0,0,0]  BM[0100,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,010,,,011,111,000]

5370:   !(FREE)
MASK620C:
        P0,     BUMP-VERIFY,                          !COUNT
        P3,     CSPD[04]_EMIT,                        !DON'T NEED TO MASK ANYTHING
        EMITC,  EMIT/177777,                          !
        NEXT,   J/GETJAM620C                          !
  (5370)  DCS[0,00,0,0,0,1]  BM[1111,,10,11,,11,11,,111,,111,,,0,0,0,,0,,0,,,0,1011,,,1,,0000,0,,,11,000,,,011,111,001]

5371:   !(FREE)
GETJAM620C:
        SETUP,  RETURN/TEST621A,                      !GO EXECUTE SUBR WHICH:
        P0,     BUMP-VERIFY,                          !COUNT
        NEXT,   CALL[CLRJAMTOD]                       !  (JAMREG)-XOR-CSP(02)/(001000) => D, BUT(D=ZERO)
  (5371)  DCS[0,00,0,0,0,1]  BM[0101,,00,11,,00,01,,111,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,101,101]





! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

!*** TEST 621 ***
!CAUSE A MICROBREAK JAM, AND CHECK ALL THE APPROPRIATE SIGNALS ARE SET

! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

!* PART A *
!TEST-621-A CAUSES A MICROBREAK JAM AT A SPECIFIC MICROADDRESS, AND CHECKS
!THAT THE MICROCODE JAMS TO LOCATION (4777) IMMEDIATELY
5617:
TEST621A:
        P0,     LOAD-ENUA(4777),                      !SETUP JAMUPP ADDRESS
                LOAD-ERROR(TEST621A),                 !ERROR DIRECTORY KEY
                DCS-CTR(C9,),                         !COMPARE AT JAMUPP WORD
        P3,     BUTA(CUA-TRACK),                      !RESET CUA TRACKING IF HASN'T BEEN
        NEXT,   J/CSP1L621A                           !
  (5617)  DCS[1,00,1,0,0,0]  BM[0110,,00,10,,01,11,,111,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,001,,,011,111,010]

5372:   !(FREE)
CSP1L621A:
        P0,     BUMP-VERIFY,                          !COUNT
```



```
        EMITC,  EMIT/100377,                          !
        P2-U,   IR_EMIT,                              !A BRANCH INSTR, SO 'PREFETCH H' = "0"
        P3,     CSPD[05]_EMIT,                        !FLAG<8>="1", FOR UBREAK EN, EXFLAG<2:1>="11", FOR ACTIVE BUT TEST
        NEXT,   J/SETBRK621A                          !BIT00="1", FOR JAMUPP ROUTINE
  (5372)  DCS[0,00,0,0,0,1]  BM[1000,,10,00,,00,11,,111,,111,,,0,0,0,,0,,0,,,1,1010,,,1,,0000,0,,,11,000,,,011,111,011]
                                                      !NOTE CSPADDR(05)/UCON-OPERATION BIT OVERLAP

5373:   !(FREE)
SETBRK621A:
        SELECT, UCON-PROC,                            !PROCESSOR UCON:
        ENABLE, EN-CLK-UBREAK(11-00),                 !  FOR UBREAK REG LOAD
                BUSDIN_EMIT(15-00),                   !  AND KEEP EMIT ON BUSDIN
        P0,     SET-UCON-CONTROL,                     !WRITE UCON REGISTER
        NEXT,   J/LODBRK621A                          !
  (5373)  DCS[0,00,0,0,0,0]  BM[0000,,01,00,,00,01,,000,,000,,,0,0,0,,0,,0,,,1,1001,,,0,,0000,0,,,11,000,,,011,111,100]

5374:   !(FREE)
LODBRK621A:
        P2-T,   SR_ZERO,                              !ZERO SR<00>, TO PREVENT SPURIOUS UBREAKS FROM GETTING THRU
                UBREAK_BUSDIN(11-00),                 !LOAD MICROBREAK REGISTER
                EMITHL/5522,                          !  WITH SELECTED ADDRESS FROM EMIT
        NEXT,   J/SETRET621A                          !
  (5374)  DCS[0,00,0,0,0,0]  BM[0011,,00,10,,11,01,,010,,010,,,0,0,1,,0,,0,,,1,1010,,,0,,0000,0,,,11,000,,,011,111,101]

5375:   !(FREE)
SETRET621A:
        P0,     BUMP-VERIFY,                          !COUNT
        P3,     CSPD[00]_EMIT, RETURN/TEST621B,       !RETURN ADDRESS FOR AFTER JAMUPP
        NEXT,   J/SETFLG621A                          !
  (5375)  DCS[0,00,0,0,0,1]  BM[0101,,10,11,,00,01,,011,,000,,,0,0,0,,0,,0,,,0,1111,,,1,,0000,0,,,11,000,,,011,111,111]

5377:   !(FREE)
SETFLG621A:
        P2-T,   D_CSPD(005), D[C]_0,                  !GET VALUES TO LOAD INTO FLAGS
        P3,     ASPHI[16]_D,                          !  AND SAVE IN ASP FOR COMPARE LATER
        NEXT,   J/LODFLG621A                          !
  (5377)  DCS[0,00,0,0,0,0]  BM[1010,,10,00,,00,00,,011,,000,,,0,1,0,,0,,0,,,0,1010,,,0,,1001,0,,,11,000,,,100,000,100]

5400:   !(FREE)
LODFLG621A:
        P0,     BUSDIN_EMIT-[I],                      !KEEP IT ON
                BUMP-VERIFY,                          !COUNT
        P3,     FLAG[8-0]_D[15-8]-[I],                !ENABLE U-BREAK, FLAG 8; SET EXFLAGS FOR LATER
        NEXT,   J/SETSR621A                           !
  (5400)  DCS[0,00,0,0,0,1]  BM[0000,,00,00,,00,01,,000,,001,,,0,0,0,,0,,0,,,1,1011,,,0,,0000,0,,,11,000,,,100,000,001]


5401:   !(FREE)
SETSR621A:
        P2-T,   SR_CSPD(005),                         !MAKE SR<00>=1, FOR JAMUPP EXPECTED
        NEXT,   J/UBRK621A                            !(FOR UBREAK JAM EXPECTED)
  (5401)  DCS[0,00,0,0,0,0]  BM[1010,,10,00,,00,00,,000,,000,,,0,0,1,,0,,0,,,0,1010,,,0,,0000,0,,,11,000,,,101,010,010]

5522:
UBRK621A:                                             !*** MICROBREAK HERE ***
```

```
15401           P3-T,   D_ZERO, D[C]_ALU15;                     ;UWORD LATCH FOR CLK-D NOT CLEARED, CLK-D SHOULD HAPPEN
15402                   SUTA(CLR-FLAG-RES-UCON),                ;JAMUPP CLEAR SHOULD ZAP ACTIVE BUT LATCH, SUTA(CLR-,..) SHOULDN'T H
15403                                                           ;CUA GETS LOCKED WITH UADDR OF THIS UWORD (5522) ON JAMUPP
15404                                                           ;PREFETCH=JAM(1)N GETS PREFETCH=N (="0") AT JAMUPP
15405           P3,     CSPD[05]_EMIT, EMIT/030004,             ;LATCH HOLDING "WRCSP" BIT SHOULD GET ZAPPED ON JAMUPP.
15406                                                           ;  THIS WRITE SHOULDN'T HAPPEN
15407           NEXT,   J/ERROR621A                             ;THIS UPP SHOULD NOT BE USED
      (5522)  DCS[0,00,0,0,0,0]  BM[0011,,10,00,,00,00,,000,,100,,,1,1,0,,0,,0,,,0,1010,,,1,,0000,0,,,11,010,,,110,001,101]
15408
15409
15410   ;(4777)  JAMUPP001:  ***COMPARE ENABLED ABOVE DONE HERE***
15411   ;               THIS WORD TESTS SR<00>, WHICH SHOULD BE SET
15412   ;               IF SR<00>=1, GOTO(JAMUPP002B), IF SR<00>=0, GOTO(JAMUPP003) (ERROR)
15413   ;               P3-T,   SR_D    SAVE OLD D IN SR, FOR NOW
15414   ;
15415   ;(4757)  JAMUPP002A:  P2-T,  D_CSPD(00)         GET RETURN ADDRESS INTO D
15416   ;
15417   ;(4XXX)  JAMUPP002C:  P0,    RETURN_D   LOAD RETURN ADDRESS
15418   ;                     P2-T,  D_SR       RESTORE OLD D FROM SR
15419   ;
15420   ;(7XXX)  JAMUPP002D:  P2-T,  SR_ZERO,  ZERO OUT SR, JAMUPPS NOW ILLEGAL
15421   ;                     NEXT,  SUTA(RETURN)       AND NOW RETURN
15422   ;
15423
15424   ;EXECUTE THE FOLLOWING WORD ONLY IF NO JAMUPP OCCURRED:
15425   5615:
15426   ERROR621A:
15427           P0,     LOAD=ENUA(0000),                        ;FORCE ERROR
15428                   LOAD=ERROR(ERROR621A),                  ;ERROR DIRECTORY KEY
15429                   DCS=CTR(C0,),                           ;FORCE COMPARE AT P3-T
15430                   BUMP-VERIFY,                            ;COUNT
15431           NEXT,   J/TEST621B                              ;
      (5615)  DCS[1,00,1,0,0,1]  BM[1111,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,110,001,011]
15432
15433
15434   ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
15435
15436   ;* PART B *
15437   ;TEST-621-B CHECKS THAT D WAS ZEROED, IE, P3 PULSE WAS SUPPRESSED IN JAM WORD,
15438   ;  BUT UWORD LATCH FOR CLK-D BIT NOT ZAPPED, SO BIT WAS SAVED FOR EXECUTE LATER
15439   5613:
15440   TEST621B:
15441           P0,     LOAD=ENUA(ZTARGET402),                  ;SETUP FOR D = ZERO TEST
15442                   LOAD=ERROR(TEST621B),                   ;ERROR DIRECTORY KEY
15443                   DCS=CTR(C3,),                           ;COMPARE AT TARGET
15444           NEXT,   J/GOBUT621B                             ;
      (5613)  DCS[1,00,1,0,0,0]  BM[1100,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,100,000,010]
15445
15446   54021  ;(FREF)
15447   GOBUT621B:
15448           SETUP,  RETURN/TEST621C,                        ;RETURN TO START OF NEXT SUBTEST
15449           NEXT,   GOTO=PAGE(7),                           ;BUT TABLE ON PAGE 7
15450                   J/BUTD=IS=ZERO                          ;BUT ON D CONTENTS
```



```
      (5402)  DCS[0,00,0,0,0,0]  BM[0101,,00,11,,00,01,,001,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,100,001]
15451
15452
15453   ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
15454
15455   ;* PART C *
15456   ;TEST-621-C CHECKS THAT *INIT JAM* STILL OFF
15457   5611:
15458   TEST621C:
15459           P0,     LOAD=ENUA(ZTARGET406),                  ;BIT<00> CLEAR
15460                   LOAD=ERROR(TEST621C),                   ;ERROR DIRECTORY KEY
15461                   DCS=CTR(C4,),                           ;COMPARE AT TARGET
15462           NEXT,   J/GOBUT621C                             ;
      (5611)  DCS[1,00,1,0,0,0]  BM[1011,,00,11,,11,00,,000,,110,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,100,000,011]
15463
15464   5603:  ;(FREF)
15465   GOBUT621C:
15466           SETUP,  RETURN/TEST621D,                        ;RETURN TO START OF NEXT SUBTEST
15467           NEXT,   GOTO=PAGE(7),                           ;BUT TABLE ON PAGE 7
15468                   J/BUTNINITJAM                           ;GO TEST INIT JAM
      (5603)  DCS[0,00,0,0,0,0]  BM[0101,,00,11,,00,00,,111,,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,011,110]
15469
15470
15471   ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
15472
15473   ;* PART D *
15474   ;TEST-621-D CHECKS THAT *OTHER-JAM-H*=1, WAS SET FOR UBREAK
15475   5607:
15476   TEST621D:
15477           P0,     LOAD=ENUA(ZTARGET403),                  ;BIT<01> SET
15478                   LOAD=ERROR(TEST621D),                   ;ERROR DIRECTORY KEY
15479                   DCS=CTR(C3,),                           ;COMPARE AT TARGET
15480           NEXT,   J/GOBUT621D                             ;
      (5607)  DCS[1,00,1,0,0,0]  BM[1100,,00,11,,11,00,,000,,011,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,100,000,100]
15481
15482   5604:  ;(FREF)
15483   GOBUT621D:
15484           SETUP,  RETURN/TEST621E,                        ;RETURN TO START OF NEXT SUBTEST
15485           P0,     BUMP-VERIFY,                            ;COUNT
15486           NEXT,   GOTO=PAGE(7),                           ;BUT TABLE ON PAGE 7
15487                   J/BUTOTHERJAM                           ;GO TEST *OTHER-JAM-H* IS SET
      (5604)  DCS[0,00,0,0,0,1]  BM[0101,,00,11,,00,00,,101,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,101,010]
15488
15489
15490   ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
15491

15492   ;* PART E *
15493   ;TEST-621-E CHECKS UBPEAK ONLY BIT SET FROM UBREAK JAMUPP IN STATUS MUX PORT 2 (JAM REG)
15494   5605:
15495   TEST621E:
15496           P0,     LOAD=ENUA(ZTARGET402),                  ;SETUP FOR D = ZERO COMPARE
15497                   LOAD=ERROR(TEST621E),                   ;ERROR DIRECTORY KEY
```

```
15498                   DCS-CTR(C10,),                          ;COMPARE AT TARGET
15499           NEXT,   J/EXPEC621E                             ;
     (5604)  DCS(1,00,1,0,0,0)  BM(0101,,00,11,,11,00,,000,,010...0,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,100,000,101)
15500
15501   5405:  ;(FREE)
15502   EXPEC621E:
15503           P3,     CSPD(02)_EMIT,                          ;WHAT WE EXPECT TO SEE IN JAM IS:
15504           EMITC,  EMIT/001001,                            ;  (0010011)= UBREAK H IN BIT00
15505           NEXT,   J/GETJAM621E                            ;
     (5405)  DCS(0,00,0,0,0,0)  BM(0000,,10,00,,10,00,,000,,001...0,0,0,0,,0,,0,,,0,1101,,,1,,0000,0,,,11,000,,,100,000,110)
15506
15507   5406:  ;(FREE)
15508   GETJAM621E:
15509           SETUP,  RETURN/TEST621F,                        ;GO TO SUBR WHICH:
15510           NEXT,   CALL(JAMTOD)                            ;  (JAMREG)-XOR-CSP(02) => D, BUT(D=ZERO)
     (5406)  DCS(0,00,0,0,0,0)  BM(0101,,00,11,,00,00,,011,,111...0,0,0,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,101,110)
15511
15512
15513
15514   ; * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - *
15515
15516   ;* PART F *
15517   ;TEST-621-F CHECKS THAT THE RIGHT CUA WAS LOCKED, AND SHORT TERM FLAGS NOT CLEARED (IE,
15518   ;  BUT4(CLR-FLAG+...) IN UBREAK WORD DIDN'T CLEAR EXFLAG11), INDICATING JAMUPP CLEAR L
15519   ;  DID IN FACT ZAP THE ACTIVE BUT ROM LATCH.  ALSO PREFETCH*JAM(1)H GETS PREFETCH-H="0".
15520   5603:
15521   TEST621F:
15522           P0,     LOAD-ENUA(ZTARGET402),                  ;SETUP FOR D = ZERO COMPARE
15523                   LOAD-ERROR(TEST621F),                   ;ERROR DIRECTORY KEY
15524                   DCS-CTR(C10,),                          ;COMPARE AT TARGET
15525           NEXT,   J/EXPEC621F                             ;
     (5603)  DCS(1,00,1,0,0,0)  BM(0101,,00,11,,11,00,,000,,010...0,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,100,000,111)
15526
15527   5407:  ;(FREE)
15528   EXPEC621F:
15529           P0,     BUMP-VERIFY,                            ;COUNT
15530           P3,     CSPD(02)_EMIT,                          ;(USE MASK OF ALL 1'S FROM BEFORE)
15531           EMITC,  EMIT/055226,                            ;WHAT THE CUA-EXFLAG-FOYP PORT OF HBMUX SHOULD BE
15532           NEXT,   J/GETCUA621F                            ;CUA=(5522), EXFLAG<2:1>="11", PREFETCH*JAM(1)H="0"
     (5407)  DCS(0,00,0,0,0,1)  BM(0101,,10,10,,10,10,,010,,110...0,0,0,,0,,0,,,0,1101,,,1,,0000,0,,,11,000,,,100,001,000)
15533
15534   5410:  ;(FREE)
15535   GETCUA621F:
15536           SETUP,  RETURN/TEST621G,                        ;GO TO SUBR WHICH:
15537           P0,     BUMP-VERIFY,                            ;COUNT
15538           NEXT,   CALL(CUATOD)                            ;  (CUA)-XOR-CSP(02) => D, BUT(D=ZERO)
     (5410)  DCS(0,00,0,0,0,1)  BM(0101,,00,11,,11,10,,110,,111...0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,101,111)
15539
15540
15541
15542
15543
```



```
15544
15545   ; * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - *
15546
15547   ;* PART G *
15548   ;TEST-621-G CHECKS THAT 'PREFETCH*JAM(1)H' GOT 'PREFETCH-H'="0" AFTER JAMUPP
15549   5766:
15550   TEST621G:
15551           P0,     LOAD-ENUA(ZTARGET401),                  ;BIT01 CLEAR
15552                   LOAD-ERROR(TEST621G),                   ;ERROR DIRECTORY KEY
15553                   DCS-CTR(C3,),                           ;COMPARE AT TARGET
15554           NEXT,   J/GOBUT621G                             ;
     (5766)  DCS(1,00,1,0,0,0)  BM(1100,,00,11,,11,00,,000,,001...0,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,100,001,001)
15555
15556   5411:  ;(FREE)
15557   GOBUT621G:
15558           SETUP,  RETURN/TEST621H,                        ;RETURN TO START OF NEXT SUBTEST
15559           P0,     BUMP-VERIFY,                            ;COUNT
15560           NEXT,   GOTO-PAGE(7),                           ;BUT TABLE ON PAGE 7
15561                   J/BUTPREFETCHJAM                        ;PREFETCH*JAM(1)H IN BIT01
     (5411)  DCS(0,00,0,0,0,1)  BM(0101,,00,11,,11,01,,100,,111...0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,110,000)
15562
15563
15564
15565
15566   ; * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - *
15567
15568   ;* PART H *
15569   ;TEST-621-H CHECKS THAT CSP(05) DID NOT GET WRITTEN IN THE JAMMED WORD ABOVE,
15570   ;  AND IS IN FACT, EQUAL TO THE SAVED COPY OF ITS CONTENTS, IN ASPHI(16)
15571   5754:
15572   TEST621H:
15573           P0,     LOAD-ENUA(ZTARGET402),                  ;SETUP FOR D=ZERO COMPARE
15574                   LOAD-ERROR(TEST621H),                   ;ERROR DIRECTORY KEY
15575                   DCS-CTR(C4,),                           ;COMPARE AT TARGET
15576                   BUMP-VERIFY,                            ;COUNT
15577           NEXT,   J/COMP621H                              ;
     (5754)  DCS(1,00,1,0,0,1)  BM(1011,,00,11,,11,00,,000,,010...0,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,100,001,010)
15578
15579   5412:  ;(FREE)
15580   COMP621H:
15581           P2=T,   D_CSPD(05)-XOR-ASPHI(R16),               ;COMPARE CURRENT:SAVED, SHOUL BE SAME
15582           NEXT,   GOTO-PAGE(7),                           ;XPER
15583                   J/GOBUT621H                             ;
     (5412)  DCS(0,00,0,0,0,0)  BM(0110,,10,00,,11,00,,011,,111...0,1,0,,0,,0,,,0,1010,,,0,,0000,0,,,11,100,,,001,110,000)
15584
15585   7160:  ;(FREE)

15586   GOBUT621H:
15587           SETUP,  RETURN/TEST622A,                        ;RETURN TO START OF NEXT SUBTEST
15588           NEXT,   GOTO-PAGE(7),                           ;BUT TABLE ON PAGE 7
15589                   J/BUTD-IS-ZERO                          ;TEST FOR EQUALITY
     (7160)  DCS(0,00,0,0,0,0)  BM(0101,,00,11,,00,00,,001,,111...0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,100,001)
15590
```

```
    ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

    !*** TEST 622 ***
    !DO A 'CLR-JAM-ERRORS' FUNCTION TO CLEAR OUT SET BITS
    !MAKE SURE THAT 'START-DELAY', 'CLR-NPR-TIMEOUT' DON'T CLEAR THEM

    ! - - - - - - - - - - - - - - - - - - - - - - - - - - -

    !* PART A *
    !TEST-622-A CHECKS THAT 'START-DELAY', 'CLR-NPR-TIMEOUT' DON'T AFFECT
    !THE 'CLR-JAM-ERRORS' FUNCTION'
    5601:
    TEST622A:
            P0,     LOAD-ENUA(ZTARGET403),                  !SETUP FOR BUT ON 'OTHER-JAM-H'= (1)
                    LOAD-ERROR(TEST622A),                   !ERROR DIRECTORY KEY
                    DCS-CTR(C5,),                           !COMPARE AT TARGET
            NEXT,   J/BC1FCN622A                            !
(5601)  DCS[1,00,1,0,0,0]  BM[1010,,00,11,,11,00,,000,,011,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,100,001,011]

    5413:  !(FREE)
    BC1FCN622A:
            P0,     BUMP-VERIFY,                            !COUNT
            P2,     START-DELAY-[I],                        !DO A 'START-DELAY'
            NEXT,   J/CNSTO622A                             !
(5413)  DCS[0,00,0,0,0,(]  BM[0100,,00,00,,01,00,,000,,000,,,0,0,0,,0,,0,,,1,1011,,,0,,0000,0,,,11,000,,,100,001,100]

    5414:  !(FREE)
    CNSTO622A:
            P2,     CLR-NPR-TIMEOUT-[I],                    !DO A 'CLR-NPR-TIMEOUT'
            NEXT,   J/GOBUT622A                             !NEITHER OF THESE SHOULD ASSERT 'CLR-JAM-ERRORS'
(5414)  DCS[0,00,0,0,0,0]  BM[0100,,00,00,,11,00,,000,,000,,,0,0,0,,0,,0,,,1,1011,,,0,,0000,0,,,11,000,,,100,001,101]

    5415:  !(FREE)
    GOBUT622A:
            SETUP,  RETURN/TEST622B,                        !RETURN TO START OF NEXT SUBTEST
            NEXT,   GOTO-PAGE(7),                           !BUT TABLE ON PAGE 7
                    J/BUTOTHERJAM                           !TEST 'OTHER-JAM-H' STILL SET
(5415)  DCS[0,00,0,0,0,0]  BM[0101,,00,10,,11,11,,111,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,101,010]


    ! - - - - - - - - - - - - - - - - - - - - - - - - - - -

    !* PART B*
    !TEST-622-B NOW CHECKS THAT 'CLR-JAM-ERRORS' DOES JUST THAT
    5577:
    TEST622B:
            P0,     LOAD-ENUA(ZTARGET401),                  !SETUP FOR BUT ON 'OTHER-JAM-H'=(0)
                    LOAD-ERROR(TEST622B),                   !ERROR DIRECTORY KEY
                    DCS-CTR(C4,),                           !COMPARE AT TARGET
            NEXT,   J/BCERC622B                             !
(5577)  DCS[1,00,1,0,0,0]  BM[1011,,00,11,,11,00,,000,,001,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,100,001,110]
```



```
    5416:  !(FREE)
    BCERC622B:
            P2,     CLR-JAM-ERRORS-[I],                     !DO IT
            NEXT,   J/GOBUT622B                             !
(5416)  DCS[0,00,0,0,0,0]  BM[0100,,00,00,,10,00,,000,,000,,,0,0,0,,0,,0,,,1,1011,,,0,,0000,0,,,11,000,,,100,001,111]

    5417:  !(FREE)
    GOBUT622B:
            SETUP,  RETURN/TEST622C,                        !RETURN TO START OF NEXT SUBTEST
            NEXT,   GOTO-PAGE(7),                           !BUT TABLE ON PAGE 7,
                    J/BUTOTHERJAM                           !TEST 'OTHER-JAM-H' NOW CLEAR
(5417)  DCS[0,00,0,0,0,0]  BM[0101,,00,10,,11,11,,101,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,101,010]


    ! - - - - - - - - - - - - - - - - - - - - - - - - - - -

    !* PART C *
    !TEST-622-C TESTS THAT STATUS MUX PORT 2 (JAM REG) READS (001000) WHEN RESET
    5575:
    TEST622C:
            P0,     LOAD-ENUA(ZTARGET402),                  !SETUP FOR D = ZERO TEST
                    LOAD-ERROR(TEST622C),                   !ERROR DIRECTORY KEY
                    DCS-CTR(C10,),                          !COMPARE AT TARGET
            P3,     BUTA(CLR-FLAG-REG-UCON),                !PUT ENTY ON BUSDIN, CLEAR SHORT TERM FLAGS
            NEXT,   J/GETJAM622C                            !
(5575)  DCS[1,00,1,0,0,0]  BM[0101,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,010,,,100,010,000]

    5420:  !(FREE)
    GETJAM622C:
            SETUP,  RETURN/TEST623,                         !GO TO SUBR WHICH:
            NEXT,   CALL(CLRJAMTOD)                         !  (JAMREG)-XOR-CSP(02)/(001000) -> D, BUT(D=ZERO)
(5420)  DCS[0,00,0,0,0,0]  BM[0101,,00,10,,11,11,,011,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,101,101]







    ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

    !*** TEST 623 ***
    !TEST-623 CHECKS THAT BUTA(CUA-TRACK) RESTARTS CUA TRACKING
    5573:
    TEST623:

            P0,     LOAD-ENUA(ZTARGET402),                  !SETUP FOR D = ZERO TEST
                    LOAD-ERROR(TEST623),                    !ERROR DIRECTORY KEY
                    DCS-CTR(C10,),                          !COMPARE AT TARGET
            P3,     BUTA(CUA-TRACK),                        !RESET TO TRACKING CUA MODE
            NEXT,   J/EXPEC623                              !
(5573)  DCS[1,00,1,0,0,0]  BM[0101,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,001,,,100,010,001]
```

```
15686
15687    5421:  ;(FREE)
15688    EXPEC623:
15689            P0,     BUMP=VERIFY,                            ;COUNT
15690            P3,     CSPD[02]_EMIT,                          ;(PREV MASK)
15691            EMITC,  EMIT/013734,                            ;CUA IS WORD WHICH LOADS CUA INTO CSP, ONLY
15692            NEXT,   J/GETCUA623                             ;  EXPL[00:1> CLEARED FROM BUTA(CLR=FLAG=,,,), ABOVE
     (5421)  DCS[0,00,0,0,0,1]  BM[0111,,10,01,,11,11,,011,,100,,,0,0,0,,0,,0,,,0,1101,,,1,,0000,0,,,11,000,,,100,010,010]
15693
15694    5422:  ;(FREE)
15695    GETCUA623:
15696            BSTUP,  RETURN/SCOPE623,                        ;GO TO SUBR WHICH:
15697                                                            ;  CUA=(WORD WHICH READS CUA), SINCE TRACKING RESET
15698            NFXT,   CALL[CUATOD]                            ;  (CUB]=ROW=CSP(02) => D, BUT(D=ZERO)
     (5422)  DCS[0,00,0,0,0,0]  BM[0101,,00,10,,00,10,,011,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,101,111]
15699
15700
15701    5423:  ;(FREE)
15702    SCOPE623:
15703            P0,     BUSDIN_EMIT=[I],                        ;RESET PROC UCONS
15704                    EN=CLK=IR[15=00],                       ;IR GETS JUNK, BUT DON'T CARE
15705            P3,     FLAG[8=0]_D[15=8]=[I],                  ;RESET FLAGS TO ALL ZERO
15706            NEXT,   BUTD[SCOPE],                            ;NO ERROR:  "TEST624A" [+1, WORDS]
15707                    J/TEST624A                              ;   ERROR:  "BCERC620A" [-48. WORDS]
     (5423)  DCS[0,00,0,1,0,0]  BM[0000,,00,00,,00,01,,000,,101,,,0,0,0,,0,,0,,,1,1011,,,0,,0000,0,,,11,000,,,101,111,001]
15708
15709
15710
15711
15712
15713    ;- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
15714
15715    ;*** TEST 624 ***
15716    ;DO ANOTHER UBREAK JAM, CHECK JAM OCCURS, AND THAT P2 IS SEEN IN JAM WORD
15717
15718    ;- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
15719
15720    ;* PART A *
15721    ;TEST-624-A CAUSES A UBREAK JAM, AND CHECKS TO SEE IT OCCURS
15722
15723    5571:
15724    TEST624A:
15725            P0,     LOAD-EMUA(4777),                        ;SETUP JAMUPP ADDRESS
15726                    LOAD-ERROR(TEST624A),                   ;ERROR DIRECTORY KEY
15727                    DC8-CTR(C9.),                           ;COMPARE AT JAMUPP WORD
15728            NEXT,   J/LODDIR624A                            ;
     (5571)  DCS[1,00,1,0,0,0]  BM[0110,,00,10,,01,11,,111,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,101,110,010]
15729
15730    5562:
15731    LODIR624A:
15732            P2=0,   IR_EMIT,                                ;IR PATTERN ASSERTS PREFETCH=N:
15733                    EMIT/030603,                            ;  PREFETCH=N=(DOPH=NOVBL=&=(QN=&R7L=DMQH=DR7L)
15734            P3,     FLAG[8=0]_D[15=8],                      ;FLAGS GET D, WHICH IS ZERO




15735            NEXT,   J/SETBRK624A                            ;       + (SOPH=NEGL=ESCL=RORL=&BRL=DMQH=DR7L)
     (5562)  DCS[0,00,0,0,0,0]  BM[0011,,00,00,,01,10,,000,,011,,,0,0,0,,0,,0,,,1,1010,,,0,,0000,0,,,11,000,,,100,010,100]
15736
15737    5424:  ;(FREE)
15738    SETBRK624A:
15739            SELECT, UCON=PROC,                              ;ENABLE PROCESSOR UCON:
15740            ENABLE, EN=CLK=UBREAK[11=00],                   ;  EN UBREAK LOAD,
15741                    BUSDIN_EMIT[15=00],                     ;  KEEP EMIT ON BUSDIN
15742            P0,     SET-UCON-CONTROL,                       ;WRITE CONTROLS
15743                    BUMP=VERIFY,                            ;COUNT
15744            NEXT,   J/LODBRK624A                            ;
     (5424)  DCS[0,00,0,0,0,1]  BM[0000,,01,00,,00,01,,000,,000,,,0,0,0,,0,,0,,,1,1001,,,0,,0000,0,,,11,000,,,100,010,101]
15745
15746    5425:  ;(FREE)
15747    LODBRK624A:
15748            P0,     BUMP=VERIFY,                            ;COUNT
15749            P2=T,   UBREAK_BUSDIN[11=00],                   ;LOAD UBREAK REGISTER
15750                    EMITNL/6255,                            ;  WITH SELECTED ADDRESS FROM EMIT
15751            NEXT,   J/SETRET624A                            ;
     (5425)  DCS[0,00,0,0,0,1]  BM[0000,,00,11,,00,10,,101,,101,,,0,0,0,,0,,0,,,1,1010,,,0,,0000,0,,,11,000,,,100,010,110]
15752
15753    5426:  ;(FREE)
15754    SETRET624A:
15755            P3,     CSPD[00]_EMIT, RETURN/TEST624B,         ;RETURN ADDRESS FOR AFTER JAMUPP
15756            NFXT,   GOTO-PAGE(6),                           ;  RETURN TO START OF NEXT SUBTEST
15757                    J/SFTD624A                              ;
     (5426)  DCS[0,00,0,0,0,0]  BM[0101,,10,10,,11,10,,101,,110,,,0,0,0,,0,,0,,,0,1111,,,1,,0000,0,,,11,100,,,011,000,110]
15758
15759    6306:  ;(FREE)
15760    SETD624A:
15761            P2=T,   D_CSPD(D05), D[C]_ALU15,                ;SETUP D WITH VALUE TO GO INTO FLAGS<8:10>
15762            NEXT,   J/SETFLG624A                            ;
     (6306)  DCS[0,00,0,0,0,0]  BM[1010,,10,00,,00,00,,000,,100,,,0,1,0,,0,,0,,,0,1010,,,0,,0000,0,,,11,000,,,100,101,011]
15763
15764    6453:  ;(FREE)
15765    SETFLG624A:
15766            P0,     BUSDIN_EMIT=[I],                        ;KEEP IT ON
15767            P3,     FLAG[8=0]_D[15=8]=[I],                  ;SET UBRK FLAG<8>
15768            NEXT,   J/SETBR624A                             ;
     (6453)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,01,,000,,001,,,0,0,0,,0,,0,,,1,1011,,,0,,0000,0,,,11,000,,,100,101,100]
15769
15770    6454:  ;(FREE)
15771    SETBR624A:
15772            P2=T,   BR_CSPD(D05),                           ;SET BR<00>=1 FOR UBRK JAMUPP EXPECTED
15773            NEXT,   J/UBRK624A                              ;
     (6454)  DCS[0,00,0,0,0,0]  BM[1010,,10,00,,00,00,,000,,000,,,0,0,1,,0,,0,,,0,1010,,,0,,0000,0,,,11,000,,,010,101,101]

15774
15775    6255:
15776    UBRK624A:                                               ;*** MICROBREAK HERE ***
15777            P2=T,   D_ZERO, D[C]_ALU15,                     ;P2 PULSE SHOULD OCCUR, SO D SHOULD BE ZEROED,
15778                                                            ;CUA GETS LOCKED WITH UADDR OF THIS UWORD (6255) ON JAMUPP
15779                                                            ;PREFETCH=JAM[1]N GETS PREFTCH=N (='1') AT JAMUPP
```

```
        P3,     FLAG(6-0)_D(15-8),                      ;ZERO ALL THE FLAGS IN THIS UWORD.  THE UBREAK SHOULD
                                                        ;STILL OCCUR, HOWEVER, AS IT LOOKS AT THE "MUA",
        NEXT,   J/ERROR624A                             ;UPF SHOULD NOT BE USED.
 (6255)  DCS[0,00,0,0,0,0]  BM[0011,,00,00,,00,00,,000,,100,,,0,1,0,,0,,0,,,1,1010,,,0,,0000,0,,,11,000,,,101,110,111]


;(4777)  JAMUPP001:  ***COMPARE ENABLED ABOVE DONE HERE***
;                THIS WORD TESTS SR<00>, WHICH SHOULD BE SET
;                IF SR<00>=1, GOTO(JAMUPP002B), IF SR<00>=0, GOTO(JAMUPP003) (ERROR)
;                P3-T,   SR_D     SAVE OLD D IN SR, FOR NOW
;
;(4757)  JAMUPP002B:  P2-T,  D_CSPD(00)          GET RETURN ADDRESS INTO D
;
;(4XXX)  JAMUPP002C:  P0,     RETURN_D  LOAD RETURN ADDRESS
;                     P2-T,   D_SR      RESTORE OLD D FROM SR
;
;(?XXX)  JAMUPP002D:  P2-T,  SR_ZERO,  ZERO OUT SR, JAMUPPS NOW ILLEGAL
;                     NEXT,  BUTA(RETURN)       AND NOW RETURN
;



;EXECUTE THE FOLLOWING WORD ONLY IF NO JAMUPP OCCURRED:
6567:
ERROR624A:
        P0,     LOAD-ERUA(0005),                        ;FORCE ERROR
                LOAD-ERROR(ERROR624A),                  ;ERROR DIRECTORY KEY
                DCS-CTR(C0,),                           ;FORCE COMPARE AT P3-T
                BUMP-VERIFY,                            ;COUNT
        NEXT,   GOTO-PAGE(5),                           ;XFER TO 5
                J/TEST624B                              ;
 (6567)  DCS[1,00,1,0,0,1]  BM[1111,,00,00,,00,00,,000,,101,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,101,110,101]


;  - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

;* PART B *
;TEST-624-B CHECKS THAT D WAS ZEROED, IE, P2 DID OCCUR IN JAM WORD
5565:
TEST624B:
        P0,     LOAD-ENUA(ZTARGET434),                  ;SETUP FOR IR=ZERO TEST
                LOAD-ERROR(TEST624B),                   ;ERROR DIRECTORY KEY
                DCS-CTR(C6,),                           ;COMPARE AT TARGET
        NEXT,   J/ZEROIR624B                            ;
 (5565)  DCS[1,00,1,0,0,0]  BM[1001,,00,11,,11,00,,011,,100,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,100,010,111]

5427:  ;(FREE)
ZEROIR624B:
        SETUP,  RETURN/TEST624C,                        ;PUT THE ZEROES IN D INTO THE IR,
        NEXT,   CALL[DINTOIR-5]                         ;  TO NEGATE PREFETCH-H  (000000)=HALT
 (5427)  DCS[0,00,0,0,0,0]  BM[0101,,00,11,,11,11,,110,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,111,011]
```



```
;  - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

;* PART C *
;TEST-624-C CHECKS THAT THE RIGHT CUA WAS LOCKED,
;THAT THE EXFLAGS<2:1>H READ AS "00", AND
;THAT PREFETCH*JAM(1)H = "1", FROM TEST-624-A, EVEN THOUGH TEST-624-B
;  RESET THE IR SO THAT PREFETCH-H="0"
5725:
TEST624C:
        P0,     LOAD-ENUA(ZTARGET402),                  ;SETUP FOR D = ZERO COMPARE
                LOAD-ERROR(TEST624C),                   ;ERROR DIRECTORY KEY
                DCS-CTR(C10,),                          ;COMPARE AT TARGET
        NEXT,   J/EXPEC624C                             ;
 (5725)  DCS[1,00,1,0,0,0]  BM[0101,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,100,011,000]

5430:  ;(FREE)
EXPEC624C:
        P3,     CSPD[02]_EMIT,                          ;(USE MASK OF ALL 1'S FROM BEFORE)
                EMIT/062551,                            ;WHAT THE CUA-EXFLAG-FOVP PORT OF HBMUX SHOULD BE
        NEXT,   J/GETCUA624C                            ;CUA=(625E), EXFLAG="00", PREFETCH*JAM(1)H="1"
 (5430)  DCS[0,00,0,0,0,0]  BM[0110,,10,01,,01,01,,101,,001,,,0,0,0,,0,,0,,,0,1101,,,1,,0000,0,,,11,000,,,100,011,001]

5431:  ;(FREE)
GETCUA624C:
        SETUP,  RETURN/TEST624D,                        ;GO TO SUBR WHICH:
        P0,     BUMP-VERIFY,                            ;COUNT
        NEXT,   CALL[CUATOD]                            ;  (CUA)-XOR-CSP(02) => D, BUT(D=ZERO)
 (5431)  DCS[0,00,0,0,0,1]  BM[0101,,00,11,,11,01,,110,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,101,111]


;  - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

;* PART D *
;TEST-624-D CHECKS THAT "PREFETCH*JAM(1)H" GOT "PREFETCH-H"="1" AFTER JAMUPP
5756:
TEST624D:
        P0,     LOAD-ENUA(ZTARGET403),                  ;BIT01 SET
                LOAD-ERROR(TEST624D),                   ;ERROR DIRECTORY KEY
                DCS-CTR(C3,),                           ;COMPARE AT TARGET
        NEXT,   J/GOBUT624D                             ;
 (5756)  DCS[1,00,1,0,0,0]  BM[1100,,00,11,,11,00,,000,,011,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,100,011,010]

5432:  ;(FREE)
GOBUT624D:
        SETUP,  RETURN/CLEAR624,                        ;RETURN TO SCOPE LOOP TEST WORD
        NEXT,   GOTO-PAGE(7),                           ;BUT TABLE ON PAGE 7
                J/BUTPREFETCHJAM                        ;PREFETCH*JAM(1)H IN BIT01
 (5432)  DCS[0,00,0,0,0,0]  BM[0111,,00,00,,11,10,,101,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,110,000]
```

```
15875
15876
15877    7165:  ;(FREE)
15878    CLEAR624:
15879            SETUP,  RETURN/SCOPE624,                        ;RETURN FOR SCOPE LOOP TEST
15880            NEXT,   CALL!CLEAR-I-C-B)                       ;CLEAR OUT WHAT I'VE DONE
      (7165)  DCS[0,00,0,0,0,0]  BM[0101,,00,01,,00,10,,100,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,011,000]
15881
15882
15883    5224:  ;(FREF)
15884    SCOPE624:
15885            P0,     BUMP-VERIFY,                            ;COUNT
15886                    BUSDIN_EMIT-II),                        ;RESET PROC UCON
15887                    EM-CLK-IR[15-00],                       ;
15888            P3,     BUTA(CUA-TRACK),                        ;AND START CUR TRACKING AGAIN
15889            NEXT,   BUTD[SCOPE],                            ;NO ERROR:  "TEST701A" [+1, WORDS]
15890                    J/TEST701A                              ;   ERROR:  "SETIR624A" [-18, WORDS]
      (5224)  DCS[0,00,0,1,0,1]  BM[0000,,00,00,,00,01,,000,,100,,,0,0,0,,0,,0,,,1,1001,,,0,,0000,0,,,11,001,,,101,110,011]
15891
15892
15893
15894
15895
15896
15897
15898    1,PAGE*****************************************************************************************************
15899
15900
15901    .TOC    *   TEST701-702:  LOAD/READ THE BA, FULL 16. BITS
15902
15903    !*************************************************************************************
15904    !*
15905    !*      TESTS:  701A - 702B                             UWORDS:  033 + 042
15906    !*
15907    !*      FUNCTIONS:
15908    !*
15909    !*      THIS NEXT SET OF SIX TESTS CHECKS THE "BA" REGISTER RELATED FUNCTIONS:
15910    !*              SPECIFICALLY,
15911    !*              1) LOAD/READ OF BA<15:00>, FROM BUS-A => BA => STATUS-MUX/PSA-PORT
15912    !*              2) BUT ON "BA<00>"
15913    !*              3) LOAD READ OF BA<7:16>, FROM BUS-B => BA => STATUS-MUX/SERVICE-PORT
15914    !*                      THE EFFECT OF PARTICULAR KT-UCON-FUNCTIONS/CONSOLE-16.-BIT-MODE
15915    !*                      IN THIS SITUATION IS ALSO EXAMINED FOR FUNCTIONALITY.
15916    !*
15917    !*************************************************************************************
15918
15919
15920
15921    ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
15922
15923    !TEST-701-A CHECKS THAT BA<15:00> CAN BE LOADED, AND READ BACK WITH (052525)
15924    5563:
15925    TEST701A:
15926            P0,     LOAD-ENUA(ETARGET402),                  ;SETUP FOR 0=ZERO COMPARE
```



```
15927                    LOAD-ERROR(TEST701A),                  ;ERROR DIRECTORY KEY
15928                    DCS-CTR(C12.),                         ;COMPARE AT TARGET
15929                    BUMP-VERIFY,                           ;COUNT
15930            NEXT,   J/MASK701A                             ;
      (5563)  DCS[1,00,1,0,0,1]  BM[0011,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,100,111,110]
15931
15932    5478:
15933    MASK701A:
15934            P3,     CSPD[04]_EMIT, EMIT/177777,            ;MASK TO READ ALL THE BITS
15935            NEXT,   GOTO-PAGE(7),                          ;XFER
15936                    J/EXPECT01A                            ;
      (5476)  DCS[0,00,0,0,0,0]  BM[1111,,10,11,,11,11,,111,,111,,,0,0,0,,0,,0,,,0,1011,,,1,,0000,0,,,11,100,,,001,110,110]
15937
15938    7166:  ;(FREE)
15939    EXPECT01A:
15940            P3,     CSPD[02]_EMIT, EMIT/052525,            ;EXPECTED DATA TO BE READ OUT OF PSA AFTER LOAD:
15941            NEXT,   J/LOADBA701A                           ;"0101 0101 0101 0101"
      (7166)  DCS[0,00,0,0,0,0]  BM[0101,,10,01,,01,01,,010,,101,,,0,0,0,,0,,0,,,0,1101,,,1,,0000,0,,,11,000,,,001,111,000]
15942
15943    7170:  ;(FREE)
15944    LOADBA701A:
15945            P1,     BA_ASPHI(C052525),                     ;LOAD BA<15:00> WITH PATTERN
15946            NEXT,   J/GOTEST701A                           ;
      (7170)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,11,01,,111,,000,,,0,0,0,,1,,0,,,0,0000,,,0,,0000,0,,,11,000,,,001,111,001]
15947
15948    7171:  ;(FREF)
15949    GOTEST701A:
15950            SETUP,  RETURN/TEST701B,                       ;GO TO SUBR THAT:
15951            NEXT,   CALL(PBATOD]                           ;  (PSA)-XOR-CSP(02)=>D, BUT(D-16-ZERO)
      (7171)  DCS[0,00,0,0,0,0]  BM[0101,,00,10,,10,10,,011,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,101,001]
15952
15953
15954
15955
15956
15957
15958    ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
15959
15960    !TEST-701-B CHECKS THAT BA<00> WAS LOADED WITH A (1)
15961    5523:
15962    TEST701B:
15963            P0,     LOAD-ENUA(ETARGET403),                  ;BIT<00> SET
15964                    LOAD-ERROR(TEST701B),                  ;ERROR DIRECTORY KEY
15965                    DCS-CTR(C3.),                          ;COMPARE AT TARGET
15966            NEXT,   J/GOBUT701B                            ;
      (5523)  DCS[1,00,1,0,0,0]  BM[1100,,00,11,,11,00,,000,,011,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,100,011,100]

15967
15968    5414:  ;(FREE)
15969    GOBUT701B:
15970            SETUP,  RETURN/TEST701C,                       ;RETURN TO START OF NEXT SUBTEST
15971            NEXT,   GOTO-PAGE(7),                          ;BUT TABLE
15972                    J/BUTBA00                              ;BA<00>=M IN BIT<00>
```

```
(5434)  DCS[0,00,0,0,0,0]  BM[0110,,00,11,,11,11,,000,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,101,001]




        ; - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        ;TEST-701-C CHECKS THAT BA<15:00> CAN BE LOADED, AND READ BACK WITH (125252)
        6770:
        TEST701C:
                P0,     LOAD-ENUA(ZTARGET402),                  ;SETUP FOR D=ZERO COMPARE
                        LOAD-ERROR(TEST701C),                   ;ERROR DIRECTORY KEY
                        DCS-CTR(C11,),                          ;COMPARE AT TARGET
                NEXT,   J/EXPECT701C                            ;
(6770)  DCS[1,00,1,0,0,0]  BM[0100,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,100,101,101]

        6455:  ;(FREE)
        EXPECT701C:
                P3,     CSPD[02]_EMIT, EMIT/125252,             ;EXPECTED DATA TO BE READ OUT OF PBA AFTER LOAD:
                NEXT,   J/LOADBA701C                            ;"1010 1010 1010 1010"
(6455)  DCS[0,00,0,0,0,0]  BM[1010,,10,10,,10,10,,101,,010,,,0,0,0,,0,,0,,,0,1101,,,1,,0000,0,,,11,000,,,100,101,110]

        6456:  ;(FREE)
        LOADBA701C:
                P1,     BA_ASPHI(C125252),                      ;LOAD BA<15:00> WITH PATTERN
                NEXT,   J/GOTEST701C                            ;
(6456)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,11,01,,110,,000,,,0,0,0,,1,,0,,,0,0000,,,0,,0000,0,,,11,000,,,100,101,111]

        6457:  ;(FREE)
        GOTEST701C:
                SETUP,  RETURN/TEST701D,                        ;GO TO SUBR THAT:
                NEXT,   CALL[PBATOD]                            ;  (PBA)-XOR-CSP(02)->D, BUT(D-IS-ZERO)
(6457)  DCS[0,00,0,0,0,0]  BM[0101,,00,10,,10,01,,010,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,101,001]






        ; - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        ;TEST-701-D CHECKS THAT BA<00> WAS LOADED WITH A (0)
        5512:
        TEST701D:
                P0,     LOAD-ENUA(ZTARGET402),                  ;BIT<00> CLEAR
                        LOAD-ERROR(TEST701D),                   ;ERROR DIRECTORY KEY
                        DCS-CTR(C3,),                           ;COMPARE AT TARGET
                NEXT,   J/GOBUT701D                             ;
(5512)  DCS[1,00,1,0,0,0]  BM[1100,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,100,011,101]

        5435:  ;(FREE)
        GOBUT701D:
                SETUP,  RETURN/SCOPE701,                        ;RETURN TO SCOPE LOOP TEST WORD
```



```
                NEXT,   GOTO-PAGE(7),                           ;BUT TABLE
                        J/BUTBA00                               ;BA<00>H IN BIT<00>
(5435)  DCS[0,00,0,0,0,0]  BM[0101,,00,10,,00,11,,110,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,101,001]




        5436:  ;(FREE)
        SCOPE701:
                P0,     BUMP-VERIFY,                            ;COUNT
                NEXT,   BUTD[SCOPE],                            ;NO ERROR:  "TEST702A" (+1, WORDS)
                        J/TEST702A                              ;   ERROR:  "MASK701A"  (-12, WORDS)
(5436)  DCS[0,00,0,1,0,1]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,100,111,111]




        ; - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        ;TEST-702-A CHECKS THAT BA<17:16> CAN BE LOADED, AND READ BACK WITH "01"
        ;  WHEN IN 18, BIT CONSOLE MODE AND KT-NO-RELOCATE MODE
        5477:
        TEST702A:
                P0,     LOAD-ENUA(ZTARGET402),                  ;SETUP FOR D=ZERO COMPARE
                        LOAD-ERROR(TEST702A),                   ;ERROR DIRECTORY KEY
                        DCS-CTR(C13,),                          ;COMPARE AT TARGET
                NEXT,   J/MASK702A                              ;
(5477)  DCS[1,00,1,0,0,0]  BM[0010,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,100,111,100]

        5474:
        MASK702A:
                P3,     CSPD[04]_EMIT, EMIT/001400,             ;MASK TO READ ONLY BITS<9:8>
                NEXT,   J/EXPECT02A                             ;
(5474)  DCS[0,00,0,0,0,0]  BM[0000,,10,00,,11,00,,000,,000,,,0,0,0,,0,,0,,,0,1011,,,1,,0000,0,,,11,000,,,100,011,111]

        5437:  ;(FREE)
        EXPECT02A:
                P3,     CSPD[02]_EMIT, EMIT/000400,             ;EXPECTED DATA TO BE READ OUT OF SERVICE<9:8> AFTER LOAD:
                NEXT,   J/SETLED702A                            ;"0000 0001 0000 0000"
(5437)  DCS[0,00,0,0,0,0]  BM[0000,,10,00,,01,00,,000,,000,,,0,0,0,,0,,0,,,0,1101,,,1,,0000,0,,,11,000,,,100,100,001]

        5441:  ;(FREE)
        SETLED702A:
                P3,     SET-CONSOLE-LED,                        ;ENTER INTO 18, BIT MODE FOR PBA<17:16> READ
                NEXT,   J/SETKT702A                             ;
(5441)  DCS[0,00,0,0,0,0]  BM[0100,,00,00,,00,00,,100,,001,,,0,0,0,,0,,0,,,1,1011,,,0,,0000,0,,,11,000,,,100,100,010]

        5442:  ;(FREE)
        SETKT702A:
                P0,     BUMP-VERIFY,                            ;COUNT
                        BUSDIN_SERVICE-[I],                     ;READ SERVICE PORT BITS<9:8>
                        KT-NO-RELOCATE-[I],                     ;SETUP KT FOR BA<17:16> LOADABILITY
                NEXT,   J/LOADBA702A                            ;
```

```
(5442)  DCS[0,00,0,0,0,1]  BM[0101..01,00..00,00..000..010...0,0,0..0..0...1,1001...0..0000,0...11,000...100,100,011]

    5443:  ;(FREE)
    LOADBA702A:
            P1,     BA_BSPHI(C052525),                      ;LOAD BA<17:16> WITH PATTERN "01"
            NEXT,   J/GOTEST702A                            ;
(5443)  DCS[0,00,0,0,0,0]  BM[0000..01,11..00,00..111..000...0,0,0..1..0...0,0000...0..0000,0...11,000...100,100,100]

    5444:  ;(FREE)
    GOTEST702A:
            SETUP,  RETURN/TEST702B,                        ;GO TO SUBR THAT:
            NEXT,   CALL[GETPROCDAT]                        ;  (SERVICE<9:8>)-XOR-CSP(02)->D, BUT(D-IS-ZERO)
(5444)  DCS[0,00,0,0,0,0]  BM[0110..00,11..11,10..001..111...0,0,0..0..0...0,0000...0..0000,0...11,100...011,111,011]







    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

    ;TEST-702-B CHECKS THAT BA<17:16> CAN BE LOADED, AND READ BACK WITH "10"
    ;  WHEN IN 18, BIT CONSOLE MODE AND KT-NO-RELOCATE MODE
    6761:
    TEST702B:
            P0,     LOAD-ENUA(ZTARGET402),                  ;SETUP FOR D=ZERO COMPARE
                    LOAD-ERROR(TEST702B),                   ;ERROR DIRECTORY KEY
                    DCS-CTR(C11,),                          ;COMPARE AT TARGET
            NEXT,   J/EXPEC702B                             ;
(6761)  DCS[1,00,1,0,0,0]  BM[0100..00,11..11,00..000..010...0,0,0..0..0...0,0000...0..0000,0...11,000...100,110,000]

    6460:  ;(FREE)
    EXPEC702B:
            P3,     CSPD(02)_EMIT, EMIT/001000,             ;EXPECTED DATA TO BE READ OUT OF SERVICE<9:8> AFTER LOAD:
            NEXT,   J/SETKT702B                             ;"0000 0010 0000 0000"
(6460)  DCS[0,00,0,0,0,0]  BM[0000..10,00..10,00..000..000...0,0,0..0..0...0,1101...1..0000,0...11,000...100,110,001]

    6461:  ;(FREE)
    SETKT702B:
            P0,     BUSDIN_SERVICE-[I],                     ;READ SERVICE PORT BITS<9:8>
                    KT-NO-RELOCATE-[Y],                     ;SETUP KT FOR BA<17:16> LOADABILITY
            NEXT,   J/LOADBA702B                            ;
(6461)  DCS[0,00,0,0,0,0]  BM[0101..01,00..00,00..000..010...0,0,0..0..0...1,1001...0..0000,0...11,000...100,110,010]

    6462:  ;(FREE)
    LOADBA702B:
            P0,     BUMP-VERIFY,                            ;COUNT
            P1,     BA_BSPHI(C125252),                      ;LOAD BA<17:16> WITH PATTERN "10"
            NEXT,   J/GOTEST702B                            ;
(6462)  DCS[0,00,0,0,0,1]  BM[0000..01,11..00,00..110..000...0,0,0..1..0...0,0000...0..0000,0...11,000...100,110,011]

    6463:  ;(FREE)
```



```
    GOTEST702B:
            SETUP,  RETURN/SCOPE702,                        ;GO TO SUBR THAT:
            NEXT,   CALL[GETPROCDAT]                        ;  (SERVICE<9:8>)-XOR-CSP(02)->D, BUT(D-IS-ZERO)
(6463)  DCS[0,00,0,0,0,0]  BM[0101..00,10..00,11..011..111...0,0,0..0..0...0,0000...0..0000,0...11,100...011,111,011]



    5433:  ;(FREE)
    SCOPE702:
            P3,     CSPD(04)_EMIT, EMIT/177777,             ;RESET MASK FOR SUBSEQUENT TESTS
            NEXT,   BUTD(SCOPE),                            ;NO ERROR:  "TEST710A"  (+1, WORDS)
                    J/TEST710A                              ;   ERROR:  "MASK702A"  (-11, WORDS)
(5433)  DCS[0,00,0,1,0,0]  BM[1111..10,11..11,11..111..111...0,0,0..0..0...0,1011...1..0000,0...11,000...100,111,101]





    ;.PAGE*************************************************************************************************


    .TOC    *    TEST710-722:  BUS FUNCTION DECODE, BUS ERROR CONDITIONS

    ;*****************************************************************************************
    ;*
    ;*      TESTS:  710A - 722C                             UWORDS:  133 + 274
    ;*
    ;*      FUNCTIONS:
    ;*
    ;*      WE GET SNEAKY HERE AND TEST THE FULL BUS DECODE/STATUS MUX LOGIC, AND THE BUS  ERROR
    ;*      CONDITIONS (ODD ADDRESS, INTERNAL ADDRESS, ETC) ALMOST WITHOUT EVER GOING OUT ON THE
    ;*      BUS (WE DO ONCE, FOR SSYN TIMEOUT).
    ;*
    ;*****************************************************************************************



    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

    ;TESTING "DATO", "ODD-ADDRESS" JAMUPP, 18./16, BIT IO-PAGE DECODE

    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

    ;TEST-710-A FIRST ATTEMPTS TO DO A BUS "DATO" FUNCTION, TRYING TO FORCE AN "ODD ADDRESS"
    ;  ABORT/JAMUPP
    5475:

    TEST710A:
            P0,     LOAD-ENUA(4777),                        ;JAMUPP ADDRESS
                    LOAD-ERROR(TEST710A),                   ;ERROR DIRECTORY KEY
                    DCS-CTR(C6.),                           ;COMPARE JUST AFTER BUS CYCLE UWORD, AT JAM
            NEXT,   J/LOADRET710A                           ;
(5475)  DCS[1,00,1,0,0,0]  BM[1001..00,10..01,11..111..111...0,0,0..0..0...0,0000...0..0000,0...11,000...100,111,010]
```

```
        5472:
        LOADRET710A:
                P3,     CSPD[00]_EMIT, RETURN/TEST710B,        ;RETURN AFTER SUCCESSFUL JAM TO NEXT TEST
                NEXT,   GOTO-PAGE(4),                          ;XFER
                        J/LOADADR710A                          ;
   (5472)  DCS[0,00,0,0,0,0]  BM[0110,,10,11,,01,00,,100,,100,,,0,0,0,,0,,0,,,0,1111,,,1,,0000,0,,,11,100,,,011,000,111]

        4307:  ;(FREE)
        LOADADR710A:
                P3,     CSPD[16]_EMIT, EMIT/160001,            ;"DIAGNOSTIC" UNIBUS I/O PAGE ADDRESS; ODD BYTE
                NEXT,   J/SETJAM710A                           ;
   (4307)  DCS[0,00,0,0,0,0]  BM[1110,,10,00,,00,00,,000,,001,,,0,0,0,,0,,0,,,0,0001,,,1,,0000,0,,,11,000,,,011,001,010]

        4312:  ;(FREE)
        SETJAM710A:
                P2-T,   SR_CSPD(D16),                          ;SET B[illegible]<00>=(1) FOR JAMUPP EXPECTED
                NEXT,   J/BUSFCN710A                           ;
   (4312)  DCS[0,00,0,0,0,0]  BM[1010,,10,00,,00,00,,000,,000,,,0,0,1,,0,,0,,,0,0001,,,0,,0000,0,,,11,000,,,011,001,011]

        4313:  ;(FREE)
        BUSFCN710A:
                P1,     BA_SR, BSPHI(C177777),                 ;SET BA<17:00>=(760001), ALTERED TO(160001)
                                                               ;  SINCE WE'VE LEFT 18. BIT MODE (IN LOADING BA, THAT IS)
                P2-T,   D_ZERO,                                ;
                P3,     DATO,                                  ;DO A BUS "DATO", SHOULD GET ODD ADDRESS ABORTED
                NEXT,   J/NEXT710A                             ;GO DELAY
   (4313)  DCS[0,00,0,0,0,0]  BM[0011,,01,11,,00,00,,101,,000,,,0,1,0,,1,,0,,,1,0010,,,0,,0000,0,,,11,000,,,100,100,101]

        ;** AT THIS POINT JAMUPP SHOULD OCCUR **
        ;** CLASSIC FLOW (4777) -> (4757) -> (7XXX) -> (4XXX), AND THEN WE'RE BACK HERE **
        ;**  RETURN TO ADDRESS LEFT IN CSP(00) **

        ;*** END UP HERE IF NO JAMUPP ***
        4445:
        NEXT710A:
                SETUP,  RETURN/TEST710A,                       ;FORCE A SCOPE LOOP ON THIS TEST, BUT FIRST
                NEXT,   GOTO-PAGE(7),                          ;  MUST DELAY A FEW MICROWORDS FOR BUS
                        J/BUTD-IS-ZERO                         ;  ERROR TO TAKE EFFECT  (IGNORE "BUT" OUTCOME HERE)
   (4445)  DCS[0,00,0,0,0,0]  BM[0101,,00,10,,01,11,,101,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,100,001]


        ;*** END UP HERE IF JAMUPP ***


        ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        ;TEST-710-B CHECKS THAT THE RIGHT JAM (ODD ADDRESS) IS INDICATED IN THE JAMREG:
        ;
        ;       BIT:      15   14   13   12   11   10   09   08   07   06   05   04   03   02   01   00
        ;       FCN:      ODD   0  SSYN YEL  RED  WCS  PON  MEM  SSYN CACH ILL  NOT  RED  ODD  WCS  UBRK
        ;                 ADR      TIME ZON  ZON  PAR  DIS  PAR  TIME ERR  ADR  ABT  ZON  ADR  PAR  TRAP
        ;       (101004)  1    0    0    0    0    0    1    0    0    0    0    0    0    1    0    0
```



```
        6644:
        TEST710B:
                P0,     LOAD-ENUA(ZTARGET402),                 ;SETUP FOR D=ZERO COMPARE
                        LOAD-ERROR(TEST710B),                  ;ERROR DIRECTORY KEY
                        DCS-CTR(C10.),                         ;COMPARE AT TARGET
                NEXT,   J/GOTEST710B                           ;
   (6644)  DCS[1,00,1,0,0,0]  BM[0101,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,100,110,100]

        6464:  ;(FREE)
        GOTEST710B:
                SETUP,  RETURN/TEST710C,                       ;GO TO SUBR WHICH:
                NEXT,   CALL[ODDJAMTOD]                        ;  (JAMR[illegible]G)-XOR-(101004) -> D, BUT(D=ZERO)  (ODD-ADDRESS)
   (6464)  DCS[0,00,0,0,0,0]  BM[0110,,00,11,,01,00,,101,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,101,100]



        ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        ;TEST-710-C CHECKS THAT THE RIGHT BUS FUNCTION DECODE / PBA<17:16> ARE INDICATED IN SERVICE REG:
        ;  NOTE PBA<17:16> READ "TRUE" (OUT OF BA) WHEN IN 18. BIT CONSOLE MODE
        ;
        ;       BIT:      15   14   13   12   11    10   09   08   07   06   05   04   03   02   01   00
        ;       FCN:      DATI  BG   0   NPR DATOB DATO PBA  PBA  HIB  LOB  TAG  CON  FLT  POW  CACH YEL
        ;                      SERV      TIME            17   16   ERR  ERR  ERR  SERV SERV FAIL  ERR ZON
        ;       (002340)  0    0    0    0    0     1    0    0    1    1    1    0    0    0    0    0

        6645:
        TEST710C:
                P0,     LOAD-ENUA(ZTARGET402),                 ;SETUP FOR D=ZERO COMPARE
                        LOAD-ERROR(TEST710C),                  ;ERROR DIRECTORY KEY
                        DCS-CTR(C10.),                         ;COMPARE AT TARGET
                NEXT,   J/EXPECT710C                           ;
   (6645)  DCS[1,00,1,0,0,0]  BM[0101,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,100,110,101]

        6465:  ;(FREE)
        EXPECT710C:
                P3,     CSPD[02]_EMIT, EMIT/002340,            ;DATO(1)N SET, PBA<17:16>="00" IN 18. BIT CONSOLE MODE
                NEXT,   J/GOTEST710C                           ;  IN SERVICE REG
   (6465)  DCS[0,00,0,0,0,0]  BM[0000,,10,01,,00,11,,100,,000,,,0,0,0,,0,,0,,,0,1101,,,1,,0000,0,,,11,000,,,100,110,110]

        6466:  ;(FREE)
        GOTEST710C:
                SETUP,  RETURN/TEST710D,                       ;GO TO SUBR WHICH:
                NEXT,   CALL[SERVICETOD]                       ;  (SERVICE)-XOR-CSP(02) -> D, BUT(D-IS-ZERO)
   (6466)  DCS[0,00,0,0,0,0]  BM[0110,,00,10,,11,11,,100,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,101,000]



        ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        ;TEST-710-D CHECKS THAT PBA<17:16> ARE INDICATED IN SERVICE REG:
```

```
        ; NOTE PBA<17:16> READ AS BA<15:13>="111" WHEN IN 16. BIT MODE
        ;
        ;       BIT:       15   14    13   12   11    10   09  08  07  06  05  04  03  02  01  00
        ;       FCN:     DATI  BG     0  NPR DATOB DATO PBA PBA NXM LOB TAG CON FLT POW CACK YEL
        ;                     SERV       TIME             17  16 ERR ERR ERR SERV SERV FAIL ERR ZON
        ;       (003740)   0    0     0    0     0    1    1   1   1   1   1   0   0   0   0   0

        6574:
        TEST710D:
                P0,     LOAD-ENUA(ZTARGET402),                  ;SETUP FOR D=ZERO COMPARE
                        LOAD-ERROR(TEST710D),                   ;ERROR DIRECTORY KEY
                        DCS-CTR(C11.),                          ;COMPARE AT TARGET
                NEXT,   J/EXPECT710D                            ;
   (6574)  DCS[1,00,1,0,0,0]  BM[0100,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,100,110,111]

        6467:  ;(FREE)
        EXPECT710D:
                P0,     BUMP-VERIFY,                            ;COUNT
                P3,     CSPD[02]_ENIT, ENIT/003740,             ;DATO(1)N SET, PBA<17:16>="11" IN 16. BIT MODE
                NEXT,   J/CLEAR710D                             ;  IN SERVICE REG
   (6467)  DCS[0,00,0,0,0,1]  BM[0000,,10,01,,11,11,,100,,000,,,0,0,0,,0,,0,,,0,1101,,,1,,0000,0,,,11,000,,,100,111,000]

        6470:  ;(FREE)
        CLEAR710D:
                P3,     CLR-JAM-ERRORS-[T],                     ;CLEAR OUT JAM-REG, FOR USE IN NEXT TEST
                        CLEAR-CONSOLE-LED,                      ;BACK TO 16. BIT MODE
                NEXT,   J/GOTEST710D                            ;
   (6470)  DCS[0,00,0,0,0,0]  BM[0100,,00,00,,10,00,,010,,001,,,0,0,0,,0,,0,,,1,1011,,,0,,0000,0,,,11,000,,,100,111,001]

        6471:  ;(FREE)
        GOTEST710D:
                SETUP,  RETURN/TEST710E,                        ;GO TO SUBR WHICH:
                NEXT,   CALL[SERVICETOD]                        ;  (SERVICE)-XOR-CSP(02) => D, BUT(D-IS-ZERO)
   (6471)  DCS[0,00,0,0,0,0]  BM[0110,,00,10,,11,11,,101,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,101,000]



        ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        ;TEST-710-E CHECKS THAT THE CLEAR-JAM-ERRORS CLEARED THE JAM REG TO (001000)
        6575:
        TEST710E:
                P0,     LOAD-ENUA(ZTARGET402),                  ;SETUP FOR D=ZERO COMPARE
                        LOAD-ERROR(TEST710E),                   ;ERROR DIRECTORY KEY
                        DCS-CTR(C10.),                          ;COMPARE AT TARGET
                NEXT,   J/GOTEST710E                            ;
   (6575)  DCS[1,00,1,0,0,0]  BM[0101,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,100,111,010]

        6472:  ;(FREE)
        GOTEST710F:
                P0,     BUMP-VERIFY,                            ;COUNT
                SETUP,  RETURN/SCOPE710,                        ;GO TO SUBR WHICH:
                                                                ;  CSP(02) <= (001000)
```



```
                NEXT,   CALL[CLRJAMTOD]                         ;  (JAMREG)-XOR-CSP(02) => D, BUT(D-IS-ZERO)
   (6472)  DCS[0,00,0,0,0,1]  BM[0101,,00,10,,01,00,,101,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,101,101]



        5445:  ;(FREE)
        SCOPE710:
                P2-U,   IF_ENIT, ENIT/123456,                   ;A "CMP-BYTE" INSTR, FOR NEXT TEST SERIES
                NEXT,   BUTD[SCOPE],                            ;NO ERROR:  "TEST711A"  (+1. WORDS)
                        J/TEST711A                              ;   ERROR:  "LOADRET710A"  (-17. WORDS)
   (5445)  DCS[0,00,0,1,0,0]  BM[1010,,00,01,,11,00,,101,,110,,,0,0,0,,0,,0,,,1,1010,,,0,,0000,0,,,11,000,,,100,111,011]






        ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        ;TESTING "DATOB+BYTE", "SSYN TIMEOUT" JAMUPP

        ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        ;TEST-711-A DOES A BUS "DATOB+BYTE" FUNCTION, TRYING TO FORCE AN "SSYN TIMEOUT" ABORT/JAMUPP
        5473:
        TEST711A:
                P0,     LOAD-ENUA(4777),                        ;JAMUPP ADDRESS
                        LOAD-ERROR(TEST711A),                   ;ERROR DIRECTORY KEY
                        DCS-CTR(C4.),                           ;COMPARE JUST AFTER BUS CYCLE UWORD, AT JAM
                NEXT,   J/LOADRET711A                           ;
   (5473)  DCS[1,00,1,0,0,0]  BM[1011,,00,10,,01,11,,111,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,100,111,000]

        5470:
        LOADRET711A:
                P0,     BUMP-VERIFY,                            ;COUNT
                P3,     CSPD[00]_ENIT, RETURN/TEST711B,         ;RETURN AFTER SUCCESSFUL JAM TO NEXT TEST
                NEXT,   GOTO-PAGE(4),                           ;XFER
                NEXT,   J/BUSFCN711A                            ;
   (5470)  DCS[0,00,0,0,0,1]  BM[0110,,10,10,,10,11,,100,,100,,,0,0,0,,0,,0,,,0,1111,,,1,,0000,0,,,11,100,,,011,001,001]

        4311:  ;(FREE)
        BUSFCN711A:
                ;NOTE:  BA=(160001) FROM PREVIOUS TEST
                P2-T,   SR_CSPB(B16),                           ;SET BIT<00>=(1) FOR JAMUPP EXPECTED
                P3,     DATOB,                                  ;DO A BUS "DATOB+BYTE", SHOULD GET SSYN TIMEOUT ABORTED
                                                                ;  SINCE WERE USING THE "DIAGNOSTIC" UNIBUS TIMEOUT ADDR

                NEXT,   J/NEXT711A                              ;GO DELAY
   (4311)  DCS[0,00,0,0,0,0]  BM[1010,,11,01,,00,00,,000,,000,,,0,0,1,,0,,0,,,1,0101,,,0,,0000,0,,,11,000,,,100,100,110]

        ;** AT THIS POINT JAMUPP SHOULD OCCUR **
        ;** CLASSIC FLOW (4777) => (4757) => (7XXX) => (4XXX), AND THEN WE'RE BACK HERE **
        ;**  RETURN TO ADDRESS LEFT IN CSP(00) **
```

```
    ;*** END UP HERE IF NO JAMUPP ***
    4446:
    NEXT711A:
            SETUP,  RETURN/TEST711A,                    ;FORCE A SCOPE LOOP ON THIS TEST, BUT FIRST
            NEXT,   GOTO-PAGE(7),                       ;  MUST DELAY A FEW MICROWORDS FOR BUS
                    J/BUTD-IS-ZERO                      ;  ERROR TO TAKE EFFECT  (IGNORE "BUT" OUTCOME HERE)
 (4446)  DCS[0,00,0,0,0,0]  BM[0101,,00,10,,01,11,,011,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,100,001]


    ;*** END UP HERE IF JAMUPP ***


    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

    ;TEST-711-B CHECKS THAT THE RIGHT JAM (SSYN TIMEOUT) IS INDICATED IN THE JAMREG;
    ;
    ;       BIT:     15   14   13   12   11   10   09   08   07   06   05   04   03   02   01   00
    ;       FCN:     ODD   0  SSYN  YEL  RED  WCS  POW  MEM SSYN CACH ILL  NOT  RED  ODD  WCS  UBRK
    ;                ADR       TIME ZON  ZON  PAR  DIS  PAR TIME  ERR ADR  ABT  ZON  ADR  PAR  TRAP
    ;       (021200)  0    0    1    0    0    0    1    0    1    0    0    0    0    0    0    0

    6534:
    TEST711B:
            P0,     LOAD-ENUA(ZTARGET402),              ;SETUP FOR D=ZERO COMPARE
                    LOAD-ERROR(TEST711B),               ;ERROR DIRECTORY KEY
                    DCS-CTR(C10,),                      ;COMPARE AT TARGET
            NEXT,   J/EXPECT711B                        ;
 (6534)  DCS[1,00,1,0,0,0]  BM[0101,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,100,111,011]

    6473:  ;(FREE)
    EXPECT711B:
            P0,     BUMP-VERIFY,                        ;COUNT
            P3,     CSPD[02]_EMIT, EMIT/021200,         ;"SSYN TIMEOUT(1)H" SET
            NEXT,   J/GOTEST711B                        ;  IN JAMREG
 (6473)  DCS[0,00,0,0,0,1]  BM[0010,,10,00,,10,10,,000,,000,,,0,0,0,,0,,0,,,0,1101,,,1,,0000,0,,,11,000,,,100,111,100]

    6474:  ;(FREE)
    GOTEST711B:
            SETUP,  RETURN/TEST711C,                    ;GO TO SUBR WHICH:
            NEXT,   CALL[JAMTOD]                        ;  (JAMREG)-XOR-CSP(02) => D, BUT(D-IS-ZERO)
 (6474)  DCS[0,00,0,0,0,0]  BM[0110,,00,10,,10,11,,000,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,101,110]



    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

    ;TEST-711-C CHECKS THAT THE RIGHT BUS FUNCTION DECODE / PBA<17:16> ARE INDICATED IN SERVICE REG;
    ;  BACK IN 16. BIT MODE, SINCE I-D PAGE, PBA<17:16> READ AS "11"
    ;
    ;       BIT:      15   14   13   12   11    10    09   08   07   06   05   04   03   02   01   00
    ;       FCN:     DATI  BG    0  NPR DATOB DATO PBA  PBA  RIB  LOB  TAG  CON  FLT  POW  CACH YEL
```



```
    ;                    SERV      TIME                 17   16  ERR  ERR  ERR SERV SERV FAIL  ERR ZON
    ;       (005740)  0    0    0    0    1    0    1    1    1    1    1    0    0    0    0    0

    6530:
    TEST711C:
            P0,     LOAD-ENUA(ZTARGET402),              ;SETUP FOR D=ZERO COMPARE
                    LOAD-ERROR(TEST711C),               ;ERROR DIRECTORY KEY
                    DCS-CTR(C11,),                      ;COMPARE AT TARGET
            NEXT,   J/EXPECT711C                        ;
 (6530)  DCS[1,00,1,0,0,0]  BM[0100,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,100,111,101]

    6475:  ;(FREE)
    EXPECT711C:
            P3,     CSPD[02]_EMIT, EMIT/005740,         ;DATOB(1)H SET, PBA<17:16>="11"
            NEXT,   J/GOTEST711C                        ;  IN SERVICE REG
 (6475)  DCS[0,00,0,0,0,0]  BM[0000,,10,10,,11,11,,100,,000,,,0,0,0,,0,,0,,,0,1101,,,1,,0000,0,,,11,000,,,100,111,110]

    6476:  ;(FREE)
    GOTEST711C:
            SETUP,  RETURN/TEST711D,                    ;GO TO SUBR WHICH:
                                                        ;  CLR-JAM-ERRORS-[I], FOR NEXT TEST
            NEXT,   CALL[CJESERVICETOD]                 ;  (SERVICE)-XOR-CSP(02) => D, BUT(D-IS-ZERO)
 (6476)  DCS[0,00,0,0,0,0]  BM[0110,,00,10,,10,10,,110,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,100,111]



    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

    ;TEST-711-D CHECKS THAT THE CLEAR-JAM-ERRORS CLEARED THE JAM REG TO (001000)
    6526:
    TEST711D:
            P0,     LOAD-ENUA(ZTARGET402),              ;SETUP FOR D=ZERO COMPARE
                    LOAD-ERROR(TEST711D),               ;ERROR DIRECTORY KEY
                    DCS-CTR(C10,),                      ;COMPARE AT TARGET
                    BUMP-VERIFY,                        ;COUNT
            NEXT,   J/GOTEST711D                        ;
 (6526)  DCS[1,00,1,0,0,1]  BM[0101,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,100,111,111]

    6477:  ;(FREE)
    GOTEST711D:
            SETUP,  RETURN/SCOPE711,                    ;GO TO SUBR WHICH:
                                                        ;  CSP(02) <= (001000)
            NEXT,   CALL[CLRJAMTOD]                     ;  (JAMREG)-XOR-CSP(02) => D, BUT(D-IS-ZERO)
 (6477)  DCS[0,00,0,0,0,0]  BM[0101,,00,10,,01,00,,110,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,101,101]



    5146:  ;(FREE)
    SCOPE711:
            NEXT,   BUTD[SCOPE],                        ;NO ERROR:  "TEST712A"  (+1, WORDS)
                    J/TEST712A                          ;   ERROR:  "LOADRET711A"  (-11, WORDS)
 (5146)  DCS[0,00,0,1,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,100,111,001]
```

```
	;- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

	;TESTING 'DATI', 'INTERNAL ADDRESS' JAMUPP

	; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

	;TEST-712-A DOES A BUS "DATI" FUNCTION TO AN INTERNAL ADDRESS, TRYING TO FORCE AN "INTERNAL ADDRESS" ABORT/JAMUPP
	5471:
	TEST712A:
		P0,	LOAD-ENUA(4777),			;JAMUPP ADDRESS
			LOAD-ERROR(TEST712A),			;ERROR DIRECTORY KEY
			DCS-CTR(C5,),				;COMPARE JUST AFTER BUS CYCLE UWORD, AT JAM
		NEXT,	J/LOADRET712A				;
(5471)	DCS[1,00,1,0,0,0]  BM[1010,,00,10,,01,11,,111,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,100,110,110]

	5466:
	LOADRET712A:
		P0,	BUMP-VERIFY,				;COUNT
		P3,	CSPD[00]_EMIT, RETURN/TEST712B,		;RETURN AFTER SUCCESSFUL JAM TO NEXT TEST
		NEXT,	GOTO-PAGE(4),				;XFER
			J/GENADR712A				;
(5466)	DCS[0,00,0,0,0,1]  BM[0101,,10,10,,10,00,,101,,100,,,0,0,0,,0,,0,,,0,1111,,,1,,0000,0,,,11,100,,,011,001,100]

	4314:	;(FREE)
	GENADR712A:
		P2-T,	D_NOT-ASPHI(C000001),			;MAKE D=(177776), INTERNAL ADDRESS OF PSW
		P3,	ASPLO[11]_D,				;  AND SAVE IT
		NEXT,	J/BUSFCN712A				;
(4314)	DCS[0,00,0,0,0,0]  BM[0000,,00,00,,11,01,,000,,000,,,0,1,0,,0,,0,,,0,0000,,,0,,0001,0,,,11,000,,,011,001,110]

	4316:	;(FREE)
	BUSFCN712A:
		P1,	BA_ASPLO(R11),				;BA=(177776)
		P2-T,	SR_BSPHI(C000001),			;SET BIT<00>=(1) FOR JAMUPP EXPECTED
		P3,	DATI,					;DO A BUS "DATI", SHOULD GET INTERNAL ADDR ABORTED
		NEXT,	J/NEXT712A				;GO DELAY
(4316)	DCS[0,00,0,0,0,0]  BM[1010,,01,11,,10,01,,000,,000,,,0,0,1,,1,,0,,,1,0110,,,0,,0000,0,,,11,000,,,100,100,111]

	;** AT THIS POINT JAMUPP SHOULD OCCUR **
	;** CLASSIC FLOW (4777) -> (4757) -> (7XXX) -> (4XXX), AND THEN WE'RE BACK HERE **
	;**  RETURN TO ADDRESS LEFT IN CSP(00) **

	;*** END UP HERE IF NO JAMUPP ***
	4447:
	NEXT712A:
		SETUP,	RETURN/TEST712A,			;FORCE A SCOPE LOOP ON THIS TEST, BUT FIRST
		NEXT,	GOTO-PAGE(7),				;  MUST DELAY A FEW MICROWORDS FOR BUS
```



```
			J/BUTD-IS-ZERO				;  ERROR TO TAKE EFFECT  (IGNORE "BUT" OUTCOME HERE)
(4447)	DCS[0,00,0,0,0,0]  BM[0101,,00,10,,01,11,,001,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,100,001]


	;*** END UP HERE IF JAMUPP ***


	; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

	;TEST-712-B CHECKS THAT THE RIGHT JAM (INTERNAL ADDRESS) IS INDICATED IN THE JAMREG:
	;  FOR THIS JAM, THE JAMREG SHOULD REMAIN CLEAR
	;
	;	BIT:	  15	14   13   12   11   10   09   08   07   06   05   04   03   02   01   00
	;	FCN:	 ODD	 0 ASYN YEL  RED  NCS  PON  MEM  ASYN CACH ILL  NGT  RED  ODD  NCS  UBRK
	;		 ADR	   TIME ZON  ZON  PAR  DIS  PAR  TIME ERR  ADR  ABT  ZON  ADR  PAR  TRAP
	;	(001000)  0	 0    0    0    0    0    1    0    0    0    0    0    0    0    0    0

	5505:
	TEST712B:
		P0,	LOAD-ENUA(ZTARGET402),			;SETUP FOR D=ZERO COMPARE
			LOAD-ERROR(TEST712B),			;ERROR DIRECTORY KEY
			DCS-CTR(C10,),				;COMPARE AT TARGET
		NEXT,	J/GOTEST712B				;
(5505)	DCS[1,00,1,0,0,0]  BM[0101,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,100,101,000]

	5450:	;(FREE)
	GOTEST712B:
		SETUP,	RETURN/TEST712C,			;GO TO SUBR WHICH:
								;  (001000) -> CSP(02)
		NEXT,	CALL[CLRJAMTOD]				;  (JAMREG)-XOR-CSP(02) -> D, BUT(D-IS-ZERO)
(5450)	DCS[0,00,0,0,0,0]  BM[0110,,00,10,,10,10,,100,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,101,101]



	; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

	;TEST-712-C CHECKS THAT THE "OTHER-JAM-H" SIGNAL IS LOW, INDICATING AN INTERNAL ADDRESS JAM
	6524:
	TEST712C:
		P0,	LOAD-ENUA(ZTARGET401),			;BIT<01> CLEAR
			LOAD-ERROR(TEST712C),			;ERROR DIRECTORY KEY
			DCS-CTR(C3,),				;COMPARE AT TARGET
		NEXT,	J/GOBUT712C				;
(6524)	DCS[1,00,1,0,0,0]  BM[1100,,00,11,,11,00,,000,,001,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,101,000,000]

	6500:	;(FREE)

	GOBUT712C:
		SETUP,	RETURN/TEST712D,			;RETURN TO START OF NEXT SUBTEST
		NEXT,	GOTO-PAGE(7),				;BUT TABLE
			J/BUTOTHERJAM				;"OTHER JAM H" IN BIT<01>
(6500)	DCS[0,00,0,0,0,0]  BM[0110,,00,10,,10,10,,011,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,101,010]
```

```
16541
16542
16543
16544    ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
16545
16546    !TEST-712-D CHECKS THAT THE RIGHT BUS FUNCTION DECODE / PSA<17:16> ARE INDICATED IN SERVICE REG!
16547    !  BACK IN 16. BIT MODE, SINCE I-D PAGE, PSA<17:16> READ AS "11"
16548    !
16549    !       BIT!      15   14   13   12   11   10   09   08   07   06   05   04   03   02   01   00
16550    !       FCN!    DATI   BG    0  NPR DATOB DATO PSA  PSA  NIB  LOB  YAG  CON  FLT  PDN  CACH YEL
16551    !                    SERV      TIME              17   16  ERR  ERR  ERR SERV SERV FAIL   ERR ZON
16552    !       (101740)   1    0    0    0    0    0    1    1    1    1    1    0    0    0    0    0
16553
16554    6523!
16555    TEST712D!
16556            P0,     LOAD-ENUA(XTARGET402),                  !SETUP FOR 8-BERO COMPARE
16557                    LOAD-ERROR(TEST712D),                   !ERROR DIRECTORY KEY
16558                    DCS-CTR(C11,),                          !COMPARE AT TARGET
16559            NEXT,   J/EXPEC712D                             !
     (6523)  DCS[1,00,1,0,0,0]  BM[0100,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,101,000,001]
16560
16561    6501!  !(FREE)
16562    EXPEC12D!
16563            P3,     CSPD[02]_EMIT, EMIT/101740,             !DATI(1)H SET, PSA<17:16>="11"
16564            NEXT,   J/GOTEST712D                            !  IN SERVICE REG
     (6501)  DCS[0,00,0,0,0,0]  BM[1000,,10,00,,11,11,,100,,000,,,0,0,0,,0,,0,,,0,1101,,,1,,0000,0,,,11,000,,,101,000,010]
16565
16566    6502!  !(FREE)
16567    GOTEST712D!
16568            SETUP,  RETURN/SCOPE712,                        !GO TO SUBR WHICH:
16569                                                            !  CLR-JAM-ERRORS-(I), FOR NEXT TEST
16570            NEXT,   CALL[CJESERVICETOD]                     !  (SERVICE)-XOR-CSP(02) -> D, SUT(D=IS-ZERO)
     (6502)  DCS[0,00,0,0,0,0]  BM[0101,,00,10,,01,00,,111,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,100,111]
16571
16572
16573
16574
16575    5447!  !(FREE)
16576    SCOPE712!
16577            NEXT,   SUTD[SCOPE],                            !NO ERROR:  "TEST713A"  (+1, WORDS)
16578                    J/TEST713A                              !   ERROR:  "LOADRET712A"  (-12, WORDS)
     (5447)  DCS[0,00,0,1,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,100,110,111]
16579
16580
16581
16582
16583
16584
16585    ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
16586
16587    !TESTING "DATI-NOINT", "ILLEGAL INTERNAL ADDRESS" JAMUPP
16588
16589    ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
```



```
16590
16591    !TEST-713-A DOES A BUS "DATI-NOINT" FUNCTION TO AN INTERNAL ADDRESS, TRYING TO FORCE
16592    !  AN "ILLEGAL INTERNAL ADDRESS" ABORT/JAMUPP
16593    5467!
16594    TEST713A!
16595            P0,     LOAD-ENUA(4777),                        !JAMUPP ADDRESS
16596                    LOAD-ERROR(TEST713A),                   !ERROR DIRECTORY KEY
16597                    DCS-CTR(C4,),                           !COMPARE JUST AFTER BUS CYCLE UWORD, AT JAM
16598                    BUMP-VERIFY,                            !COUNT
16599            NEXT,   J/LOADRET713A                           !
     (5467)  DCS[1,00,1,0,0,1]  BM[1011,,00,10,,01,11,,111,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,100,110,100]
16600
16601    5464!
16602    LOADRET713A!
16603            P1,     CSPD[00]_EMIT, RETURN/TEST713B,         !RETURN AFTER SUCCESSFUL JAM TO NEXT TEST
16604            NEXT,   GOTO-PAGE(7),                           !XFER
16605                    J/BUSFCN713A                            !
     (5464)  DCS[0,00,0,0,0,0]  BM[0101,,10,11,,00,01,,110,,111,,,0,0,0,,0,,0,,,0,1111,,,1,,0000,0,,,11,100,,,001,110,111]
16606
16607    7167!  !(FREE)
16608    BUSFCN713A!
16609            P1,     BA_ASPLO(R11),                          !BAs(177776), GENERATED IN PREVIOUS SET OF TESTs
16610            P2-T,   SR_ASPHI(C000001),                      !SET BIT<00>=(1) FOR JAMUPP EXPECTED
16611            P3,     DATI-NOINT,                             !DO A BUS "DATI-NOINT",
16612                                                            ! SHOULD GET "ILLEGAL INTERNAL ADDR" ABORTED
16613            NEXT,   J/NEXT713A                              !GO DELAY
     (7167)  DCS[0,00,0,0,0,0]  BM[1010,,01,11,,10,01,,000,,000,,,0,0,1,,1,,0,,,1,0001,,,0,,0000,0,,,11,000,,,011,110,101]
16614
16615    !** AT THIS POINT JAMUPP SHOULD OCCUR **
16616    !** CLASSIC FLOW (4777) -> (4757) -> (7XXX) -> (4XXX), AND THEN WE'RE BACK HERE **
16617    !**  RETURN TO ADDRESS LEFT IN CSP(00) **
16618
16619    !*** END UP HERE IF NO JAMUPP ***
16620    7365!
16621    NEXT713A!
16622            SETUP,  RETURN/TEST713A,                        !FORCE A SCOPE LOOP ON THIS TEST, BUT FIRST
16623            NEXT,   GOTO-PAGE(7),                           !  MUST DELAY A FEW MICROWORDS FOR BUS
16624                    J/SUTD-IS-ZERO                          !  ERROR TO TAKE EFFECT  (IGNORE "SUT" OUTCOME HERE)
     (7365)  DCS[0,00,0,0,0,0]  BM[0101,,00,10,,01,10,,111,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,100,001]
16625
16626
16627    !*** END UP HERE IF JAMUPP ***
16628
16629
16630    ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
16631

16632    !TEST-713-B CHECKS THAT THE RIGHT JAM (ILLEGAL INTERNAL ADDRESS) IS INDICATED IN THE JAMREG:
16633    !
16634    !       BIT!      15   14   13   12   11   10   09   08   07   06   05   04   03   02   01   00
16635    !       FCN!     ODD    0  SSYN YEL  RED  NCS  PDN  MEM  SSYN CACH ILL  MGT  RED  ODD  NCS  UBRK
16636    !                ADR      TIME ZON  ZON  PAR  DIS  PAR  TIME  ERR ADR  ADR  ZON  ADR  PAR  TRAP
16637    !       (001040)  0    0    0    0    0    0    1    0    0    0    1    0    0    0    0    0
16638
```

```
16639    5516;
16640    TEST713B;
16641            P0,     LOAD-ENUA(ZTARGET402),                  ;SETUP FOR D=ZERO COMPARE
16642                    LOAD-ERROR(TEST713B),                   ;ERROR DIRECTORY KEY
16643                    DCS-CTR(C10,),                          ;COMPARE AT TARGET
16644                    BUMP-VERIFY,                            ;COUNT
16645            NEXT,   J/EXPECT713B                            ;
     (5516)  DCS[1,00,1,0,0,1]  BM[0101,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,0,0000,,,0,,0000,0,,,11,000,,,100,101,010]
16646
16647    5452;  ;(FREE)
16648    EXPECT713B;
16649            P3,     CSPD[02]_EMIT, EMIT/001040,             ;"ILLEGAL INTERNAL ADDRESS=(1)H" SET
16650            NEXT,   J/GOTEST713B                            ;
     (5452)  DCS[0,00,0,0,0,0]  BM[0000,,10,00,,10,00,,100,,000,,,0,0,0,,0,,0,,,0,1101,,,1,,0000,0,,,11,000,,,100,101,011]
16651
16652    5453;  ;(FREE)
16653    GOTEST713B;
16654            SETUP,  RETURN/TEST713C,                        ;GO TO SUBR WHICH:
16655            NEXT,   CALL[JAMTOD]                            ;  (JAMPRO)=XOR=CSP(02) -> D, BUT(D=IS=ZERO)
     (5453)  DCS[0,00,0,0,0,0]  BM[0110,,00,10,,10,10,,010,,111,,,0,0,0,,0,,0,,0000,,,0,,0000,0,,,11,100,,,010,101,110]
16656
16657
16658
16659    ;  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -
16660
16661    ;TEST-713-C CHECKS THAT THE "OTHER-JAM-H" SIGNAL IS HIGH, INDICATING A JAM PRESENT OTHER
16662    ;   THAN ONLY A VALID "INTERNAL ADDRESS" JAM
16663    5522;
16664    TEST713C;
16665            P0,     LOAD-ENUA(ZTARGET403),                  ;BIT<01> KEY
16666                    LOAD-ERROR(TEST713C),                   ;ERROR DIRECTORY KEY
16667                    DCS-CTR(C3,),                           ;COMPARE AT TARGET
16668            NEXT,   J/GOBUT713C                             ;
     (5522)  DCS[1,00,1,0,0,0]  BM[1100,,00,11,,11,00,,000,,011,,,0,0,0,,0,,0,,0,0000,,,0,,0000,0,,,11,000,,,101,000,011]
16669
16670    5503;  ;(FREE)
16671    GOBUT713C;
16672            SETUP,  RETURN/TEST713D,                        ;RETURN TO START OF NEXT SUBTEST
16673            NEXT,   GOTO-PAGE(7),                           ;BUT TABLE
16674                    J/BUTOTHERJAM                           ;"OTHER JAM H" IN BIT<01>
     (5503)  DCS[0,00,0,0,0,0]  BM[0110,,00,10,,10,10,,001,,111,,,0,0,0,,0,,0,,0,0000,,,0,,0000,0,,,11,100,,,011,101,010]
16675
16676
16677
16678    ;  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -
16679
16680    ;TEST-713-D CHECKS THAT THE RIGHT BUS FUNCTION DECODE / PBA<17:16> ARE INDICATED IN SERVICE REG:
16681    ;  BACK IN 16, BIT MODE, SINCE I-D PAGE, PBA<17:16> READ AS "11"
16682    ;
16683    ;       BIT:      15   14   13   12   11   10   09   08   07   06   05   04   03   02   01   00
16684    ;       FCN:    DATI   BG    0  NPR DATOB DATO PBA  PBA  HIE  LOS  TAG  CON  PLT  PON  CACH YEL
16685    ;                  SERV      TIME              17   16  ERR  ERR  ERR SERV SERV FAIL   ERR ZON
```



```
16686    ;       [101740]   1    0    0    0    0    0    1    1    1    1    1    0    0    0    0    0
16687
16688    5521;
16689    TEST713D;
16690            P0,     LOAD-ENUA(ZTARGET402),                  ;SETUP FOR D=ZERO COMPARE
16691                    LOAD-ERROR(TEST713D),                   ;ERROR DIRECTORY KEY
16692                    DCS-CTR(C11,),                          ;COMPARE AT TARGET
16693                    BUMP-VERIFY,                            ;COUNT
16694            NEXT,   J/EXPECT713D                            ;
     (5521)  DCS[1,00,1,0,0,1]  BM[0100,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,0,0000,,,0,,0000,0,,,11,000,,,101,000,100]
16695
16696    5504;  ;(FREE)
16697    EXPECT713D;
16698            P3,     CSPD[02]_EMIT, EMIT/101740,             ;DATI(1)H SET, PBA<17:16>="11"
16699            NEXT,   J/GOTEST713D                            ;  IN SERVICE REG
     (5504)  DCS[0,00,0,0,0,0]  BM[1000,,10,00,,11,11,,100,,000,,,0,0,0,,0,,0,,,0,1101,,,1,,0000,0,,,11,000,,,101,000,101]
16700
16701    5505;  ;(FREE)
16702    GOTEST713D;
16703            SETUP,  RETURN/SCOPE713,                        ;GO TO SUBR WHICH:
16704                                                            ;  CLR-JAM-ERRORS-[1], FOR NEXT TEST
16705            NEXT,   CALL[CJESERVICETOD]                     ;  (SERVICE)=XOR=CSP(02) -> D, BUT(D=IS=ZERO)
     (5505)  DCS[0,00,0,0,0,0]  BM[0101,,00,10,,01,01,,001,,111,,,0,0,0,,0,,0,,0,0000,,,0,,0000,0,,,11,100,,,010,100,111]
16706
16707
16708
16709
16710    5451;  ;(FREE)
16711    SCOPE713;
16712            NEXT,   SUTD[SCOPE],                            ;NO ERROR:  "TEST720A"  (+1, WORDS)
16713                    J/TEST720A                              ;   ERROR:  "LOADRET713A"  (-11, WORDS)
     (5451)  DCS[0,00,0,1,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,100,110,101]
16714
16715
16716
16717
16718
16719
16720
16721    ;  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -
16722
16723    ;TESTING "DATIB=-BYTE", "ODD-ADDRESS" JAMUPP, I/O-PAGE-PBA<17:16> DECODE
16724
16725    ;  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -
16726
16727    ;TEST-720-A ATTEMPTS TO DO A BUS "DATIB=-BYTE" FUNCTION, TRYING TO FORCE AN "ODD ADDRESS"

16728    ;  ABORT/JAMUPP, ALSO CHECKING THAT ALL SERVICE / JAM STATUS BITS SET CORRECTLY
16729    5465;
16730    TEST720A;
16731            P0,     LOAD-ENUA(4777),                        ;JAMUPP ADDRESS
16732                    LOAD-ERROR(TEST720A),                   ;ERROR DIRECTORY KEY
16733                    DCS-CTR(C7,),                           ;COMPARE JUST AFTER BUS CYCLE UWORD, AT JAM
16734            NEXT,   J/LOADRET720A                           ;
```

```
     (5465)  DCS[1,00,1,0,0,0]  BM[1000,,00,10,,01,11,,111,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,100,110,010]

        5462:
        LOADRET720A:
                P1,     CSPD[00]_EMIT, RETURN/TEST720B,          !RETURN AFTER SUCCESSFUL JAM TO NEXT TEST
                NEXT,   GOTO-PAGE(7),                            !XFER
                        J/LOADIR720A                             !
     (5462)  DCS[0,00,0,0,0,0]  BM[0110,,10,10,,10,10,,000,,111,,,0,0,0,,0,,0,,,0,1111,,,1,,0000,0,,,11,100,,,001,111,010]

        7172:  !(FREE)
        LOADIR720A:
                P2=U,   IR_EMIT, EMIT/000000,                    !"HALT" INSTRUCTION IS +BYTE
                NEXT,   J/SETADR720A                             !
     (7172)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,1,1010,,,0,,0000,0,,,11,000,,,001,111,100]

        7174:  !(FREE)
        SETADR720A:
                P3,     CSPD[16]_EMIT, EMIT/060001,              !BIT<15:13>="011", -IOPAGE, ODD-ADDRESS
                NEXT,   J/SETJAM720A                             !
     (7174)  DCS[0,00,0,0,0,0]  BM[0110,,10,00,,00,00,,000,,001,,,0,0,0,,0,,0,,,0,0001,,,1,,0000,0,,,11,000,,,001,111,101]

        7175:  !(FREE)
        SETJAM720A:
                P2=T,   SR_CSPD(D16),                            !SET BIT<00>=(1) FOR JAMUPP EXPECTED
                NEXT,   J/BUSFCN720A                             !
     (7175)  DCS[0,00,0,0,0,0]  BM[1010,,10,00,,00,00,,000,,000,,,0,0,1,,0,,0,,,0,0001,,,0,,0000,0,,,11,000,,,001,111,110]

        7176:  !(FREE)
        BUSFCN720A:
                P1,     BA_SR,                                   !SET BA<17:00>=(060001)
                P3,     DATIB,                                   !DO A BUS "DATIB=-BYTE", SHOULD GET ODD ADDRESS ABORTED
                NEXT,   J/NEXT720A                               !GO DELAY
     (7176)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,1,,0,,,1,0011,,,0,,0000,0,,,11,000,,,011,110,100]

        !** AT THIS POINT JAMUPP SHOULD OCCUR **
        !** CLASSIC FLOW (4777) -> (4757) -> (7XXX) -> (4XXX), AND THEN WE'RE BACK HERE **
        !**  RETURN TO ADDRESS LEFT IN CSP(00) **

        !*** END UP HERE IF NO JAMUPP ***
        7364:
        NEXT720A:
                SETUP,  RETURN/TEST720A,                         !FORCE A SCOPE LOOP ON THIS TEST, BUT FIRST
                NEXT,   GOTO-PAGE(7),                            !  MUST DELAY A FEW MICROWORDS FOR BUS
                        J/BUTD=IS=ZERO                           !  ERROR TO TAKE EFFECT  (IGNORE "BUT" OUTCOME HERE)
     (7364)  DCS[0,00,0,0,0,0]  BM[0101,,00,10,,01,10,,101,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,100,001]


        !*** END UP HERE IF JAMUPP ***


        ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        !TEST-720-B CHECKS THAT THE RIGHT JAM (ODD ADDRESS) IS INDICATED IN THE JAMREG:
```



```
        !
        !       BIT:      15   14   13   12   11   10   09   08   07   06   05   04   03   02   01   00
        !       FCN:      ODD   0  ASYN YEL  RED  MCB  PON  MEM  ASYN CACH ILL  MGT  RED  ODD  MCB  UBRK
        !                 ADR      TIME ZON  ZON  PAR  DIS  PAR  TIME ERR  ADR  ABT  ZON  ADR  PAR  TRAP
        !       (101004)   1    0    0    0    0    0    1    0    0    0    0    0    0    1    0    0

        6520:
        TEST720B:
                P0,     LOAD-ENUA(ZTARGET402),                   !SETUP FOR D=ZERO COMPARE
                        LOAD-ERROR(TEST720B),                    !ERROR DIRECTORY KEY
                        DCS-CTR(C10,),                           !COMPARE AT TARGET
                        BUMP-VERIFY,                             !COUNT
                NEXT,   J/GOTEST720B                             !
     (6520)  DCS[1,00,1,0,0,1]  BM[0101,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,101,000,110]

        6506:  !(FREE)
        GOTEST720B:
                SETUP,  RETURN/TEST720C,                         !GO TO SUBR WHICH:
                NEXT,   CALL[ODDJAMTOD]                          !  (JAMREG)-XOR-(101004) -> D, BUT(D=ZERO)   [ODD-ADDRESS]
     (6506)  DCS[0,00,0,0,0,0]  BM[0100,,00,11,,11,10,,000,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,101,100]



        ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        !TEST-720-C CHECKS THAT THE RIGHT BUS FUNCTION DECODE / PBA<17:16> ARE INDICATED IN SERVICE REG:
        !  NOTE:  PBA<17:16> SHOULD NOT BE FORCED TO "11", IN 16. BIT MODE, WHEN BA<15:13>="011"
        !
        !       BIT:      15   14   13   12   11   10   09   08   07   06   05   04   03   02   01   00
        !       FCN:     DATI  BG    0  NPR DATOB DATO PBA  PBA  PIB  LOB  TAG  CON  FLT  PON  CACH TEL
        !                      SERV     TIME                17   16  ERR  ERR  ERR SERV SERV FAIL  ERR ZON
        !       (100340)   1    0    0    0    0    0    0    0    1    1    1    0    0    0    0    0

        4760:
        TEST720C:
                P0,     LOAD-ENUA(ZTARGET402),                   !SETUP FOR D=ZERO COMPARE
                        LOAD-ERROR(TEST720C),                    !ERROR DIRECTORY KEY
                        DCS-CTR(C11,),                           !COMPARE AT TARGET
                NEXT,   J/GOTEST720C                             !
     (4760)  DCS[1,00,1,0,0,0]  BM[0100,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,011,001,111]

        4317:  !(FREE)
        GOTEST720C:
                SETUP,  RETURN/TEST721A,                         !GO TO SUBR WHICH:
                                                                 !  CLR-JAM-ERRORS, TO RESET FOR NEXT TEST
                NEXT,   CALL[DATISERVICETOD]                     !  (SERVICE)-XOR-(100340) -> D, BUT(D=ZERO)   (DATI[1]H)

     (4317)  DCS[0,00,0,0,0,0]  BM[0100,,00,10,,11,11,,001,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,100,101]
```

```


	;- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

	;TESTING "DATOB*-BYTE", "ODD-ADDRESS" JAMUPP, I/O-PAGE-PBA<17:16> DECODE

	;- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

	;TEST-721-A ATTEMPTS TO DO A BUS "DATOB*-BYTE" FUNCTION, TRYING TO FORCE AN "ODD ADDRESS"
	;  ABORT/JAMUPP, ALSO CHECKING THAT ALL SERVICE / JAM STATUS BITS SET CORRECTLY
	4571:
	TEST721A:
		P0,	LOAD-ENUA(4777),			;JAMUPP ADDRESS
			LOAD-ERROR(TEST721A),			;ERROR DIRECTORY KEY
			DCS-CTR(C6.),				;COMPARE JUST AFTER BUS CYCLE UWORD, AT JAM
		NEXT,	J/LOADRET721A				;
 (4571)	DCS[1,00,1,0,0,0]  BM[1001,,00,10,,01,11,,111,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,011,010,000]

	4320:	;(FREE)
	LOADRET721A:
		P3,	CSPD[00]_EMIT, RETURN/TEST721B,		;RETURN AFTER SUCCESSFUL JAM TO NEXT TEST
		NEXT,	GOTO-PAGE(7),				;XFER
		NEXT,	J/SETADR721A				;
 (4320)	DCS[0,00,0,0,0,0]  BM[0100,,10,10,,11,11,,011,,111,,,0,0,0,,0,,0,,,0,1111,,,1,,0000,0,,,11,100,,,001,111,011]

	7173:	;(FREE)
	SETADR721A:
		P3,	CSPD[16]_EMIT, EMIT/120001,		;BIT<15:13>="101", -IOPAGE; ODD-ADDRESS
		NEXT,	J/SETJAM721A				;
 (7173)	DCS[0,00,0,0,0,0]  BM[1010,,10,00,,00,00,,000,,001,,,0,0,0,,0,,0,,,0,0001,,,1,,0000,0,,,11,000,,,010,000,000]

	7200:	;(FREE)
	SETJAM721A:
		P2-T,	SR_CSPD(D16),				;SET BIT<00>=(1) FOR JAMUPP EXPECTED
		NEXT,	J/BUSFCN721A				;
 (7200)	DCS[0,00,0,0,0,0]  BM[1010,,10,00,,00,00,,000,,000,,,0,0,1,,0,,0,,,0,0001,,,0,,0000,0,,,11,000,,,010,000,001]

	7201:	;(FREE)
	BUSFCN721A:
							;NOTE:  "IR"=(000000)=-BYTE FROM PREV TEST
		P1,	BA_SR,					;SET BA<17:00>=(120001)
		P3,	DATOB,					;DO A BUS "DATOB*-BYTE", SHOULD GET ODD ADDRESS ABORTED
		NEXT,	J/NEXT721A				;GO DELAY
 (7201)	DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,1,,0,,,1,0101,,,0,,0000,0,,,11,000,,,011,110,010]

	;** AT THIS POINT JAMUPP SHOULD OCCUR **
	;** CLASSIC FLOW (4777) -> (4757) -> (7XXX) -> (4XXX), AND THEN WE'RE BACK HERE **
	;**  RETURN TO ADDRESS LEFT IN CSP(00) **

	;*** END UP HERE IF NO JAMUPP ***
	7362:
	NEXT721A:
		SETUP,	RETURN/TEST721A,			;FORCE A SCOPE LOOP ON THIS TEST, BUT FIRST
```



```
		NEXT,	GOTO-PAGE(7),				; MUST DELAY A FEW MICROWORDS FOR BUS
			J/BUTD-IS-ZERO				; ERROR TO TAKE EFFECT  (IGNORE "BUT" OUTCOME HERE)
 (7362)	DCS[0,00,0,0,0,0]  BM[0100,,00,10,,11,11,,001,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,100,001]


	;*** END UP HERE IF JAMUPP ***


	;- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

	;TEST-721-B CHECKS THAT THE RIGHT JAM (ODD ADDRESS) IS INDICATED IN THE JAMREG:
	;
	;	BIT:	 15   14   13   12   11   10   09   08   07   06   05   04   03   02   01   00
	;	FCN:	ODD    0  SSYN YEL  RED  WCS  PON  WEM  SSYN CACM ILL  MGT  RED  ODD  WCS  UBRK
	;		ADR       TIME ZON  ZON  PAR  DIS  PAR  TIME ERR  ADR  ABT  ZON  ADR  PAR  TRAP
	;	(101004) 1    0    0    0    0    0    1    0    0    0    0    0    0    1    0    0

	4573:
	TEST721B:
		P0,	LOAD-ENUA(ZTARGET402),			;SETUP FOR D=ZERO COMPARE
			LOAD-ERROR(TEST721B),			;ERROR DIRECTORY KEY
			DCS-CTR(C10.),				;COMPARE AT TARGET
		NEXT,	J/GOTEST721B				;
 (4573)	DCS[1,00,1,0,0,0]  BM[0101,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,011,010,001]

	4321:	;(FREE)
	GOTEST721B:
		SETUP,	RETURN/TEST721C,			;GO TO SUBR WHICH:
		NEXT,	CALL[ODDJAMTOD]				; (JAMREG)-XOR-(101004) => D, BUT(D=ZERO)  [ODD-ADDRESS]
 (4321)	DCS[0,00,0,0,0,0]  BM[0100,,00,10,,11,11,,111,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,101,100]



	;- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

	;TEST-721-C CHECKS THAT THE RIGHT BUS FUNCTION DECODE / PBA<17:16> ARE INDICATED IN SERVICE REG:
	;  NOTE:  PBA<17:16> SHOULD NOT BE FORCED TO "11", IN 16, BIT MODE, WHEN BA<15:13>="101"
	;
	;	BIT:	 15   14   13   12   11   10   09   08   07   06   05   04   03   02   01   00
	;	FCN:	DATI  BG    0  NPR DATOB DATO PBA  PBA  HIB  LOB  TAG  CON  FLT  POW  CACH YEL
	;		     SERV     TIME                17   16  ERR  ERR  ERR SERV SERV FAIL  ERR ZON
	;	(002340) 0    0    0    0    0    1    0    0    1    1    1    0    0    0    0    0

	4577:
	TEST721C:
		P0,	LOAD-ENUA(ZTARGET402),			;SETUP FOR D=ZERO COMPARE

			LOAD-ERROR(TEST721C),			;ERROR DIRECTORY KEY
			DCS-CTR(C11.),				;COMPARE AT TARGET
		NEXT,	J/GOTEST721C				;
 (4577)	DCS[1,00,1,0,0,0]  BM[0100,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,011,010,010]

	4322:	;(FREE)
	GOTEST721C:
```

```
        SETUP,  RETURN/SCOPE721,                        !GO TO SUBR WHICH:
                                                        !  CLR-JAM-ERRORS, TO RESET FOR NEXT TEST
        NEXT,   CALL[DATOSERVICETOD]                    !  [SERVICE]-XOR-[002340] -> D, BUT(D=ZERO)  (DATO(1)H)
 (4322)  DCS[0,00,0,0,0,0]  BM[0101,,00,10,,01,01,,100,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,100,110]




   5454:  !(FREE)
   SCOPE721:
        NEXT,   BUTD[SCOPE],                            !NO ERROR:  "TEST722A"  (+1, WORDS)
                J/TEST722A                              !   ERROR:  "LOADRET720A"  (-24, WORDS)
 (5454)  DCS[0,00,0,1,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,100,110,011]





   ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

   !TESTING "INVALIDATE", "ODD-ADDRESS" JAMUPP, I/O-PAGE-PBA<17:16> DECODE

   ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

   !TEST-722-A ATTEMPTS TO DO A BUS "INVALIDATE" FUNCTION, TRYING TO FORCE AN "ODD ADDRESS"
   !  ABORT/JAMUPP, ALSO CHECKING THAT ALL SERVICE / JAM STATUS BITS SET CORRECTLY
   5463:
   TEST722A:
        P0,     LOAD-ENUA(4777),                        !JAMUPP ADDRESS
                LOAD-ERROR(TEST722A),                   !ERROR DIRECTORY KEY
                DCS-CTR(C6,),                           !COMPARE JUST AFTER BUS CYCLE UWORD, AT JAM
        NEXT,   J/LOADRET722A                           !
 (5463)  DCS[1,00,1,0,0,0]  BM[1001,,00,10,,01,11,,111,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,100,101,110]

   5456:  !(FREE)
   LOADRET722A:
        P3,     CSPD[00]_EMIT, RETURN/TEST722B,         !RETURN AFTER SUCCESSFUL JAM TO NEXT TEST
        NEXT,   GOTO-PAGE(4),                           !XFER
        NEXT,   J/SETADR722A                            !
 (5456)  DCS[0,00,0,0,0,0]  BM[0100,,10,10,,01,10,,100,,100,,,0,0,0,,0,,0,,,0,1111,,,1,,0000,0,,,11,100,,,100,111,000]

   4410:
   SETADR722A:
        P3,     CSPD[16]_EMIT, EMIT/140001,             !BIT<15:13>="110", =IOPAGE; ODD-ADDRESS
        NEXT,   J/SETJAM722A                            !
 (4410)  DCS[0,00,0,0,0,0]  BM[1100,,10,00,,00,00,,000,,001,,,0,0,0,,0,,0,,,0,0001,,,1,,0000,0,,,11,000,,,011,010,011]

   4323:  !(FREE)
   SETJAM722A:
        P7-T,   SR_CSPD(D16),                           !SET BIT<08>=(1) FOR JAMUPP EXPECTED
        NEXT,   J/BUSFCN722A                            !
 (4323)  DCS[0,00,0,0,0,0]  BM[1010,,10,00,,00,00,,000,,000,,,0,0,1,,0,,0,,,0,0001,,,0,,0000,0,,,11,000,,,011,010,100]
```



```
   4324:  !(FREE)
   BUSFCN722A:
        P1,     BA_SR,                                  !SET BA<17:00>=(140001)
        P3,     INVALIDATE,                             !DO A BUS "INVALIDATE", SHOULD GET ODD ADDRESS ABORTED
        NEXT,   J/NEXT722A                              !GO DELAY
 (4324)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,1,,0,,,1,0111,,,0,,0000,0,,,11,000,,,101,010,100]

   !** AT THIS POINT JAMUPP SHOULD OCCUR **
   !** CLASSIC FLOW (4777) -> (4757) -> (7XXX) -> (4XXX), AND THEN WE'RE BACK HERE **
   !**  RETURN TO ADDRESS LEFT IN CSP(00) **

   !*** END UP HERE IF NO JAMUPP ***
   4524:
   NEXT722A:
        SETUP,  RETURN/TEST722A,                        !FORCE A SCOPE LOOP ON THIS TEST, BUT FIRST
        NEXT,   GOTO-PAGE(7),                           !  MUST DELAY A FEW MICROWORDS FOR BUS
                J/BUTD-IS-ZERO                          !  ERROR TO TAKE EFFECT  (IGNORE "BUT" OUTCOME HERE)
 (4524)  DCS[0,00,0,0,0,0]  BM[0101,,00,10,,01,10,,011,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,100,001]


   !*** END UP HERE IF JAMUPP ***



   ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

   !TEST-722-B CHECKS THAT THE RIGHT JAM (ODD ADDRESS) IS INDICATED IN THE JAMREG:
   !
   !       BIT:      15    14   13   12   11   10   09   08   07   06   05   04   03   02   01   00
   !       FCN:      ODD    0  SSYN YEL  RED  NCS  POW  MEM  SSYN CACH ILL  NOT  RED  ODD  NCS  UBRK
   !                 ADR       TIME ZON  ZON  PAR  DIS  PAR  TIME ERR  ADR  ABT  ZON  ADR  PAR  TRAP
   !       (101004)   1     0    0    0    0    0    1    0    0    0    0    0    0    1    0    0

   4464:
   TEST722B:
        P0,     LOAD-ENUA(ETARGET402),                  !SETUP FOR D=ZERO COMPARE
                LOAD-ERROR(TEST722B),                   !ERROR DIRECTORY KEY
                DCS-CTR(C11,),                          !COMPARE AT TARGET
                BUMP-VERIFY,                            !COUNT
        NEXT,   J/GOTEST722B                            !
 (4464)  DCS[1,00,1,0,0,1]  BM[0100,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,011,010,101]

   4125:  !(FREE)
   GOTEST722B:
        SETUP,  RETURN/TEST722C,                        !GO TO SUBR WHICH:
        NEXT,   CALL[ODDJAMTOD]                         !  (JAMREG)-XOR-[101004] -> D, BUT(D=ZERO)  [ODD-ADDRESS]

 (4125)  DCS[0,00,0,0,0,0]  BM[0100,,00,10,,01,10,,101,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,101,100]



   ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
```

```
        ;TEST-722-C CHECKS THAT THE RIGHT BUS FUNCTION DECODE / PBA<17:16> ARE INDICATED IN SERVICE REG:
        ;  NOTE:  PBA<17:16> SHOULD NOT BE FORCED TO "11", IN 16. BIT MODE, WHEN BA<15:13>="110"
        ;
        ;        BIT:        15   14    13   12   11    10    09   08   07   06   05   04   03   02   01   00
        ;        FCN:       DATI  BG     0  NPR DATOB DATO  PBA  PBA  NXM  LOB  TAG  CON  FLT  PON  CACH YEL
        ;                        SERV       TIME              17   16  ERR  ERR  ERR  SERV SERV FAIL  ERR ZON
        ;        (002340)    0    0     0    0    0     1    0    0    1    1    1    0    0    0    0    0

        4465:
        TEST722C:
                P0,     LOAD-ENUA(ZTARGET402),                 ;SETUP FOR D=ZERO COMPARE
                        LOAD-ERROR(TEST722C),                  ;ERROR DIRECTORY KEY
                        DCS-CTR(C11.),                         ;COMPARE AT TARGET
                NEXT,   J/GOTEST722C                           ;
   (4465)  DCS[1,00,1,0,0,0]  BM[0100,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,011,010,110]

        4326:  ;(FREE)
        GOTEST722C:
                SETUP,  RETURN/SCOPE722,                       ;GO TO ADDR WHICH:
                                                               ;  CLR-TRAP-ERRORS, TO RESET FOR NEXT TEST
                NEXT,   CALL[DATOSERVICETOD]                   ;  (SERVICE)-XOR-(002340) => D, BUT(D=ZERO)  (DATO[1]H)
   (4326)  DCS[0,00,0,0,0,0]  BM[0100,,00,01,,10,10,,111,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,100,110]




        4327:  ;(FREE)
        SCOPE722:
                NEXT,   BUTO[SCOPE],                           ;NO ERROR:  "TEST730A"  (+1. WORDS)
                        J/TEST730A                             ;  ERROR:   "SETADR720A"  (-11. WORDS)
   (4327)  DCS[0,00,0,1,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,100,111,001]





        ;.PAGE=================================================================================================

        .TOC    =    TEST730-731:  BUS CYCLES TO/FROM MEMORY

        ;******************************************************************************************
        ;*
        ;*      TESTS:  730A - 731E                             UWORDS:  063 + 057
        ;*
        ;*      FUNCTIONS:
        ;*
        ;*      THESE TESTS CHECK THAT ACTUAL BUS CYCLES CAN BE CORRECTLY EXECUTED,
        ;*
        ;******************************************************************************************
```



```
        ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        ;THIS FIRST SERIES OF TESTS DOES A DATO/DATIP/DATO/DATIB SEQUENCE, CHECKING THAT
        ;EACH FUNCTION OPERATES AS EXPECTED.

        ;  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -

        ;TEST-730-A DOES A DATO, AND THEN CHECKS THAT THE DBUF LATCH (DS) ALSO GETS LOADED WITH THE
        ;  DATA, AND THAT IT IS ENABLED ON BUSDIN IN THE MICROWORD AFTER THE BUS CYCLE (IE, EMIT
        ;  IS TEMPORARILY DISABLED)
        4471:
        TEST730A:
                P0,     LOAD-ENUA(ZTARGET402),                 ;SETUP FOR D=ZERO COMPARE
                        LOAD-ERROR(TEST730A),                  ;ERROR DIRECTORY KEY
                        DCS-CTR(C9.),                          ;COMPARE AT TARGET
                NEXT,   J/LOADIR730A                           ;
   (4471)  DCS[1,00,1,0,0,0]  BM[0110,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,111,101,100]

        4754:
        LOADIR730A:
                P2-U,   IR_EMIT, EMIT/125200,                  ;"CMP-BYTE" INSTR; INSTRS-E09(412) DECODE
                NEXT,   J/LOADDATA730A                         ;
   (4754)  DCS[0,00,0,0,0,0]  BM[1010,,00,10,,10,10,,000,,000,,,0,0,0,,0,,0,,,1,1010,,,0,,0000,0,,,11,000,,,011,011,000]

        4330:  ;(FREE)
        LOADDATA730A:
                P3,     CSPD[16]_EMIT, EMIT/125252,            ;PATTERN (125252) IN BASCON AREA
                NEXT,   J/BUSFCN730A                           ;
   (4330)  DCS[0,00,0,0,0,0]  BM[1010,,10,10,,10,10,,101,,010,,,0,0,0,,0,,0,,,0,0001,,,1,,0000,0,,,11,000,,,011,011,001]

        4331:  ;(FREE)
        BUSFCN730A:
                P0,     BUMP-VERIFY,                           ;COUNT
                P1,     BA_ASPHI(C000000),                     ;USE MEMORY ADDR(000000)
                P2-T,   D_CSPR(B16), DIC]_0,                   ;USE DATA (125252)
                P3,     DATO,                                  ;FOR A BUS "DATO" CYCLE
                NEXT,   J/GETDBUF730A                          ;
   (4331)  DCS[0,00,0,0,0,1]  BM[1010,,10,01,,11,01,,100,,000,,,0,1,0,,1,,0,,,1,0010,,,0,,0000,0,,,11,000,,,011,011,010]

        4332:  ;(FREE)
        GETDBUF730A:
                P0,     BUMP-VERIFY,                           ;COUNT
                P1,     CSPD[17]_BUSDIN, EMIT/037777,          ;DBUF SHOULD BE ENABLED; EMIT IS NOISE

                P3-T,   D_JUNK, SAVE-DIC],                     ;MANGLE DATA IN D, DONT CARE WHAT RESULTS
                NEXT,   J/COMP730A                             ;
   (4332)  DCS[0,00,0,0,0,1]  BM[0011,,10,11,,11,11,,111,,111,,,1,1,0,,0,,0,,,0,0000,,,1,,0000,0,,,11,000,,,011,011,011]

        4333:  ;(FREE)
        COMP730A:
                P0,     BUMP-VERIFY,                           ;COUNT
                P2-T,   D_CSPD[17]-XOR-ASPHI(C125252),         ;COMPARE RECEIVED:EXPECTED
```

```
        NEXT,   J/ZAPDBUF730A                             ;
(4333)  DCS[0,00,0,0,0,1]  BM[0110,,10,00,,11,01,,110,,000,,,0,1,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,011,011,100]

    4334: ;(FREE)
    ZAPDBUF730A:
        P3,     DBUF_D-[I],                               ;COPY ZEROED(?) D-REG INTO DBUF
        NEXT,   J/GOBUT730A                               ;
(4334)  DCS[0,00,0,0,0,0]  BM[0100,,00,00,,00,00,,000,,100,,,0,0,0,,0,,0,,,1,1011,,,0,,0000,0,,,11,000,,,011,011,101]

    4335: ;(FREE)
    GOBUT730A:
        SETUP,  RETURN/TEST730B,                          ;RETURN TO START OF NEXT SUBTEST
        NEXT,   GOTO-PAGE(7),                             ;BUT TABLE
                J/BUTD-IS-ZERO                            ;CHECK FOR EQUALITY
(4335)  DCS[0,00,0,0,0,0]  BM[0100,,00,10,,01,10,,110,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,100,001]







    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

    ;TEST-730-B DOES A DATIP, AND THEN CHECKS THAT:
    ;  1) NO ODD ADDRESS ERROR RESULTS
    ;  2) THE RIGHT DATA (OUTPUT ABOVE) IS RETRIEVED  (NOTE DBUF LATCH WAS ZEROED
    ;     TO ALTER ITS COPY OF THE DATA)
    ;  3) THE BUS HOLDING FUNCTION OF THE "DATIP" SHOULD BE EMPLOYED; "BBSY" SHOULD REMAIN ASSERTED
    ;      WELL PAST THE "NORMAL" 1 MICROWORD AFTER THE BUS CYCLE.  IN FACT, IT SHOULD REMAIN ASSERTED
    ;      (HOLDING BUSDIN=UNIBUS-DATA-BUFFER, NOT EMIT, ETC) UNTIL CLEARED BY ANOTHER BUS CYCLE
    ;      (NOT A DATIP), OR A BUTA(LAST) [DONE HERE],
    4466:
    TEST730B:
        P0,     LOAD-ENUA(ZTARGET402),                    ;SETUP FOR D=ZERO COMPARE
                LOAD-ERROR(TEST730B),                     ;ERROR DIRECTORY KEY
                DCS-CTR(C7,),                             ;COMPARE AT TARGET
        NEXT,   J/EXPECT730B                              ;
(4466)  DCS[1,00,1,0,0,0]  BM[1000,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,011,011,110]

    4336: ;(FREE)
    EXPECT730B:
        P3,     CSPD[02]_EMIT, EMIT/000340,               ;SERVICE PORT BITS FOR A "DATIP"
        NEXT,   J/BUSFCN730B                              ;
(4336)  DCS[0,00,0,0,0,0]  BM[0000,,10,00,,00,11,,100,,000,,,0,0,0,,0,,0,,,0,1101,,,1,,0000,0,,,11,000,,,011,011,111]

    4337: ;(FREE)
    BUSFCN730B:
        P1,     BA_ASPHI(C000001),                        ;USE MEMORY ADDR(000001), ODD BYTE
        P2-T,   D_ZERO, D[C]_ALU15,                       ;ZAP D
        P3,     DATIP,                                    ;FOR A BUS "DATIP" CYCLE
                BUTA(CLR-FLAG-RES-UCON),                  ;RESET BUSDIN TO EMIT/ZAP DBUF_D UCON
        NEXT,   J/GETIT730B                               ;
```



```
(4337)  DCS[0,00,0,0,0,0]  BM[0011,,00,00,,11,01,,000,,100,,,0,1,0,,1,,0,,,1,0100,,,0,,0000,0,,,11,010,,,011,100,000]

    4340: ;(FREE)
    GETIT730B:
        P0,     BUMP-VERIFY,                              ;COUNT
        P3,     CSPD[17]_BUSDIN, EMIT/052525,             ;UNIBUS DATA SHOULD BE ENABLED; EMIT IS NOISE
        NEXT,   J/GOBUT730B                               ;
(4340)  DCS[0,00,0,0,0,1]  BM[0101,,10,01,,01,01,,010,,101,,,0,0,0,,0,,0,,,0,0000,,,1,,0000,0,,,11,000,,,011,100,001]

    4341: ;(FREE)
    GOBUT730B:
        SETUP,  RETURN/TEST730C,                          ;EXEC SUBR WHICH:
        NEXT,   CALL[CSP17XOR125252]                      ;  CSP(17)-XOR-(125252) => D, BUT(D=ZERO)
(4341)  DCS[0,00,0,0,0,0]  BM[0100,,00,10,,01,10,,000,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,000,011]






    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

    ;TEST-730-C NOW CHECKS THAT THE "DATIP" FUNCTION IS STILL HOLDING THE BUS BY
    ;  VERIFYING THAT THE UNIBUS DATA BUFFER IS STILL ENABLED ON BUSDIN, DATA=(125252)
    4460:
    TEST730C:
        P0,     LOAD-ENUA(ZTARGET402),                    ;SETUP FOR D=ZERO COMPARE
                LOAD-ERROR(TEST730C),                     ;ERROR DIRECTORY KEY
                DCS-CTR(C5,1),                            ;COMPARE AT TARGET
                BUMP-VERIFY,                              ;COUNT
        NEXT,   J/GOBUT730C                               ;
(4460)  DCS[1,00,1,0,0,1]  BM[1010,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,011,100,010]

    4342: ;(FREE)
    GOBUT730C:
        SETUP,  RETURN/TEST730C1,                         ;EXEC SUBR WHICH:
                                                          ;  1) BUSDIN => CSP(17), EMIT=(052525)
        NEXT,   CALL[BUSDINXOR125252]                     ;  2) CSP(17)-XOR-(125252) => D, BUT(D=ZERO)
(4342)  DCS[0,00,0,0,0,0]  BM[0100,,00,10,,01,01,,000,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,001,111,111]





    ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

    ;TEST-730-C1 NOW CHECKS THAT THE "DATIP" FUNCTION WILL RELEASE THE BUS BY

    ;  VERIFYING THAT THE UNIBUS DATA BUFFER IS NOT ENABLED ON BUSDIN, DATA=(125252),
    ;  AFTER ISSUING BUTA(LAST), WHICH SHOULD CLEAR THE DATIP/BBSY FLOP
    4450:
    TEST730C1:
        P0,     LOAD-ENUA(ZTARGET402),                    ;SETUP FOR D=ZERO COMPARE
                LOAD-ERROR(TEST730C1),                    ;ERROR DIRECTORY KEY
```

```
17214                   DCS=CTR(C8,),                           ;COMPARE AT TARGET
17215           P3,     BUTA(LAST),                             ;ACTIVE BUT EFFECT;  CLEAR OUT DATIP
17216           NEXT,   J/GOBUT730C1                            ;
     (4450)  DCS(1,00,1,0,0,0)  BM(1010..00,11..11,00..000..010...0.0.0..0..0...0.0000...0..0000,0...10,000...100,100,011)
17217
17218   4443;
17219   GOBUT730C1:
17220           SETUP,  RETURN/TEST730D,                        ;EXEC SUBR WHICH:
17221                                                           ;  1) BUSDIN => CSP(17);  EMIT=(052525)
17222           NEXT,   CALL(BUSDTNXOR052525)                   ;  2) CSP(17)=XOR=(052525) => D, BUT(D=ZERO)
     (4443)  DCS(0,00,0,0,0,0)  BM(0100..00,10..01,10..001..111...0.0.0...0..0...0.0000...0..0000,0...11,100...010,000,100)
17223
17224
17225
17226
17227   ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
17228
17229   ;TEST-730-D DOES A DATO FOLLOWED BY A DATIB+BYTE+ODD, AND THEN CHECKS THAT THE RIGHT DATA
17230   ;  IS WRITTEN/RETRIEVED FROM MEMORY LOCATIONS (000000)/(000001)
17231   ;  AND THAT THE EMIT-DISABLE/UNIBUS DATA BUFFER ENABLE IS HANDLED CORRECTLY
17232   4461;
17233   TEST730D:
17234           P0,     LOAD=ENUA(%TARGET402),                  ;SETUP FOR D=ZERO COMPARE
17235                   LOAD=ERROR(TEST730D),                   ;ERROR DIRECTORY KEY
17236                   DCS=CTR(C15,),                          ;HOLD UP FOR NOW
17237           NEXT,   J/LOADDATA730D                          ;
     (4461)  DCS(1,00,1,0,0,0)  BM(0000..00,11..11,00..000..010...0.0.0..0..0...0.0000...0..0000,0...11,000...011,100,011)
17238
17239   4343;  ;(FREE)
17240   LOADDATA730D:
17241           P3,     CSPD[16]_EMIT, EMIT/052525,             ;PATTERN (052525) IN BASCON AREA
17242           NEXT,   J/SETZER730D                            ;
     (4343)  DCS(0,00,0,0,0,0)  BM(0101..)0.01,.01,01..010..101...0.0.0..0..0...0.0001...1..0000,0...11,000...011,100,100)
17243
17244   4344;  ;(FREE)
17245   SETZER730D:
17246           P1,     BA_ASPHI(C000000),                      ;USE MEMORY ADDR(000000)
17247           P2=T,   D_CSPB(B16), D(C)_0,                    ;USE DATA (052525)
17248           P3,     DATO,                                   ;FOR A BUS "DATO" CYCLE
17249           NEXT,   J/MANGLED730D                           ;
     (4344)  DCS(0,00,0,0,0,0)  BM(1010..11,01..11,01..100..000...0.1.0..1..0...1.0010...0..0000,0...11,000...011,100,101)
17250
17251   4345;  ;(FREF)
17252   MANGLED730D:
17253           P3=T,   D_ASPHI(C125252), SAVE=D(C),            ;MANGLE D CONTENTS, AFTER/AT P3=T
17254           NEXT,   J/BUSFCN730D                            ;
     (4345)  DCS(0,00,0,0,0,0)  BM(1111..00,00..11,01..110..111...1.1.0..0..0...0.0000...0..0000,0...11,000...011,100,110)
17255
17256   4346;  ;(FREE)
17257   BUSFCN730D:
17258           P0,     BUMP=VERIFY,                            ;COUNT
17259                   DCS=CTR(C12,),                          ;COMPARE AT TARGET
17260           P1,     BA_ASPHI(C0000001),                     ;ADDRESS ODD BYTE
```



```
17261           P3,     DATIB,                                  ;BYTE READ => PLAIN DATI; NO ODD ADDRESS ERROR
17262           NEXT,   J/GETIT730D                             ;
     (4346)  DCS(0,00,1,0,0,1)  BM(0011..00,00..11,01..000..000...0.0.0..1..0...1.0011...0..0000,0...11,000...011,100,111)
17263
17264   4347;  ;(FREE)
17265   GETIT730D:
17266           P3,     CSPD[17]_BUSDIN, EMIT/000000,           ;UNIBUS DATA SHOULD BE ENABLED; EMIT IS NOISE
17267           NEXT,   J/GOBUT730D                             ;
     (4347)  DCS(0,00,0,0,0,0)  BM(0000..10,00..00,00..000..000...0.0.0..0..0...0.0000...1..0000,0...11,000...011,101,000)
17268
17269   4350;  ;(FREE)
17270   GOBUT730D:
17271           SETUP,  RETURN/TEST730E,                        ;GO TO SUBR WHICH:
17272           NEXT,   CALL(CLR&SERVICE730D]                   ;  (&SERVICE)+XOR=(000340) => D, BUT(D=ZERO)
     (4350)  DCS(0,00,0,0,0,0)  BM(0100..00,10..01,10..010..111...0.0.0...0..0...0.0000...0..0000,0...11,100...010,100,100)
17273
17274
17275
17276
17277
17278
17279
17280   ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
17281
17282
17283   ;TEST-730-E NOW CHECKS THAT THE BUS FUNCTION ABOVE ACTUALLY RETRIEVED THE RIGHT DATA:
17284   ;  THE (052525) WRITTEN TO MEMORY LOCATION (000000) IN TEST730D
17285   4462;
17286   TEST730E:
17287           P0,     LOAD=ENUA(%TARGET402),                  ;SETUP FOR D=ZERO COMPARE
17288                   LOAD=ERROR(TEST730E),                   ;ERROR DIRECTORY KEY
17289                   DCS=CTR(C4,),                           ;COMPARE AT TARGET
17290           NEXT,   J/GOBUT730E                             ;
     (4462)  DCS(1,00,1,0,0,0)  BM(1011..00,11..11,00..000..010...0.0.0..0..0...0.0000...0..0000,0...11,000...011,101,001)
17291
17292   4351;  ;(FREE)
17293   GOBUT730E:
17294           SETUP,  RETURN/SCOPE730,                        ;EXEC SUBR WHICH:
17295           NEXT,   CALL(CSP17XOR052525]                    ;  CSP(17)=XOR=(052525) => D, BUT(D=ZERO)
     (4351)  DCS(0,00,0,0,0,0)  BM(0100..00,01..11,01..010..111...0.0.0...0..0...0.0000...0..0000,0...11,100...010,000,101)
17296
17297
17298   4352;  ;(FREE)
17299   SCOPE730:
17300           NEXT,   RUTD(SCOPE),                            ;NO ERROR:  "TEST731A"  (+1, WORDS)
17301                   J/TEST731A                              ;   ERROR:  "LOADER730A"  (-24, WORDS)

     (4352)  DCS(0,00,0,1,0,0)  BM(0000..00,00..00,00..000..000...0.0.0...0..0...0.0000...0..0000,0...11,000...111,101,101)
17302
17303
17304
17305
17306
```

```
!- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

!THIS SECOND SERIES OF TESTS DOES A DATOB/DATI/DATI-CLKIR SEQUENCE, CHECKING THAT
!EACH FUNCTION OPERATES AS EXPECTED.

! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

!TEST-731-A DOES A DATOB*BYTE*ODD, AND THEN CHECKS THAT THE DBUF LATCH (DS) ALSO GETS LOADED WITH THE
!  DATA, AND THAT IT IS ENABLED ON BUSDIN IN THE MICROWORD AFTER THE BUS CYCLE (IE, EMIT
!  IS TEMPORARILY DISABLED)
4755:
TEST731A:
        P0,     LOAD-ENUA(ZTARGET402),              !SETUP FOR D=ZERO COMPARE
                LOAD-ERROR(TEST731A),               !ERROR DIRECTORY KEY
                DCS-CTR(C6,),                       !COMPARE AT TARGET
        NEXT,   J/BUSFCN731A                        !
   (4755)  DCS[1,00,1,0,0,0]  BM[1001,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,111,110,100]

    4764:
    BUSFCN731A:
        P1,     BA_ASPHI(C000001),                  !USE MEMORY ADDR(000001), ODD BYTE
        P2-T,   D_ZERO, D(C)_ALU15,                 !USE DATA (000), ONE BYTE (ODD) ONLY
        P3,     DATOB,                              !FOR A BUS "DATOB*BYTE*ODD" CYCLE
        NEXT,   J/GETDBUF731A                       !
   (4764)  DCS[0,00,0,0,0,0]  BM[0011,,00,00,,11,01,,000,,100,,,0,1,0,,1,,0,,,1,0101,,,0,,0000,0,,,11,000,,,011,101,011]

    4353:  !(FREE)
    GETDBUF731A:
        P0,     BUMP-VERIFY,                        !COUNT
        P3,     CSPD[17]_BUSDIN,                    !DBUF SHOULD BE ENABLED; EMIT IS NOISE
        P1-T,   D_ASPHI(C177777), SAVE-D[C],        !MANGLE DATA IN D, DONT CARE WHAT RESULTS
        NEXT,   J/COMP731A                          !
   (4353)  DCS[0,00,0,0,0,1]  BM[1111,,10,00,,11,01,,101,,111,,,1,1,0,,0,,0,,,0,0000,,,1,,0000,0,,,11,000,,,011,101,100]

    4354:  !(FREE)
    COMP731A:
        P2-T,   D_CSPD(D17), SAVE-D[C],             !COMPARE RECEIVED:(000000)
        NEXT,   J/GOBUT731A                         !
   (4354)  DCS[0,00,0,0,0,0]  BM[1010,,10,00,,00,00,,000,,111,,,0,1,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,011,101,101]

    4355:  !(FREE)
    GOBUT731A:
        SETUP,  RETURN/TEST731B,                    !RETURN TO START OF NEXT SUBTEST
        NEXT,   GOTO-PAGE(7),                       !BUT TABLE
                J/BUTD-IS-ZERO                      !CHECK FOR EQUALITY
   (4355)  DCS[0,00,0,0,0,0]  BM[0100,,00,10,,01,01,,111,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,100,001]
```



```
! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

!TEST-731-B NOW CHECKS THAT NONE OF THE ABOVE BUS FUNCTIONS HAS ALTERED THE IR FROM WHEN
!  IT WAS LOADED IN TEST730A, WITH THE VALUE (125200), INSTRS-E88(412) DECODE
4457:
TEST731B:
        P0,     LOAD-ENUA(ZTARGET412),              !SETUP FOR INSTRS/E88 DECODE
                LOAD-ERROR(TEST731B),               !ERROR DIRECTORY KEY
                DCS-CTR(C5,),                       !COMPARE AT TARGET
        NEXT,   J/BUSFCN731B                        !
   (4457)  DCS[1,00,1,0,0,0]  BM[1010,,00,11,,11,00,,001,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,011,101,110]

    4356:  !(FREE)
    BUSFCN731B:
        P1,     BA_ASPHI(C000000),                  !USE MEMORY ADDR(000000)
        P3,     DATI,                               !FOR A BUS "DATI" CYCLE
        NEXT,   J/GETIT731B                         !
   (4356)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,11,01,,100,,000,,,0,0,0,,1,,0,,,1,0110,,,0,,0000,0,,,11,000,,,011,101,111]

    4357:  !(FREE)
    GETIT731B:
        P0,     BUMP-VERIFY,                        !COUNT
        P3,     CSPD[17]_BUSDIN, EMIT/152500,       !UNIBUS DATA SHOULD BE ENABLED; EMIT IS NOISE
        NEXT,   J/GOBUT731B                         !  THIS DATA IS INSTRS-E88(405) DECODE
   (4357)  DCS[0,00,0,0,0,1]  BM[1101,,10,01,,01,01,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,1,,0000,0,,,11,000,,,011,110,000]

    4360:  !(FREE)
    GOBUT731B:
        SETUP,  RETURN/TEST731C,                    !RETURN TO START OF NEXT SUBTEST
        NEXT,   GOTO-PAGE(7),                       !BUT TABLE
                J/BUTINSTRS                         !CHECK THAT IR STILL HAS DATA (125200)
   (4360)  DCS[0,00,0,0,0,0]  BM[0100,,00,10,,01,01,,110,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,000,001]

! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

!TEST-731-C NOW CHECKS THAT THE BUS FUNCTION ABOVE ACTUALLY RETRIEVED THE RIGHT DATA:

!  THE (052525) WRITTEN TO MEMORY LOCATION (001)/(000) IN TEST730D
!  AND THE (000) WRITTEN TO MEMORY LOCATION (001) IN TEST731A
!  TOGETHER THESE FORM A (000125) IN MEMORY LOCATION (001)#(000):  INSTRS-E78(432) DECODE
4456:
TEST731C:
        P0,     LOAD-ENUA(ZTARGET432),              !SETUP FOR INSTRS-E78(412) DECODE
                LOAD-ERROR(TEST731C),               !ERROR DIRECTORY KEY
```

```
                        DCS-CTR(C7,);                        ;COMPARE AT TARGET
                        BUMP-VERIFY,                         ;COUNT
                NEXT,   J/COMP731C                           ;
      (4456)  DCS[1,00,1,0,0,1]  BM[1000,,00,11,,11,00,,011,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,011,110,001]

        4361:  ;(FREE)
        COMP731C:
                P2-T,   D_CSPD(D17), D[C]_0,                ;GET DATA READ FROM DATI, ABOVE
                NEXT,   J/GOBUT731C                          ;
      (4361)  DCS[0,00,0,0,0,0]  BM[1010,,10,00,,00,00,,000,,000,,,0,1,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,011,110,010]

        4362:  ;(FREE)
        GOBUT731C:
                SETUP,  RETURN/TEST731D,                     ;RETURN TO START OF NEXT SUBTEST
                NEXT,   CALL[DINTOIR-5]                      ;SUBR FOR:  D -> IR, BUT(INSTR5)
      (4362)  DCS[0,00,0,0,0,0]  BM[0100,,00,10,,01,01,,101,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,111,011]




        ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        ;TEST-731-D DOES A DATI-CLKIR, AND THEN CHECKS THAT THE RIGHT DATA
        ;  IS WRITTEN/RETRIEVED FROM MEMORY LOCATIONS (000000)/(000001)
        ;  AND THAT THE EMIT-DISABLE/UNIBUS DATA BUFFER ENABLE IS HANDLED CORRECTLY
        4455:
        TEST731D:
                P0,     LOAD-ENUA(ZTARGET402),               ;SETUP FOR D=ZERO COMPARE
                        LOAD-ERROR(TEST731D),                ;ERROR DIRECTORY KEY
                        DCS-CTR(C14,),                       ;COMPARE AT TARGET
                NEXT,   J/LOADIR731D                         ;
      (4455)  DCS[1,00,1,0,0,0]  BM[0001,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,011,110,011]

        4363:  ;(FREE)
        LOADIR731D:
                P2-U,   IR_EMIT, EMIT/125200,                ;PREV DATA IN IR FOR INSTR5-E08(412)
                NEXT,   J/BUSFCN731D                         ;
      (4363)  DCS[0,00,0,0,0,0]  BM[1010,,00,10,,10,10,,000,,000,,,0,0,0,,0,,0,,,1,1010,,,0,,0000,0,,,11,000,,,011,110,100]

        4364:  ;(FREE)
        BUSFCN731D:
                P1,     BA_ASPHI(C000000),                   ;FROM LOCATION (000000)
                P3,     DATI-CLKIR,                          ;DO A DATI, AND CLKIR
                NEXT,   J/MANGLED731D                        ;
      (4364)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,11,01,,100,,000,,,0,0,0,,1,,0,,,1,0000,,,0,,0000,0,,,11,000,,,011,110,101]

        4365:  ;(FREE)
        MANGLED731D:
                P2-U,   ;IR_DATA,                            ;IR SHOULD GET DATA HERE
                        EMIT/152500,                         ;NOISE ON EMIT:  INSTR5-E08(405)
                NEXT,   J/GOBUT731D                          ;
      (4365)  DCS[0,00,0,0,0,0]  BM[1101,,00,01,,01,01,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,011,110,110]
```



```
        4366:  ;(FREE)
        GOBUT731D:
                SETUP,  RETURN/TEST731E,                     ;GO TO SUBR WHICH:
                NEXT,   CALL(CLRSERVICETOD)                  ;  (SERVICE)_XOR-(000340) -> D, BUT(D=ZERO)
      (4366)  DCS[0,00,0,0,0,0]  BM[0100,,00,10,,01,01,,100,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,100,100]




        ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        ;TEST-731-E NOW CHECKS THAT THE BUS FUNCTION ABOVE ACTUALLY RETRIEVED THE RIGHT DATA:
        4454:
        TEST731E:
                P0,     LOAD-ENUA(ZTARGET432),               ;SETUP FOR INSTR5-E78(432) DECODE
                        LOAD-ERROR(TEST731E),                ;ERROR DIRECTORY KEY
                        DCS-CTR(C3,);                        ;COMPARE AT TARGET
                NEXT,   J/GOBUT731E                          ;
      (4454)  DCS[1,00,1,0,0,0]  BM[1100,,00,11,,11,00,,011,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,011,110,111]

        4367:  ;(FREE)
        GOBUT731E:
                SETUP,  RETURN/SCOPE731,                     ;RETURN TO SCOPE LOOP TEST WORD
                NEXT,   GOTO-PAGE(7),                        ;BUT TABLE
                        J/BUTINSTR5                          ;CHECK FOR RIGHT DATA
      (4367)  DCS[0,00,0,0,0,0]  BM[0100,,00,01,,11,11,,000,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,000,001]


        4370:  ;(FREE)
        SCOPE731:
                P2-U,   IR_EMIT, EMIT/125200,                ;RESET IR FOR SCOPE LOOP
                NEXT,   AUTO[SCOPE],                         ;NO ERROR:  "LOADIR740A"  (+5, WORDS)
                        J/LOADIR740A                         ;   ERROR:  "BUSFCN731A"  (-18, WORDS)
      (4370)  DCS[0,00,0,1,0,0]  BM[1010,,00,10,,10,10,,000,,000,,,0,0,0,,0,,0,,,1,1010,,,0,,0000,0,,,11,000,,,111,110,101]


        ;       -     -     -     -     -     -     -     -     -     -     -     -

        ;       THESE SUBROUTINES ARE USED IN THE PREVIOUS TESTS:

        7177:  ;(FREE)

        BDX12:
                P3,     CSPD[17]_BUSDIN, EMIT/052525,        ;ENTRY FOR "BUSDINXOR125252"
                NEXT,   J/C17X12                             ;
      (7177)  DCS[0,00,0,0,0,0]  BM[0101,,10,01,,01,01,,010,,101,,,0,0,0,,0,,0,,,0,0000,,,1,,0000,0,,,11,000,,,010,000,011]

        7203:  ;(FREE)
```

```
	C17X12:
		P2-T,	D_CSPD[17]-XOR-ASPHI(C125252),		;ENTRY FOR "CSP17XOR125252"
			SAVE-D[C],				;
		NEXT,	J/BUTD-IS-ZERO				;FINISH OUT BY TESTING RESULT OF COMPARISON
    (7203)  DCS[0,00,0,0,0,0]  BM[0110,,10,00,,11,01,,110,,111,,,0,1,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,011,100,001]

	7204:	;(FREE)
	BDX05:
		P3,	CSPD[17]_BUSDIN, EMIT/052525,		;ENTRY FOR "BUSDINXOR052525"
		NEXT,	J/C17X05				;
    (7204)  DCS[0,00,0,0,0,0]  BM[0101,,10,01,,01,01,,010,,101,,,0,0,0,,0,,0,,,0,0000,,,1,,0000,0,,,11,000,,,010,000,101]

	7205:	;(FREE)
	C17X05:
		P2-T,	D_CSPD[17]-XOR-ASPHI(C052525),		;ENTRY FOR "CSP17XOR052525"
			SAVE-D[C],				;
		NEXT,	J/BUTD-IS-ZERO				;FINISH OUT BY TESTING RESULT OF COMPARISON
    (7205)  DCS[0,00,0,0,0,0]  BM[0110,,10,00,,11,01,,111,,111,,,0,1,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,011,100,001]

	;	-	-	-	-	-	-	-	-	-	-	-	-	-	-

	;.PAGE==========================================================================================

	.TOC	*	TEST740:  BUS CYCLE MODIFICATION - PREFETCH ALTERATION, OVERLAP YANK

	;**********************************************************************************
	;*
	;*	TESTS:	740A - 740D				UWORDS:  020 + 043
	;*
	;*	FUNCTIONS:
	;*
	;*	THESE FOUR TESTS CHECK THAT LOGIC CONCERNED WITH:
	;*
	;*		1) BC0/PREFETCH = BC0[1] * NOT(PREFETCH*BUT[INSTR1])
	;*			IE, ALTER BUS CODE (**1) => (**0), IN A PREFETCH SITUATION
	;*
	;*	AND	2) BEGIN-DATA-XFER = BEGIN[BUS/UCON] * NOT(PULSE-SUPPRESS(0)) * NOT(INH-SPLO)
	;*			IE, 'YANK' (STOP) A BUS CYCLE IN A NON-OVERLAP SITUATION
	;*
	;**********************************************************************************

	; - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
```



```
	;TEST-740-A CHECKS THAT BC(0)-H DOES NOT GET ALTERED FROM (1) => (0) WHEN:
	;  BUTA[INSTR1]-H IS ASSERTED, BUT PREFETCH-H IS NEGATED
	4765:
	LOADIR740A:
		P2-U,	IR_EMIT, EMIT/056000,			;(OVERLAP,*PREFETCH), (-BYTE,DOP,-BMO)
		NEXT,	J/TEST740A				;
    (4765)  DCS[0,00,0,0,0,0]  BM[0101,,00,11,,00,00,,000,,000,,,0,0,0,,0,,0,,,1,1010,,,0,,0000,0,,,11,000,,,111,100,100]

	4744:
	TEST740A:
		P0,	LOAD-ENUA(ZTARGET402),			;SETUP FOR D=ZERO COMPARE
			LOAD-ERROR(TEST740A),			;ERROR DIRECTORY KEY
			DCS-CTR(C12,),				;COMPARE AT TARGET
		NEXT,	J/BUSFCN740A				;
    (4744)  DCS[1,00,1,0,0,0]  BM[0011,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,011,111,001]

	4371:	;(FREE)
	BUSFCN740A:
		P0,	BUMP-VERIFY,				;COUNT
		P1,	BA_ASPHI(C000000),			;USE MEMORY ADDR (000000)
		P2-T,	SP_ZERO,				;SR<00>=(0), FOR UNEXPECTED JAMUPPS
		P3,	DATIB,					;CODE=(3)/DATIB, POSSIBLY ALTERED TO (2)/DATO
			BUTA[INSTR=1],				;ACTIVE EFFECT ONLY - BRANCH MASKED
		NEXT,	J/GOBUT740A				;(056000) TARGETS TO (316), MASKED UNDER (776)
    (4371)  DCS[0,00,0,0,0,1]  BM[0011,,00,00,,11,01,,100,,000,,,0,0,1,,1,,0,,,1,0011,,,0,,0000,0,,,00,110,,,111,111,110]

	4776:
	GOBUT740A:
		SETUP,	RETURN/LOADIR740B,			;GO TO SUBR WHICH:
								;  CLR-JAM-ERRORS, FOR INSURANCE
		NEXT,	CALL[DATISERVICEYDD]			;  (SERVICE)-XOR-(100340) => D, BUT(D=ZERO)
    (4776)  DCS[0,00,0,0,0,0]  BM[0100,,00,01,,11,11,,010,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,100,101]

	; - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

	;TEST-740-B CHECKS THAT BC[0]-H DOES GET ALTERED FROM (1) => (0) WHEN:
	;  BUTA[INSTR1]-H IS ASSERTED, AND PREFETCH-H IS ASSERTED
	4372:	;(FREE)
	LOADIR740B:
		P2-U,	IR_EMIT, EMIT/020606,			;(OVERLAP,PREFETCH), (-BYTE,DOP,SMO)
		NEXT,	J/TEST740B				;
    (4372)  DCS[0,00,0,0,0,0]  BM[0010,,00,00,,01,10,,000,,110,,,0,0,0,,0,,0,,,1,1010,,,0,,0000,0,,,11,000,,,100,101,011]

	4453:
	TEST740B:
		P0,	LOAD-ENUA(ZTARGET402),			;SETUP FOR D=ZERO COMPARE
			LOAD-ERROR(TEST740B),			;ERROR DIRECTORY KEY
			DCS-CTR(C12,),				;COMPARE AT TARGET
		NEXT,	J/BUSFCN740B				;
```

```
    (4453)  DCS[1,00,1,0,0,0]  BM[0011,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,011,111,011]

        4373:  ;(FREE)
        BUSFCN740B:
                P1,      BA_ASPHI(C000000),                 ;USE MEMORY ADDR (000000)
                P3,      DATIB,                             ;CODE=(3)/DATIB, POSSIBLY ALTERED TO (2)/DATO
                         BUTA[INSTR=1],                     ;ACTIVE EFFECT ONLY = BRANCH MASKED
                NEXT,    J/GOBUT740B                        ;(020606) TARGETS TO (042), MASKED UNDER (766)
    (4373)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,11,01,,100,,000,,,0,0,0,,1,,0,,,1,0011,,,0,,0000,0,,,00,110,,,111,110,110]

        4766:
        GOBUT740B:
                SETUP,   RETURN/TEST740C,                   ;GO TO SUBR WHICH:
                                                            ;  CLR-JAM-ERRORS, FOR INSURANCE
                NEXT,    CALL[DATOSERVICETOD]               ;  (SERVICE)-XOR-(002340) => D, BUT(D=ZERO)
    (4766)  DCS[0,00,0,0,0,0]  BM[0100,,00,10,,01,01,,010,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,100,110]





        ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        ;TEST-740-C CHECKS THAT BC[0]-H DOES NOT GET ALTERED FROM (1) => (0) WHEN:
        ;  BUTA[INSTR1]-H IS NEGATED, BUT PREFETCH-H IS ASSERTED
        4452:
        TEST740C:
                P0,      LOAD-ENUA(ZTARGET402),             ;SETUP FOR D=ZERO COMPARE
                         LOAD-ERROR(TEST740C),              ;ERROR DIRECTORY KEY
                         DCS-CTR(C12,),                     ;COMPARE AT TARGET
                         BUMP-VERIFY,                       ;COUNT
                NEXT,    J/BUSFCN740C                       ;IR AS ABOVE;  (OVERLAP,PREFETCH), (-BYTE,DOP,SNO)
    (4452)  DCS[1,00,1,0,0,1]  BM[0011,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,011,111,100]

        4374:  ;(FREE)
        BUSFCN740C:
                P1,      BA_ASPHI(C000000),                 ;USE MEMORY ADDR (000000)
                P3,      DATIB,                             ;CODE=(3)/DATIB, POSSIBLY ALTERED TO (2)/DATO
                NEXT,    J/GOBUT740C                        ;(020606) TARGETS TO (042), MASKED UNDER (766)
    (4374)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,11,01,,100,,000,,,0,0,0,,1,,0,,,1,0011,,,0,,0000,0,,,11,000,,,011,111,101]

        4375:  ;(FREE)
        GOBUT740C:
                SETUP,   RETURN/LOADIR740D,                 ;GO TO SUBR WHICH:
                                                            ;  CLR-JAM-ERRORS, FOR INSURANCE
                NEXT,    CALL[DATISERVICETOD]               ;  (SERVICE)-XOR-(100340) => D, BUT(D=ZERO)
    (4375)  DCS[0,00,0,0,0,0]  BM[0100,,00,10,,00,00,,000,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,100,101]
```



```
        ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        ;TEST-740-D CHECKS THAT WHEN BUTA[INSTR1]-H IS ASSERTED, AND OVERLAP-L IS
        ;  NEGATED, THEN THE BUS CYCLE IS NOT EVEN ALLOWED TO BEGIN
        4400:  ;(FREE)
        LOADIR740D:
                P0,      BUMP-VERIFY,                       ;COUNT
                P2-0,    IR_EMIT, EMIT/076000,              ;(-OVERLAP,-PREFETCH), (-BYTE,-DOP,-SOP)
                NEXT,    J/TEST740D                         ;
    (4400)  DCS[0,00,0,0,0,1]  BM[0111,,00,11,,00,00,,000,,000,,,0,0,0,,0,,0,,,1,1010,,,0,,0000,0,,,11,000,,,100,101,001]

        4451:
        TEST740D:
                P0,      LOAD-ENUA(ZTARGET402),             ;SETUP FOR D=ZERO COMPARE
                         LOAD-ERROR(TEST740D),              ;ERROR DIRECTORY KEY
                         DCS-CTR(C11,),                     ;COMPARE AT TARGET
                NEXT,    J/BUSFCN740D                       ;
    (4451)  DCS[1,00,1,0,0,0]  BM[0100,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,100,000,001]

        4401:  ;(FREE)
        BUSFCN740D:
                P1,      BA_ASPHI(C000001),                 ;USE MEMORY ADDR (000001), TRY TO FORCE ODD ADDR
                P3,      DATO,                              ;TRY TO ALTER SERVICE FROM (100340)/DATI TO (002340)/DATO
                         BUTA(INSTR=1),                     ;ACTIVE EFFECT ONLY = BRANCH MASKED
                NEXT,    J/GOBUT740D                        ;(076000) TARGETS TO (047), MASKED UNDER (767)
    (4401)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,11,01,,000,,000,,,0,0,0,,1,,0,,,1,0010,,,0,,0000,0,,,00,110,,,111,110,111]

        4767:
        GOBUT740D:
                SETUP,   RETURN/SCOPE740,                   ;GO TO SUBR WHICH:
                                                            ;  (CSP(02) LOADED IN LAST TEST; SAME VALUE= (100340))
                NEXT,    CALL[CJESERVICETOD]                ;  (SERVICE)-XOR-CSP(02) => D, BUT(D=ZERO)
    (4767)  DCS[0,00,0,0,0,0]  BM[0100,,00,10,,00,00,,010,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,100,111]



        4402:  ;(FREE)
        SCOPE740:
                P2-0,    IR_EMIT, EMIT/056000,              ;RELOAD IR FOR TEST740A
                NEXT,    BUTD[SCOPE],                       ;NO ERROR:  "TEST761A"  (+1, WORDS)
                         J/TEST761A                         ;   ERROR:  "LOADIR740A"  (-16, WORDS)
    (4402)  DCS[0,00,0,1,0,0]  BM[0101,,00,11,,00,00,,000,,000,,,0,0,0,,0,,0,,,1,1010,,,0,,0000,0,,,11,000,,,111,100,101]




        ;.PAGE=========================================================================================

        .TOC    *    TEST761-763:  TESTING UNIBUS INTERRUPT SERVICE WITH DL11-W LINE CLOCK

        ;*****************************************************************************
```

```
;*
;*      TESTS:  761A - 763D                                       UWORDS: 060 + 045
;*
;*      FUNCTIONS:        TESTS 761A - 763D CAUSE AN INTERRUPT ON THE UNIBUS AT
;*                          LEVEL BR6, USING THE DL11-W LINE CLOCK AS A GENERATOR,
;*                          THEN TEST TO SEE THAT ALL THE APPROPRIATE BELLS AND
;*                          WHISTLES AND PARAPHERNALEALIAISOOURE ALSO RESPOND.
;*************************************************************************************************






; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

;THIS FIRST SET OF THREE TESTS CLEAR OUT ALL THE I-O UCON REGISTERS;
;CLR-JAM-ERRORS, CLR-YELLOW-ZONE, CLEAR-CONSOLE-SERVICE, CLR-NPR-TIMEOUT, CLR-PWR-FAIL, AND
;ALSO THIS TIME DO AN "INIT" ON THE UNIBUS, VIA THE BUS-INIT-UCON FUNCTION,
;ALSO SET ALL THE FLAGS<8:0>H=ZERO, AND THE FULL PS<15:00>H=ZERO,
;(IE, PROCESSOR PRIO=0, T-BIT=0),
;
;THEN CHECK TO SEE THAT ALL THE RELEVANT BUS CONTROL INTERRUPT LOGIC IS RESET.





; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

;TEST 761A CHECKS THAT SERVICE-H=NOT(INTR-HIGH-H*FLAG7(0)H*BG-SERVICE(0)H) IS LOW,
;  WHEN INTR-HIGH-H=HIGH, FLAG7(0)H=HIGH, AND BG-SERVICE(0)H=HIGH
;  IE, AFTER UNIBUS-INIT, AND CLEAR SERVICE CONDITIONS, W/ PSW PRIO=000,
;  THERE SHOULD BE NOTHING PENDING
4745:
TEST761A:
        P0,     LOAD-ENUA(ZTARGET402),                  ;BIT<00> CLEAR
                LOAD-ERROR(TEST761A),                   ;ERROR DIRECTORY KEY
                DCS-CTR(C12,),                          ;COMPARE AT TARGET
        P3,     BUTA(CLR-FLAG-RES-UCON),                ;SET SR/LOAD, BUSDIN/EMIT
        NEXT,   J/ZERODSR761A                           ;
   (4745)  DCS[1,00,1,0,0,0]  BN[0011,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,010,,,111,111,010]

4772:
ZERODSR761A:
        P2-T,   D_ZERO, SR_ZERO, D[C]_ALU15,            ;ZERO FOR FLAGS/PS/JAMUPP FLAG
        NEXT,   J/ZEROIT761A                            ;
   (4772)  DCS[0,00,0,0,0,0]  BN[0011,,00,00,,00,00,,000,,100,,,0,1,1,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,100,000,011]

4403:  ;(FREE)
ZEROIT761A:
        P0,     BUMP-VERIFY,                            ;COUNT
                BUSDIN_EMIT-[1],                        ;KEEP IT ON
```



```
        P3,     FLAG[8-0]_D[15-8]-[1],                  ;ZERO THE FLAGS
                PS_D-[1],                               ;ZERO ALL OF THE PS
        NEXT,   J/CLEAR761A                             ;
   (4403)  DCS[0,00,0,0,0,1]  BN[1000,,00,00,,00,01,,010,,011,,,0,0,0,,0,,0,,,1,1011,,,0,,0000,0,,,11,000,,,100,000,100]

4404:  ;(FREE)
CLEAR761A:
        SETUP,  RETURN/GOBUT761A,                       ;GO TO SUBR WHICH DOES THE CLEARS AND BUS-INIT
        NEXT,   CALL[CLEAR-I-O-A]                       ;
   (4404)  DCS[0,00,0,0,0,0]  BN[0100,,00,10,,00,00,,101,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,010,111]

4405:  ;(FREE)
GOBUT761A:
        SETUP,  RETURN/TEST761B,                        ;RETURN TO START OF NEXT SUBTEST
        NEXT,   GOTO-PAGE(7),                           ;BUT TABLE
                J/BUTSERVICE                            ;SERVICE-H IN BIT<00>
   (4405)  DCS[0,00,0,0,0,0]  BN[0100,,00,11,,11,11,,001,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,100,101]





; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

;TEST 761B CHECKS THAT VECTOR-LOAD[1]H=-(UNIBUS-INTR-L) IS LOW
4771:
TEST761B:
        P0,     LOAD-ENUA(ZTARGET401),                  ;BIT<01> CLEAR
                LOAD-ERROR(TEST761B),                   ;ERROR DIRECTORY KEY
                DCS-CTR(C3,),                           ;COMPARE AT TARGET
                BUMP-VERIFY,                            ;COUNT
        NEXT,   J/GOBUT761B                             ;
   (4771)  DCS[1,00,1,0,0,1]  BN[1100,,00,11,,11,00,,000,,001,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,100,000,110]

4406:  ;(FREE)
GOBUT761B:
        SETUP,  RETURN/TEST761C,                        ;RETURN TO START OF NEXT SUBTEST
        NEXT,   GOTO-PAGE(7),                           ;BUT TABLE
                J/BUTVECTLOAD                           ;VECTOR-LOAD[1]H IN BIT<01>
   (4406)  DCS[0,00,0,0,0,0]  BN[0100,,00,11,,11,11,,100,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,100,110]





; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -


;TEST 761C CHECKS THAT BG-SERVICE(0)H=BR>PS-L IS HIGH WHEN NO DEVICES REQUEST INTR ON UNIBUS
4774:

TEST761C:
        P0,     LOAD-ENUA(ZTARGET407),                  ;BIT<02> SET  (ACTIVE LOW)
```

```
                LOAD-ERROR(TEST761C),                   ;ERROR DIRECTORY KEY
                DCS-CTR(C3,),                           ;COMPARE AT TARGET
        NEXT,   J/GOBUT761C                             ;
 (4774) DCS[1,00,1,0,0,0] BM[1100,,00,11,,11,00,,000,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,100,000,111]

   4407: ;(FREE)
   GOBUT761C:
        SETUP,  RETURN/SCOPE761,                        ;RETURN TO SCOPE LOOP TEST WORD
        NEXT,   GOTO-PAGE(7),                           ;BUT TABLE
                J/BUTBOSERVL                            ;SC-SERVICE[0]H IN BIT<03>
 (4407) DCS[0,00,0,0,0,0] BM[0100,,00,10,,00,01,,000,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,001,110]



   4410: ;(FREE)
   SCOPE761:
        P0,     BUMP-VERIFY,                            ;COUNT
        P3,     CSPD[05]_EMIT, EMIT/177546,             ;(177546) IS UNIBUS ADDR OF CSR FOR DL11-W
        NEXT,   BUTD[SCOPE],                            ;NO ERROR: "TEST762A" (+1, WORDS)
                J/TEST762A                              ;   ERROR: "ERRODSR761A" (-8, WORDS)
 (4410) DCS[0,00,0,1,0,1] BM[1111,,10,11,,11,01,,100,,110,,,0,0,0,,0,,0,,,0,1010,,,1,,0000,0,,,11,000,,,111,111,011]





   ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

   ;THE FOLLOWING FIVE TESTS NOW CAUSE AN INTERRUPT ON THE UNIBUS, AND THEN
   ;CHECK THAT THE BUS CONTROL LOGIC RESPONDS CORRECTLY.
   ;FIRST THE LINE CLOCK INTR ENABLE BIT<06> IS SET, AND THE PROCESSOR PRIORITY IS SET TO (6).
   ;SINCE THE DL11-W IS A BR6 DEVICE, THE INTERRUPT SHOULD NOT COME THRU UNTIL THE PRIORITY IS
   ;LOWERED TO LEVEL (5) OR LOWER.  NOTE THAT THE MICROCODE MUST GO IN TO A WAIT LOOP FOR A
   ;MINIMUM OF 16.7 MILLISEC @ 60 HZ, OR 20 MILLISEC @ 50 HZ TO GUARANTEE THAT AN INTERRUPT WILL
   ;BE PENDING.  THE DELAY, BASED UPON  UP3-UP3 = 170 NS, WILL BE 22.3 MILLISEC; AN ADEQUATE MARGIN.





   ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

   ;TEST 762A ENABLES THE INTERRUPT, THEN WAITS FOR THE DELAY PERIOD.
   4773:
   TEST762A:
        P0,     LOAD-EMUA[ZAPD762A],                    ;COMPARE JUST AFTER BUS CYCLE INVOKED
                LOAD-ERROR(TEST762A),                   ;ERROR DIRECTORY KEY
                DCS-CTR(C5,),                           ;COMPARE JUST AFTER "DATO" INITIATED, BELOW
        NEXT,   J/DW11L762A                             ;
 (4773) DCS[1,00,1,0,0,0] BM[1010,,00,10,,00,11,,001,,101,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,100,001,001]

   4411: ;(FREE)
   DW11L762A:
```



```
        P2-T,   SR_CSPD(D05),                           ;GET SR=(177546), ADDR OF DL11-W CSR
        NEXT,   GOTO-PAGE(7),                           ;XFER
                J/MASK762A                              ;
 (4411) DCS[0,00,0,0,0,0] BM[1010,,10,00,,00,00,,000,,111,,,0,0,1,,0,,0,,,0,1010,,,0,,0000,0,,,11,100,,,010,000,010]

   7202: ;(FREE)
   MASK762A:
        P3,     CSPD(04)_EMIT, EMIT/177777,             ;READ ALL BITS IN REGISTERS READ
        NEXT,   J/PRI06762A                             ;
 (7202) DCS[0,00,0,0,0,0] BM[1111,,10,11,,11,11,,111,,111,,,0,0,0,,0,,0,,,0,1011,,,1,,0000,0,,,11,000,,,010,000,111]

   7207: ;(FREE)
   PRI06762A:
        P3,     CSPD(15)_EMIT,                          ;EMITCON FOR;  PS<7:5>="110"=(6), T-BIT=0
                EMIT/000300,                            ;  AND BIT<06> SET FOR DL11-W INTR ENABLE
        NEXT,   J/GETIT762A                             ;
 (7207) DCS[0,00,0,0,0,0] BM[0000,,10,00,,00,11,,000,,000,,,0,0,0,,0,,0,,,0,0010,,,1,,0000,0,,,11,000,,,010,001,000]

   7210: ;(FREE)
   GETIT762A:
        P1,     BA_SR,                                  ;BA <- (177546), BITS<17:16> FORCED TO "11" ON IO-PAGE ADDRESS
        P2-T,   D_CSPD(D15),                            ;(000300) INTO D  ** NOTE;  BIT<1:0> = 00 FOR BA LOAD **
        P3,     DATO,                                   ;WRITE IT OUT
        NEXT,   GOTO-PAGE(4),                           ;XFER FOR DCS XTN BITS
                J/ZAPD762A                              ;
 (7210) DCS[0,00,0,0,0,0] BM[1010,,11,10,,00,00,,000,,100,,,0,1,0,,1,,0,,,1,0010,,,0,,0000,0,,,11,100,,,011,001,101]

   4315: ;(FREE)
   ZAPD762A:
        P3-T,   D_ZERO, D[C]_ALU15,                     ;ZERO D, D[C] FOR LOOP; MUST DO AFTER P3-T
        NEXT,   J/NEXTD762A                             ;ENTER AT 2ND WORD OF LOOP; LET D SETTLE
 (4315) DCS[0,00,0,0,0,0] BM[0011,,00,00,,00,00,,000,,100,,,1,1,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,100,110,111]

   ;***  THE FOLLOWING TWO WORDS NOW GO INTO A COUNT LOOP TO WAIT FOR THE LINE CLOCK TO INTERRUPT
   ;***  THE WAIT WILL BE A MAXIMUM DELAY, DEPENDING UPON THE PROCESSOR UP3-UP3 CYCLE TIME:
   ;***
   ;***           PROCESSOR        TIME DELAY
   ;***           CYCLE TIME       (MILLISEC)
   ;***           (NANOSEC)
   ;***           -----------      -----------
   ;***              150              19.7
   ;***              160              21.0
   ;***            * 170 *          * 22.3 *        <NOMINAL VALUE>
   ;***              180              23.6
   ;***              190              24.9
   ;***

   ;***  NOTE THAT THE ABOVE LOOP TIME IS A MAXIMUM VALUE; WE WILL EXIT EARLY IF THE INTERRUPT COMES
   ;***  THROUGH BEFORE WE OVERFLOW THE COUNTER.  IF NO INTERRUPT COMES THROUGH BY THE TIME THE COUNT
   ;***  HAS OVERFLOWN, IT WILL BE CONSIDERED AN ERROR.
   ;***

   ;* ENTER HERE FOR ANOTHER LOOP *
```

```
4551:
BUMPD762A:
        P3-T,   D_D-PLUS-1, D[C]_COUT15,                ;BUMP D, SAVE CARRYOUT
        NEXT,   BUTR(BG-SERVCE-L),                      ;NEGATED:  "NEXTD762A"
                J/SETPR6-762A                           ;ASSERTED: "SETPR6-762A"
 (4551)  DCS[0,00,0,0,0,0]  BM[1001..01,11..01,01..000..110...1,1,0..0..0...0,0000...0..0000,0...01,100...100,110,011]

4467:
NEXTD762A:
        P0,     DCS-CTR(C15.),                          ;STALL; NOTE: NO BUMP-VERIFIES IN THIS LOOP
        NEXT,   BUTR(D[C]=B),                           ;SET:    "TEST762A1"  D OVERFLOWN, ERROR
                J/BUMPD762A                             ;CLEAR:  "BUMPD762A"  GO FOR NEXT LOOP
 (4467)  DCS[0,00,1,0,0,0]  BM[0000..00,00..00,00..000..000...0,0,0..0..0...0,0000...0..0000,0...10,011...101,101,001]


;* COME HERE IF D OVERFLOWN *
4553:
TEST762A1:
        P0,     LOAD-ERROR(TEST762A1),                  ;ERROR DIRECTORY KEY
                DCS-CTR(C0.),                           ;SIGNAL ERROR NOW
        NEXT,   J/TEST762A                              ;FORCE A SCOPE LOOP
 (4553)  DCS[1,00,1,0,0,0]  BM[1111..00,00..00,00..000..000...0,0,0..0..0...0,0000...0..0000,0...11,000...111,111,011]


;*** COME HERE IF EXIT LOOP OK:  D NOT OVERFLOWED, INTERRUPT PENDING ***
4463:
SETPR6-762A:
        P2-T,   D_CSPD(D15), D[C]_0,                    ;GET D<7:5>=(6), D<4>=(0)
        NEXT,   J/SETPRI762A                            ;
 (4463)  DCS[0,00,0,0,0,0]  BM[1010..10,00..00,00..000..000...0,1,0..0..0...0,0010...0..0000,0...11,000...100,001,011]

4413:  ;(FREE)
SETPRI762A:
        P0,     BUSDIN_EXIT-[I],                        ;KEEP IT ON
        P2,     PS[3-0]_D[3-0]-[I],                     ;FOR USE IN TEST763A, PS[CC]="0000" HERE
        P3,     PS[7-4]_D[7-4]-[I],                     ;PRIO=6, T-BIT=0
        NEXT,   J/TEST762B                              ;NOTE:  THE BR6 PENDING INTR SHOULD NOW HIDE UNDER PROCESSOR PRIO(6)
 (4413)  DCS[0,00,0,0,0,0]  BM[1000..00,00..00,01..010..000...0,0,0..0..0...1,1011...0..0000,0...11,000...111,101,001]


;  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -

;TEST 762B CHECKS THAT BG-SERVICE(0)H+BR>PS-L IS HIGH WHEN THE PROCESSOR PRIORITY(=6) IS AS HIGH
;AS THE ONLY DEVICE WISHING TO REQUEST AN INTERRUPT (AT BR6).
4751:
TEST762B:
        P0,     LOAD-ENUA(ZTARGET407),                  ;BIT<02> SET  (ACTIVE LOW)
                LOAD-ERROR(TEST762B),                   ;ERROR DIRECTORY KEY
                DCS-CTR(C3.),                           ;COMPARE AT TARGET
                BUMP-VERIFY,                            ;
        NEXT,   J/GOBUT762B                             ;
 (4751)  DCS[1,00,1,0,0,1]  BM[1100..00,11..11,00..000..111...0,0,0..0..0...0,0000...0..0000,0...11,000...111,110,010]

4762:
```



```
GOBUT762B:
        SETUP,  RETURN/TEST762C,                        ;RETURN TO START OF NEXT SUBTEST
        NEXT,   GOTO-PAGE(7),                           ;BUT TABLE
                J/BUTBGSERVL                            ;BG-SERVICE[0]H IN BIT<02>
 (4762)  DCS[0,00,0,0,0,0]  BM[0100..00,11..11,00..000..111...0,0,0..0..0...0,0000...0..0000,0...11,100...011,001,110]





;  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -

;TEST 762C CHECKS THAT BG-SERVICE[0]H+BR>PS-L IS LOW WHEN A BR6 DEVICE (DL11-W) IS REQUESTING AN INTR,
;AND THE PROCESSOR PRIORITY(=5) IS < THE BR LEVEL(=6).  NO OTHER DEVICES PRESENT.
4740:
TEST762C:
        P0,     LOAD-ENUA(ZTARGET403),                  ;BIT<02> CLEAR  (ACTIVE LOW)
                LOAD-ERROR(TEST762C),                   ;ERROR DIRECTORY KEY
                DCS-CTR(C7.),                           ;COMPARE AT TARGET
        NEXT,   J/FILL762C                              ;
 (4740)  DCS[1,00,1,0,0,0]  BM[1000..00,11..11,00..000..011...0,0,0..0..0...0,0000...0..0000,0...11,000...100,001,100]

4414:  ;(FREE)
FILL762C:
        P2-T,   D_ASPHI(C125252), D[C]_ALU00,           ;BIT<7:5>="101"=(5), BIT<4>=0
        NEXT,   J/DELAY762C                             ;
 (4414)  DCS[0,00,0,0,0,0]  BM[1111..00,00..11,01..110..010...0,1,0..0..0...0,0000...0..0000,0...11,000...100,001,101]

4415:  ;(FREE)
DELAY762C:
        SETUP,  RETURN/SETPRI762C,                      ;EXEC 3. UWORDS AFTER SETTING PRIO, FOR DELAY
        P2,     PS[3-0]_D[3-0],                         ;FOR USE IN TEST763A, PS[CC]="1010"
        P3,     PS[7-4]_D[7-4],                         ;SET PRIO=(5), AT P3-T OF THIS UWORD
        NEXT,   GOTO-PAGE(7),                           ;GO DO A JUMP, AND A BUTA(RETURN)
                J/BUTD-IS-ZERO                          ;DON'T REALLY CARE ABOUT THE RESULT OF THIS
 (4415)  DCS[0,00,0,0,0,0]  BM[0111..00,01..00,00..110..111...0,0,0..0..0...1,1010...0..0000,0...11,100...011,100,001]

7206:  ;(FREE)
SETPRI762C:
                                                        ;BG-SERVICE(0)H SHOULD BE ASSERTED BY NOW
        SETUP,  RETURN/TEST762D,                        ;RETURN TO START OF NEXT SUBTEST
        NEXT,   GOTO-PAGE(7),                           ;BUT TABLE
                J/BUTBGSERVL                            ;BG-SERVICE(0)H IN BIT<02>
 (7206)  DCS[0,00,0,0,0,0]  BM[0100..00,11..11,00..001..111...0,0,0..0..0...0,0000...0..0000,0...11,100...011,001,110]






;  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -

;TEST 762D CHECKS THAT BG-SERVICE[1]H+FLTPT-SERVICE-H IS HIGH, WHEN BG-SERVICE[1]H IS SET
```

```
        4741:
        TEST762D:
                P0,     LOAD-ENUA(ZTARGET417),                          ;BIT<03> SET
                        LOAD-ERROR(TEST762D),                           ;ERROR DIRECTORY KEY
                        DCS-CTR(C3,),                                   ;COMPARE AT TARGET
                NEXT,   J/GOBUT762D                                     ;
    (4741)  DCS[1,00,1,0,0,0]  BM[1100,,00,11,,11,00,,001,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,100,001,110]

        4416:  ;(FREE)
        GOBUT762D:
                SETUP,  RETURN/TEST762E,                                ;RETURN TO START OF NEXT SUBTEST
                NEXT,   GOTO-PAGE(7),                                   ;BUT TABLE
                        J/BUTBGFPSERV                                   ;BG-SERVICE(1)H+FLTPT-SERVICE-H IN BIT<03>
    (4416)  DCS[0,00,0,0,0,0]  BM[0100,,00,11,,10,11,,000,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,000,111]





        ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        ;TEST 762E NOW READS THE "SERVICE" PORT OF THE STATUS MUX TO SEE:
        ;SERVICE<15:00>H = "0 100 011 111 100 000"
        ;
        ;       IMPORTANT BITS ARE:
        ;       B15 = DATI(1)H = 0              B14 = BG-SERVICE(1)H = 1        B11 = DATOB(1)H = 0
        ;       B10 = DATO(1)H = 1              B09 = PBA<17>H = 1              B08 = PBA<16>H = 1
        ;
        4730:
        TEST762E:
                P0,     LOAD-ENUA(ZTARGET402),                          ;SETUP FOR D=ZERO TEST
                        LOAD-ERROR(TEST762E),                           ;ERROR DIRECTORY KEY
                        DCS-CTR(C10,),                                  ;COMPARE AT TARGET
                NEXT,   J/EXPEC762E                                     ;
    (4730)  DCS[1,00,1,0,0,0]  BM[0101,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,100,001,111]

        4417:  ;(FREE)
        EXPEC762E:
                P3,     CSPD[02]_EMIT,                                  ;EXPECTED VALUE OUT OF SERVICE PORT:
                        EMIT/043740,                                    ;"0100 0111 1110 0000"
                NEXT,   J/GOGET762E                                     ;
    (4417)  DCS[0,00,0,0,0,0]  BM[0100,,10,01,,11,11,,100,,000,,,0,0,0,,0,,0,,,0,1101,,,1,,0000,0,,,11,000,,,100,010,000]

        4420:  ;(FREE)
        GOGET762E:
                P0,     BUMP-VERIFY,                                    ;
                SETUP,  RETURN/TEST762F,                                ;GO TO SUBR WHICH:
                NEXT,   CALL[SERVICETOD]                                ;CMP(02),XOR,SERVICE -> D, BUT(D=ZERO)
    (4420)  DCS[0,00,0,0,0,1]  BM[0100,,00,11,,10,01,,001,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,101,000]
```



```
        ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        ;TEST 762F CHECKS THAT SERVICE-H=NOT(INTR-HIGH-H*FLAG7(0)H*BG-SERVICE(0)H) IS HIGH,
        ;  WHEN INTR-HIGH-H=HIGH, FLAG7(0)H=HIGH, AND BG-SERVICE(0)H=LOW
        4711:
        TEST762F:
                P0,     LOAD-ENUA(ZTARGET403),                          ;BIT<00> SET
                        LOAD-ERROR(TEST762F),                           ;ERROR DIRECTORY KEY
                        DCS-CTR(C3,),                                   ;COMPARE AT TARGET
                NEXT,   J/GOBUT762F                                     ;
    (4711)  DCS[1,00,1,0,0,0]  BM[1100,,00,11,,11,00,,000,,011,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,100,010,001]

        4421:  ;(FREE)
        GOBUT762F:
                SETUP,  RETURN/TEST763A,                                ;RETURN TO START OF NEXT SUBTEST
                NEXT,   GOTO-PAGE(7),                                   ;BUT TABLE
                        J/BUTSERVICE                                    ;SERVICE-H IN BIT<00>
    (4421)  DCS[0,00,0,0,0,0]  BM[0100,,00,11,,11,10,,001,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,100,101]





        ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        ;THIS NEXT SEQUENCE OF TWO TESTS RESPONDS TO THE BUS INTERRUPT REQUEST BY:
        ;(1) ASSERTING "ALLOW-BG(1)H", THUS ALLOWING THE BUS GRANT TO THE DL11-W, SO THAT
        ;(2) "VECTOR-LOAD[1]H" WILL BE ASSERTED, INDICATING THAT THE DEVICE HAS PUT ITS
        ;       VECTOR ON UNIBUS DATA<8:0>L, AND THEN
        ;(3) ACTUALLY READING THE VECTOR FOR THE DL11-W (100)(8), AND VALIDATING ITS CORRECTNESS.



        ; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        ;TEST 763A CHECKS THAT AFTER "ALLOW-BG[1]H" IS GIVEN TO THE INTERRUPTING DEVICE,
        ;THEN "VECTOR-LOAD[1]H" IS ASSERTED
        4761:
        TEST763A:
                P0,     LOAD-ENUA(VECTLOAD763A),                        ;COMPARE AT "VECTOR-LOAD[1]H" IN BIT<01> SET
                        LOAD-ERROR(TEST763A),                           ;ERROR DIRECTORY KEY
                        DCS-CTR(C5,),                                   ;COMPARE AT TARGET
                NEXT,   J/EXPEC763A                                     ;
    (4761)  DCS[1,00,1,0,0,0]  BM[1010,,00,10,,01,11,,011,,011,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,100,010,010]

        4422:  ;(FREE)

        EXPEC763A:
                P0,     BUMP-VERIFY,                                    ;COUNT
                P3,     CSPD[01]_EMIT, EMIT/100,                        ;DL11-W VECTOR IS (100)8
                NEXT,   J/ALLOW763A                                     ;
    (4422)  DCS[0,00,0,0,0,1]  BM[0000,,10,00,,00,01,,000,,000,,,0,0,0,,0,,0,,,0,1110,,,1,,0000,0,,,11,000,,,111,011,001]

        4731:                                                           ;*** *** *01
```

```
ALLOW763A:
        P0,     BUSDIN_PS-[I],                          !NOISE BITS ON BUSDIN TO IMPEDE READING VECTOR
                                                        !AT THIS POINT PS=(000252)
        P2,     ALLOW-BG[1]H-[I],                       !GIVE BUS GRANT TO PENDING INTERRUPT
        NEXT,   J/READVECT763A                          !
   (4731)  DCS[0,00,0,0,0,0]  BM[0100,,00,01,,10,01,,000,,000,,,0,0,0,,0,,0,,,1,1011,,,0,,0000,0,,,11,000,,,100,010,011]

   4423:  !(FREE)
   READVECT763A:
        P3,     CSPD[07]_BUSDIN,                        !MUST READ BUSDIN=VECTOR RITE HERE, OR
                                                        !WE WILL LOSE IT
                EMIT/052525,                            !NOISE ON EMIT
        NEXT,   J/TESTVECT763A                          !
   (4423)  DCS[0,00,0,0,0,0]  BM[0101,,10,01,,01,01,,010,,101,,,0,0,0,,0,,0,,,0,1000,,,1,,0000,0,,,11,000,,,100,010,100]

   4424:  !(FREE)
   TESTVECT763A:
        P2-T,   SR_CSPD(D01),                           !SR=(000100), EXPECTED VECTOR
        NEXT,   BUTR(VECTOR-LOAD),                      !IF SET, "VECTLOAD763A", THIS IS EXPECTED
                J/ALLOW763A                             !IF CLR, "ALLOW763A", VECTOR-LOAD[1]H NOT SET, ERROR
   (4424)  DCS[0,00,0,0,0,0]  BM[1010,,10,00,,00,00,,000,,000,,,0,0,1,,0,,0,,,0,1110,,,0,,0000,0,,,10,001,,,111,011,001]




   ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

   !IF VECTOR-LOAD[1]H="1", COME TO HERE.
   !
   !TEST 763B NOW CHECKS THAT THE CORRECT VECTOR FOR THE DL11-W = 100(8) WAS READ
   4733:                                                !*** *** *11
   VECTLOAD763A:
        P2-T,   D_SR-XOR-CSPD(D07),                     !COMPARE EXPECTED VECTOR:RECEIVED VECTOR
        NEXT,   J/TEST763B                              !
   (4733)  DCS[0,00,0,0,0,0]  BM[0110,,10,00,,00,00,,000,,000,,,0,1,0,,0,,0,,,0,1000,,,0,,0000,0,,,11,000,,,111,010,000]

   4720:
   TEST763B:
        P0,     LOAD-ENUA(ZTARGET402),                  !SETUP FOR D=ZERO TEST
                LOAD-ERROR(TEST763B),                   !ERROR DIRECTORY KEY
                DCS-CTR(C3,1),                          !COMPARE AT TARGET
        NEXT,   J/GOBUT763B                             !
   (4720)  DCS[1,00,1,0,0,0]  BM[1100,,00,11,,11,00,,000,,010,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,100,010,101]

   4425:  !(FREE)
   GOBUT763B:
        SETUP,  RETURN/TEST763C,                        !RETURN TO START OF NEXT SUBTEST
        NEXT,   GOTO-PAGE(7),                           !BUT TABLE
                J/BUTD-IS-ZERO                          !CHECK EQUALITY
   (4425)  DCS[0,00,0,0,0,0]  BM[0100,,00,11,,10,10,,001,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,100,001]
```



```
   ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

   !TEST 763C CHECKS THAT VECTOR-LOAD[1]H=-(UNIBUS-INTR-L) IS AGAIN LOW
   4721:
   TEST763C:
        P0,     LOAD-ENUA(ZTARGET401),                  !BIT<01> CLEAR
                LOAD-ERROR(TEST763C),                   !ERROR DIRECTORY KEY
                DCS-CTR(C3,),                           !COMPARE AT TARGET
                BUMP-VERIFY,                            !COUNT
        NEXT,   J/GOBUT763C                             !
   (4721)  DCS[1,00,1,0,0,1]  BM[1100,,00,11,,11,00,,000,,001,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,100,010,110]

   4426:  !(FREE)
   GOBUT763C:
        SETUP,  RETURN/TEST763D,                        !RETURN TO START OF NEXT SUBTEST
        NEXT,   GOTO-PAGE(7),                           !BUT TABLE
                J/BUTVECTLOAD                           !VECTOR-LOAD[1]H IN BIT<01>
   (4426)  DCS[0,00,0,0,0,0]  BM[0100,,00,11,,10,01,,000,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,100,110]




   ! - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

   !TEST 763D CHECKS THAT BG-SERVICE(0)H=BR>PS-L IS HIGH WHEN UNIBUS INTR REQUEST HAS BEEN SATISFIED
   4710:
   TEST763D:
        P0,     LOAD-ENUA(ZTARGET407),                  !BIT<02> SET  (ACTIVE LOW)
                LOAD-ERROR(TEST763D),                   !ERROR DIRECTORY KEY
                DCS-CTR(C3,),                           !COMPARE AT TARGET
        NEXT,   J/GOBUT763D                             !
   (4710)  DCS[1,00,1,0,0,0]  BM[1100,,00,11,,11,00,,000,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,100,010,111]

   4427:  !(FREE)
   GOBUT763D:
        SETUP,  RETURN/SCOPE763,                        !RETURN TO SCOPE LOOP TEST WORD
        NEXT,   GOTO-PAGE(7),                           !BUT TABLE
                J/BUTBGSERVL                            !BG-SERVICE[0]H IN BIT<02>
   (4427)  DCS[0,00,0,0,0,0]  BM[0100,,00,10,,00,11,,000,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,001,110]



   4430:  !(FREE)
   SCOPE763:

        P0,     BUSDIN_EMIT-[I],                        !RESET PROC UCON FOR EMIT
        P2,     PS_D-[I],                               !ZAP THE PS TO ZERO
        NEXT,   BUTO[SCOPE],                            !NO ERROR:  "KILL764A"  (+1, WORDS)
                J/KILL764A                              !   ERROR:  "EXPEC763A"  (-11, WORDS)
   (4430)  DCS[0,00,0,1,0,0]  BM[1000,,00,00,,00,01,,010,,010,,,0,0,0,,0,,0,,,1,1011,,,0,,0000,0,,,11,000,,,111,110,011]
```

```
	;- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -


	;*** ALL DONE WITH LINE CLOCK, DISABLE IT FROM FURTHER INTERRUPTS BEFORE LEAVING FOR EOP ***

	4763:
	KILL764A:
		P2-T,	D_ZERO, D(C)_ALU00,			;NOW MUST GO KILL THE LINE CLOCK
		P3,	DATO,					;BA=(177546), DL11-W CSR
			BUTA(CLR-FLAG-RES-UCON),		;ZAP DCON, RES-CONTROL FOR EOP ROUTINE (MUST BE DONE)
		NEXT,	J/EOP001				;
    (4763)  DCS[0,00,0,0,0,0]  BM[0011,,00,00,,00,00,,000,,010,,,0,1,0,,0,,0,,,1,0010,,,0,,0000,0,,,11,010,,,100,011,001]







	;.PAGE=========================================================================================================

	.TOC	*	END OF PASS CODE

	;*****************************************************************************************
	;
	;	FINAL ENTRY HERE TO TEST FOR MULTIPLE PASSES, VERIFY MODE, ET AL
	;
	;
	;	*** INDICATIONS ON CONSOLE DISPLAY ***
	;
	;		0,0,0,0,0,0,	DCS IS RUNNING; REQUIRES APPROX 8, SECONDS/64, PASSES
	;				"RUN"	"PROC"	"USER"	"CONSOLE"	"BATTERY"
	;				(ON)	(BLINK)	(BLINK)	(OFF)		(OFF)
	;
	;		0 0 0 0 0 0	ERROR DETECTED / PASS=1 / SCOPE LOOPING
	;	OR	2 1 2 1 2 1
	;
	;		0,0,0,0,0,0,	ERROR DETECTED / 1<PASS<64, / SCOPE LOOPING
	;	OR	2,1,2,1,2,1,
	;
	;		1,2,3,3,2,1,	SUCCESSFUL 64, PASSES COMPLETED
	;
	;
	;	AT END OF PASS, THE GENERAL REGISTERS (BASE MACHINE R0-R7)
	;	  CONTAIN THE FOLLOWING INFORMATION:
	;
	;		R0  =  (NU)
	;		R1  =  (NU)
	;		R2  =  (NU)
	;		R3  =  (NU)
	;		R4  =  (NU)
	;		R5  =  REVISION NUMBER OF DCS MICROCODE, WITH
```



```
	;			BIT15 SET TO INDICATE END OF PASS
	;		R6  =  (NU)
	;		R7  =  (123321), END OF PASS INDICATION CONSTANT
	;
	;*****************************************************************************************


	4431:	;(FREE)
	EOP001:
		P0,	DCS-CTR(C15,),				;HOLD UP ERROR COMPARE
		NEXT,	GOTO-PAGE(6),				;XFER
			J/EOP002				;
    (4431)  DCS[0,00,1,0,0,0]  BM[0000,,00,00,,00,00,,000,,110,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,100,101,010]


	6452:	;(FREE)
	EOP002:
		P3,	CSPD(01)_EMIT, EMIT/123321,		;SETUP SUCCESSFUL END OF PASS CONSTANT
		NEXT,	J/EOP003				;
    (6452)  DCS[0,00,0,0,0,0]  BM[1010,,10,01,,10,11,,010,,001,,,0,0,0,,0,,0,,,0,1110,,,1,,0000,0,,,11,000,,,101,001,000]


	6510:	;(FREE)
	EOP003:
		P2-T,	D_CSPD(D01), D(C)_0,			;GET EOP CONSTANT INTO
		P3,	PC_D,					;  GPR PC, ON BOTH SP SIDES
		NEXT,	J/EOP004				;  FOR DISPLAY FROM "CONSOLE HALT" ROUTINE
    (6510)  DCS[0,00,0,0,0,0]  BM[1010,,10,00,,10,01,,111,,000,,,0,1,0,,0,,0,,,0,1110,,,0,,0011,0,,,11,000,,,101,001,001]


	6511:	;(FREE)
	EOP004:
		SETUP,	RETURN/EOP005,				;GO TO DISPLAY-D-IN-LIGHTS ROUTINE
		NEXT,	CALL[DISPLAY]				;  WHICH SHOULD PUT (212121) INTO DISPLAY
    (6511)  DCS[0,00,0,0,0,0]  BM[0111,,00,01,,00,01,,001,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,010,001]
								;  FOR INTERMEDIATE EOP INDICATION

	7211:	;(FREE)
	EOP005:
		P0,	SIGNAL-EOP,				;TRY TO SIGNAL EOP
		P2-T,	SR_ALL-ONES,				;SET SR TO ALL ONES, FOR FULL BASE MACHINE INIT
		NEXT,	GOTO-PAGE(6),				;XFER TO 6
			BUTD[VERIFY-MODE],			;IF TRUE,  "VFY001"  (NEXT PAGE FOLLOWING)
			J/EOP006				;IF FALSE, "EOP006"  (NEXT MICROWORD)
    (7211)  DCS[0,00,0,0,1,0]  BM[1111,,00,00,,11,01,,101,,110,,,0,0,1,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,111,111,110]


	6776:
	EOP006:
		P0,	DCS-CTR(C11,),				;HOLD UP ERROR COMPARE
		P3-0,	SET-CONSOLE-DP-LEDS,			;UCON SET OF LEDS, INDICATING >=1 SUCCESSFUL PASSES
```

```
18268            NEXT,   BUTD[EOP=OVERFLOW],                     ;IF TRUE,  "EOP007" FOLLOWING, DONE 64. PASSES
18269                    J/EOP007                                ;IF FALSE, "TEST001", GO FOR NEXT PASS AT (4000)
    (6776)  DCS[0,00,1,0,0,0]  BM[0100,,00,00,,00,00,,110,,001...0,0,0,,0,,0,,1,1011,,,0,,0000,0,,,11,000,,,101,001,010]
18270                                                            ;BEGIN COUNTDOWN
18271                                                            ;  3 ... 2 ... 1 .
18272
18273
18274    6512:  ;(FREE)
18275    EOP007:
18276            SETUP,  RETURN/EOP0;0,                          ;GO TO $USR THAT PUTS REV-NUMBER, WITH
18277            NEXT,   GOTO-PAGE(7),                           ;  B<15>=(1), INTO B.M. GPR "R5"
18278                    J/INSERTEOPREVNO                        ;
    (6512)  DCS[0,00,0,0,0,0]  BM[0111,,00,01,,00,01,,010,,111...0,0,0,,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,010,001,100]
18279
18280
18281    7212:  ;(FREE)
18282    EOP010:
18283            SETUP,  RETURN/CON99;                           ;RETURN TO "FORCE CONSOLE-MODE HALT" ROUTINE IN BASE MACHINE
18284            NEXT,   GOTO-PAGE(3),                           ;GOTO "INIT---" ROUTINE FOR FULL BASE MACHINE
18285                    J/INIT01                                ;  MICROCODE INITIALIZATION, BUTA(RETURN) AT END TO "CON99"
    (7212)  DCS[0,00,0,0,0,0]  BM[0001,,00,00,,01,00,,000,,011...0,0,0,,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,100,001,010]
18286
18287
18288
18289
18290
18291    ;.PAGE=================================================================================================================
18292
18293
18294    .TOC    *   VERIFY MODE CODE
18295
18296    ;
18297    ;       VERIFY MODE ENTERS HERE:
18298    ;
18299
18300    6774:
18301    VFY001:
18302            PO,     DCS-CTR(C4,);                           ;LOAD COUNTER FOR COMPARE IN 4. MICROWORDS
18303            NEXT,   GOTO-PAGE(4),                           ;EFER
18304                    J/VFY002                                ;
    (6774)  DCS[0,00,1,0,0,0]  BM[1011,,00,00,,00,00,,000,,100...0,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,100,001,010]
18305
18306
18307    4412:  ;(FREE)
18308    VFY002:
18309            P3,     CLEAR-CONSOLE-LED,                      ;MAKE "CONSOLE" LED BLINK, JUST FOR FUN
18310            NEXT,   J/VFY003                                ;
    (4412)  DCS[0,00,0,0,0,0]  BM[0100,,00,00,,00,00,,010,,001...0,0,0,0,,0,,0,,,1,1011,,,0,,0000,0,,,11,000,,,010,101,101]
18311
18312    4255:
```



```
18313    VFY003:
18314            P0,     LOAD-ENUA(ZTARGET523),                  ;ERROR CODE = (4255), ENUA = (7523)
18315                    LOAD-ERROR(VFY003),                     ;ERROR DIRECTORY KEY
18316                    BUFP-VERIFY,                            ;GIVE A VERIFY PULSE
18317            NEXT,   J/VFY004                                ;
    (4255)  DCS[1,00,0,0,0,1]  BM[0000,,00,11,,11,01,,010,,011...0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,100,011,011]
18318
18319
18320    4433:  ;(FREE)
18321    VFY004:
18322            SETUP,  RETURN/VFY005,                          ;RETURN TO INLINE
18323            NEXT,   GOTO-PAGE(7),                           ;BUF'S ARE ON PAGE 7
18324                    J/ZTARGET522                            ;TNUA = (7522), NOT EQUAL TO ENUA
    (4433)  DCS[0,00,0,0,0,0]  BM[0111,,00,01,,11,11,,101,,111,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,101,010,010]
18325
18326
18327    ;       NEXT MICROWORD COMES FROM ZTARGET522, AT WHICH THE ENABLED COMPARE
18328    ;       TAKEN PLACE.  ENUA WAS SETUP NOT EQUAL TO TNUA, SO ERROR SHOULD BE SIGNALED
18329
18330
18331
18332    7375:
18333    VFY005:
18334            P3,     SET-CONSOLE-LED,                        ;THE OTHER HALF OF MAKING IT BLINK
18335            NEXT,   BUTD[SCOPE],                            ;NO ERROR:  "VFY005"  (SELF LOOP, SHOULDN'T HAPPEN)
18336                    J/VFY005                                ;   ERROR:  "VFY006"  (SHOULD HAPPEN)
    (7375)  DCS[0,00,0,1,0,0]  BM[0100,,00,00,,00,00,,100,,001...0,0,0,,,0,,0,,,1,1011,,,0,,0000,0,,,11,000,,,011,111,101]
18337
18338
18339    7374:
18340    VFY006:
18341            P0,     SIGNAL-EOP,                             ;GIVE AN EOP PULSE, AFTER ERROR SIGNALED
18342            NEXT,   GOTO-PAGE(6),                           ;LOOP BACK
18343                    J/VFY001                                ;  ON CONTINUOUS VERIFY
    (7374)  DCS[0,00,0,0,1,0]  BM[0000,,00,00,,00,00,,000,,110...0,0,0,,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,111,111,100]
18344
18345
18346
18347
18348
18349    ;.PAGE=================================================================================================================
18350
18351
18352    .TOC    *   DCS MICROCODE REVISION NUMBER
18353
18354    ;       THE FOLLOWING ROUTINE WILL PUT THE CURRENT DCS MICROCODE REVISION NUMBER
18355    ;       INTO B.M. GPR "R5". FROM THE EMIT FIELD OF THE MICROWORD.
18356    ;
18357    ;       THE ENTRY "INSERTREVNO" IS USED, EXCEPT AT END OF PASS; IE B<15>=(0)
18358    ;       THE ENTRY "INSERTEOPREVNO" IS USED ONLY AT END OF PASS; IE, B<15>=(1)
18359
18360
```

```
18361    7214:  ;(FREE)
18362    INSERTCOPREVNO:
18363            P0,     BUSDIN_EMIT-[1],                ;SELECT EMIT
18364            NEXT,   J/INSERT02                      ;
      (7214)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,01,,000,,000,,,0,0,0,,0,,0,,,1,1001,,,0,,0000,0,,,11,000,,,010,001,101]
18365
18366    7215:  ;(FREE)
18367    INSERT02:
18368            P3,     CSPD[17]_EMIT, EMIT/REV-NUMBER-AND-BIT, ;DCS REVISION NUMBER, BIT SET
18369            NEXT,   J/INSERT04                      ;
      (7215)  DCS[0,00,0,0,0,0]  BM[1000,,10,00,,00,01,,000,,001,,,0,0,0,,0,,0,,,0,0000,,,1,,0000,0,,,11,000,,,010,010,000]
18370
18371
18372    7216:  ;(FREE)
18373    INSERTREVNO:
18374            P0,     BUSDIN_EMIT-[1],                ;SELECT EMIT
18375            NEXT,   J/INSERT03                      ;
      (7216)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,01,,000,,000,,,0,0,0,,0,,0,,,1,1001,,,0,,0000,0,,,11,000,,,010,001,111]
18376
18377    7217:  ;(FREE)
18378    INSERT03:
18379            P3,     CSPD[17]_EMIT, EMIT/REV-NUMBER,         ;DCS REVISION NUMBER, BIT CLEAR
18380            NEXT,   J/INSERT04                      ;
      (7217)  DCS[0,00,0,0,0,0]  BM[0000,,10,00,,00,01,,000,,001,,,0,0,0,,0,,0,,,0,0000,,,1,,0000,0,,,11,000,,,010,010,000]
18381
18382
18383    7220:  ;(FREE)
18384    INSERT04:
18385            P2=T,   D_CSPD(D17), DIC]_0,            ;GET IT
18386            P3,     A5_D,                           ;  AND STUFF IT
18387            NEXT,   J/RESETUCOMP                    ;    AND RETURN
      (7220)  DCS[0,00,0,0,0,0]  BM[1010,,10,00,,10,01,,110,,000,,,0,1,0,,0,,0,,,0,0000,,,0,,0011,0,,,11,000,,,010,111,001]
18388
18389
18390
18391    1,PAGE;*********************************************************************************************************************
18392
18393
18394    .TOC    *   COMMON SUBROUTINES
18395
18396    .TOC    *      CONSOLE DISPLAY SUBROUTINE
18397
18398    ;
18399    ;       DISPLAYS NUMBER REPRESENTED BY D<05:00>*D<05:00>*D<05:00>*D<05:00>
18400    ;       AS SIX OCTAL DIGITS IN CONSOLE 7 SEGMENT DISPLAY
18401    ;
18402
18403    7221:  ;(FREE)
18404    DISPLAY:
18405            P3=U,   CLEAR-CONSOLE-COUNTER,          ;POINT TO DIGITS ....XX
18406            NEXT,   J/DISP002                       ;
      (7221)  DCS[0,00,0,0,0,0]  BM[0100,,00,00,,00,00,,010,,000,,,0,0,0,,0,,0,,,1,1011,,,0,,0000,0,,,11,000,,,010,010,010]
18407
18408
```



```
18409    7222:  ;(FREE)
18410    DISP002:
18411            P3=U,   STROBE-CONSOLE-DISPLAY,         ;WRITE OUT DIGITS ....XX
18412            NEXT,   J/DISP003                       ;
      (7222)  DCS[0,00,0,0,0,0]  BM[0100,,00,00,,00,00,,000,,001,,,0,0,0,,0,,0,,,1,1011,,,0,,0000,0,,,11,000,,,010,010,011]
18413
18414
18415    7223:  ;(FREE)
18416    DISP003:
18417            P3=U,   INCREMENT-CONSOLE-COUNTER,      ;POINT TO DIGITS ..XX..
18418            NEXT,   J/DISP004                       ;
      (7223)  DCS[0,00,0,0,0,0]  BM[0100,,00,00,,00,00,,100,,000,,,0,0,0,,0,,0,,,1,1011,,,0,,0000,0,,,11,000,,,010,010,100]
18419
18420
18421    7224:  ;(FREE)
18422    DISP004:
18423            P3=U,   STROBE-CONSOLE-DISPLAY,         ;WRITE OUT DIGITS ..XX..
18424            NEXT,   J/DISP005                       ;
      (7224)  DCS[0,00,0,0,0,0]  BM[0100,,00,00,,00,00,,000,,001,,,0,0,0,,0,,0,,,1,1011,,,0,,0000,0,,,11,000,,,010,010,101]
18425
18426
18427    7225:  ;(FREE)
18428    DISP005:
18429            P3=U,   INCREMENT-CONSOLE-COUNTER,      ;POINT TO DIGITS XX....
18430            NEXT,   J/DISP006                       ;
      (7225)  DCS[0,00,0,0,0,0]  BM[0100,,00,00,,00,00,,100,,000,,,0,0,0,,0,,0,,,1,1011,,,0,,0000,0,,,11,000,,,010,010,110]
18431
18432
18433    7226:  ;(FREE)
18434    DISP006:
18435            P3=U,   STROBE-CONSOLE-DISPLAY,         ;WRITE OUT DIGITS XX....
18436            NEXT,   J/RESETUCOMP                    ;GO RESET PROC UCON/EMIT, DO A (RETURN)
      (7226)  DCS[0,00,0,0,0,0]  BM[0100,,00,00,,00,00,,000,,001,,,0,0,0,,0,,0,,,1,1011,,,0,,0000,0,,,11,000,,,010,111,001]
18437
18438
18439
18440
18441
18442    1,PAGE;*********************************************************************************************************************
18443
18444
18445    .TOC    *      CLEAR I-O / BUS CONTROL / SERVICE AREA STATUS LATCHES SUBR
18446
18447
18448    ;
18449    ;       THIS SUBR CLEARS OUT, VIA I-O UCON COMMANDS, THOSE STATUS LATCHES
18450    ;       CONCERNED WITH SERVICE CONDITIONS, UNIBUS ERROR CONDITIONS, ETC.
18451    ;
18452
18453    7227:  ;(FREE)
18454    CLEAR-I-O-A:
18455            P3,     BUS-INIT-UCON-[1],              ;DO A 10 MILLISEC UNIBUS INIT
18456            NEXT,   J/CLEAR-I-O-B                   ;
      (7227)  DCS[0,00,0,0,0,0]  BM[0100,,00,01,,11,00,,000,,000,,,0,0,0,,0,,0,,,1,1011,,,0,,0000,0,,,11,000,,,010,011,000]
```

```
18457
18458    7230:  ;(FREE)
18459    CLEAR-I-O-R:
18460            P3,     CLR-JAM-ERRORS-(I),                     ;RESET CACHE ERROR STATUS
18461            NEXT,   J/CLEARIO02                             ;
     (7230)  DCS[0,00,0,0,0,0]  BM[0100,,00,00,,10,00,,000,,000,,,0,0,0,,0,,0,,,1,1011,,,0,,0000,0,,,11,000,,,010,011,001]
18462
18463    7231:  ;(FREE)
18464    CLEARIO02:
18465            P3,     CLEAR-CONSOLE-SERVICE,                  ;CLEAR OUT CONSOLE SRVC(1)H
18466            P3,     CLR-NPR-TIMEOUT-(I),                    ;RESET NPR/SACK TIME OUT STATUS
18467            NEXT,   J/CLEARIO04                             ;
     (7231)  DCS[0,00,0,0,0,0]  BM[0100,,00,00,,11,00,,110,,000,,,0,0,0,,0,,0,,,1,1011,,,0,,0000,0,,,11,000,,,010,011,010]
18468
18469    7232:  ;(FREE)
18470    CLEARIO04:
18471            P3,     CLEAR-CONSOLE-LED,                      ;SETUP FOR 16. BIT ADDRESS MODE ON UNIBUS
18472            P3,     CLR-PWR-FAIL-(I),                       ;RESET POWER FAIL STATUS
18473            NEXT,   J/CLEARIO05                             ;
     (7232)  DCS[0,00,0,0,0,0]  BM[0100,,00,01,,00,00,,010,,001,,,0,0,0,,0,,0,,,1,1011,,,0,,0000,0,,,11,000,,,010,011,011]
18474
18475    7233:  ;(FREE)
18476    CLEARIO05:
18477            P3,     CLR-YELLOW-ZONE-(I),                    ;RESET YELLOW ZONE STATUS
18478            NEXT,   J/RESETUCONP                            ;
     (7233)  DCS[0,00,0,0,0,0]  BM[0100,,00,01,,01,00,,000,,000,,,0,0,0,,0,,0,,,1,1011,,,0,,0000,0,,,11,000,,,010,111,001]
18479
18480    ;"RESETUCONP" IS AT END OF "DINTOIR" SUBROUTINE
18481    ;THIS WORD  (1) ENABLES BUSDIN_EMIT,  (2) ENABLES PROC EN-CLK-IR,
18482    ;AND EXITS WITH A RUTA(RETURN)
18483
18484
18485
18486
18487
18488    ;.PAGE********************************************************************************************************************
18489
18490
18491    .TOC   *      SUBR FOR PUTTING SELECTED PORTIONS OF D[15-00] INTO IR
18492
18493    ;
18494    ;       THESE SUBR(S) ARE USED IN TESTING THE ALU&LOGIC FUNCTIONS.
18495    ;
18496    ;       NOTE:  ENTRY POINTS "D[15-12]", "D[11-06]", "D[05-00]" EXPECT THAT:
18497    ;                (1) "IR_DBUF", "DBUF_D" FUNCTIONS ARE ALREADY ENABLED
18498    ;                (2) THE FOLLOWING CONSTANTS ARE IN CSP[17:14]:
18499    ;                        CSP(17) = (007700);     CSP(16) = (170000);
18500    ;                        CSP(15) = (000077);     CSP(14) = (000100);
18501    ;
18502    ;       FURTHERMORE:
18503    ;                (1) ENTRY POINT "DZERO" SETS UP THE UCONS:
18504    ;                        "IR_DBUF", "DBUF_0"
18505    ;                (2) ENTRY POINT "D[15-12]" COPIES THE ORIGINAL PATTERN, LEFT IN
18506    ;                        D, INTO ASPHI[17] FOR SAFEKEEPING, AND REUSE LATER
```



```
18507    ;
18508
18509
18510    7234:  ;(FREE)
18511    D[15-12]:
18512            P2-U,   IR_DBUF,                                ;JUST HAPPENS, DONT CARE NOW
18513            P3,     DBUF_D,                                 ;PATTERN IN D -> DBUF
18514                    ASBSPHI[17]_D,                          ;SAVE IN ASPHI, TOO
18515            NEXT,   J/D1512A                                ;
     (7234)  DCS[0,00,0,0,0,0]  BM[0000,,00,11,,00,01,,011,,000,,,0,0,0,,0,,0,,,1,1010,,,0,,1011,0,,,11,000,,,010,011,101]
18516
18517    7235:  ;(FREE)
18518    D1512A:
18519            P2-U,   IR_DBUF,                                ;SEND DBUF -> IR
18520            P3,     DBUF_D,                                 ;JUST HAPPENS, DONT CARE NOW
18521            NEXT,   J/BUTIR15-12                            ;IR<15:12>"M IN BIT<03:00> "BUT"
     (7235)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,1,1010,,,0,,0000,0,,,11,000,,,011,000,000]
18522
18523    7236:  ;(FREE)
18524    D[11-06]:
18525            P2-T,   SR_ASPHI[17]-AND-007700,                ;MASK ALL BITS EXCEPT D<11:06> TO ZEROES
18526            NEXT,   J/D1106A                                ;
     (7236)  DCS[0,00,0,0,0,0]  BM[1011,,11,00,,11,01,,011,,000,,,0,0,1,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,010,011,111]
18527
18528    7237:  ;(FREE)
18529    D1106A:
18530            P2-T,   D_SP-IOR-170000, SAVE-D(C),             ;FORCE BITS D<15:12> TO ONES
18531                                                            ;THIS MANUVER SHOULD FORCE INSTRS DIAGNOSTIC (E78)
18532                                                            ;ROM ADDRESSES (752), (725) WHEN PATTERN GOES INTO IR
18533            P2-U,   IR_DBUF,                                ;JUST HAPPENS, DONT CARE NOW
18534            P3,     DBUF_D,                                 ;PUT PATTERN JUST SET IN D INTO DBUF
18535            NEXT,   J/DTOIR8                                ;
     (7237)  DCS[0,00,0,0,0,0]  BM[1110,,11,01,,00,00,,000,,111,,,0,1,0,,0,,0,,,1,1010,,,0,,0000,0,,,11,000,,,010,100,011]
18536
18537    7240:  ;(FREE)
18538    D[05-00]:
18539            P2-T,   SR_ASPHI[17]-AND-000077,                ;MASK ALL BITS EXCEPT D<05:00> TO ZEROES
18540            NEXT,   J/D0500A                                ;
     (7240)  DCS[0,00,0,0,0,0]  BM[1011,,11,10,,11,01,,011,,000,,,0,0,1,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,010,100,001]
18541
18542    7241:  ;(FREE)
18543    D0500A:
18544            P2-T,   D_SR-IOR-000100, SAVE-D(C),             ;FORCE BIT D<06> TO A ONE
18545                                                            ;THIS MANUVER SHOULD FORCE INSTRS DIAGNOSTIC (E88)
18546                                                            ;ROM ADDRESSES (152), (125) WHEN PATTERN GOES INTO IR
18547            P2-U,   IR_DBUF,                                ;JUST HAPPENS, DONT CARE NOW

18548            P3,     DBUF_D,                                 ;PUT PATTERN JUST SET IN D INTO DBUF
18549            NEXT,   J/DTOIR8                                ;
     (7241)  DCS[0,00,0,0,0,0]  BM[1110,,11,11,,00,00,,000,,111,,,0,1,0,,0,,0,,,1,1010,,,0,,0000,0,,,11,000,,,010,100,011]
18550
18551    7242:  ;(FREE)
18552    DZERO:
18553            P2-U,   IR_DBUF-(I),                            ;JUST HAPPENS, DONT CARE
18554            P3,     DBUF_D-(I),                             ;COPY PATTERN IN 0 TO DBUF
```

```
                NEXT,   J/DTOIRB                                        ;
        (7242)  DCS[0,00,0,0,0,0]  BM[0100,,00,00,,00,01,,000,,100,,,0,0,0,,0,,0,,,1,1011,,,0,,0000,0,,,11,000,,,010,100,011]

        7243:   ;(FREE)
        DTOIRB:
                P2=U,   IR_DBUF,                                        ;NOW COPY PATTERN IN DBUF TO IR
                P3,     DBUF_D,                                         ;JUST HAPPENS, DONT CARE
                NEXT,   J/BUTINSTRS                                     ;GO DO INSTRS "BUT" ON PATTERN IN IR
        (7243)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,1,1010,,,0,,0000,0,,,11,000,,,011,000,001]





        ;.PAGE====================================================================================================

        .TOC    *       UCON SUBROUTINES (FLAGS, PS, FPS, CUA, SERVICE, JAM, PBA)

        ;
        ;       THESE SUBROUTINES MANIPULATE THE PROCESSOR UCONS, AND VARIOUS OTHER BUSDIN DRIVERS
        ;
        ;               ASPLO(17) = A TEMPORARY LOCATION, DESTROYED
        ;
        ;               CSP(02) = VALUE EXPECTED TO BE READ
        ;               CSP(03) = ACTUAL VALUE READ
        ;               CSP(04) = MASK VALUE
        ;
        ;               AT RETURN, D = (BUSDINSELECT.AND.MASKVALUE).XOR.EXPECTEDVALUE
        ;

        7244:   ;(FREE)
        CINSERVICETOD:
                P3,     CSPD[02]_EMIT, EMIT/000340,                     ;SERVICE PORT OF STATUS:  ALL BITS RESET
                NEXT,   J/CJESERVICETOD                                 ;
        (7244)  DCS[0,00,0,0,0,0]  BM[0000,,10,00,,00,11,,100,,000,,,0,0,0,,0,,0,,,0,1101,,,1,,0000,0,,,11,000,,,010,100,111]

        7245:   ;(FREE)
        DATISERVICETOD:
                P3,     CSPD[02]_EMIT, EMIT/100340,                     ;SERVICE PORT OF STATUS:  ONLY DATI[1]H SET
                NEXT,   J/CJESERVICETOD                                 ;
        (7245)  DCS[0,00,0,0,0,0]  BM[1000,,10,00,,00,11,,100,,000,,,0,0,0,,0,,0,,,0,1101,,,1,,0000,0,,,11,000,,,010,100,111]

        7246:   ;(FREE)
        DATOSERVICETOD:
                P3,     CSPD[02]_EMIT, EMIT/002340,                     ;SERVICE PORT OF STATUS:  ONLY DATO[1]H SET
                NEXT,   J/CJESERVICETOD                                 ;
        (7246)  DCS[0,00,0,0,0,0]  BM[0000,,10,01,,00,11,,100,,000,,,0,0,0,,0,,0,,,0,1101,,,1,,0000,0,,,11,000,,,010,100,111]

        7247:   ;(FREE)
        CJESERVICETOD:
                P3,     CLR-JAM-ERRORS-[I],                             ;SPECIAL ENTRY POINT TO CLEAR OUT JAM
                NEXT,   J/SERVICETOD                                    ;  PORT OF STATUS MUX, BEFORE READ SERVICE
        (7247)  DCS[0,00,0,0,0,0]  BM[0100,,00,00,,10,00,,000,,000,,,0,0,0,,0,,0,,,1,1011,,,0,,0000,0,,,11,000,,,010,101,000]
```



```
        7250:   ;(FREE)
        SERVICETOD:
                P0,     BUSDIN_SERVICE-[I],                             ;SERVICE (PORT 1) OF STATUS MUX
                NEXT,   J/GETPROCDAT                                    ;
        (7250)  DCS[0,00,0,0,0,0]  BM[0100,,01,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,1,1001,,,0,,0000,0,,,11,000,,,011,111,011]

        ;       -       -       -       -       -       -       -       -       -       -       -       -

        7251:   ;(FREE)
        PBATOD:
                P0,     BUSDIN_PBA-[I],                                 ;PBA (PORT 3) OF STATUS MUX
                NEXT,   J/GETPROCDAT                                    ;
        (7251)  DCS[0,00,0,0,0,0]  BM[1100,,01,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,1,1001,,,0,,0000,0,,,11,000,,,011,111,011]

        ;       -       -       -       -       -       -       -       -       -       -       -       -

        7252:   ;(FREE)
        FLAGFPSTOD:
                P0,     BUSDIN_FLAGS&FPS-[I],                           ;PUT FLAGS, FPS ON BUSDIN
                NEXT,   J/GETPROCDAT                                    ;
        (7252)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,11,01,,000,,000,,,0,0,0,,0,,0,,,1,1001,,,0,,0000,0,,,11,000,,,011,111,011]

        ;       -       -       -       -       -       -       -       -       -       -       -       -

        7253:   ;(FREE)
        PSTOD:
                P0,     BUSDIN_PS-[I],                                  ;PUT PS ON BUSDIN
                NEXT,   J/GETPROCDAT                                    ;
        (7253)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,10,01,,000,,000,,,0,0,0,,0,,0,,,1,1001,,,0,,0000,0,,,11,000,,,011,111,011]

        ;       -       -       -       -       -       -       -       -       -       -       -       -

        7254:   ;(FREE)
        ODDJAMTOD:
                P3,     CSPD[02]_EMIT, EMIT/101004,                     ;JAM PORT OF STATUS:  ONLY ODD-ADDR[1]H SET
                NEXT,   J/JAMTOD                                        ;
        (7254)  DCS[0,00,0,0,0,0]  BM[1000,,10,00,,10,00,,000,,100,,,0,0,0,,0,,0,,,0,1101,,,1,,0000,0,,,11,000,,,010,101,110]

        7255:   ;(FREE)
        CLRJAMTOD:
                P3,     CSPD[02]_EMIT, EMIT/001000,                     ;JAM PORT OF STATUS:  ALL BITS RESET
                NEXT,   J/JAMTOD                                        ;
        (7255)  DCS[0,00,0,0,0,0]  BM[0000,,10,00,,10,00,,000,,000,,,0,0,0,,0,,0,,,0,1101,,,1,,0000,0,,,11,000,,,010,101,110]

        7256:   ;(FREE)

        JAMTOD:
                P0,     BUSDIN_JAM-[I],                                 ;PUT JAM REG (STATUS MUX PORT 2)
                NEXT,   J/GETPROCDAT                                    ;ONTO BUSDIN
        (7256)  DCS[0,00,0,0,0,0]  BM[1100,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,1,1001,,,0,,0000,0,,,11,000,,,011,111,011]

        ;       -       -       -       -       -       -       -       -       -       -       -       -

        7257:   ;(FREE)
```

```
18650    CUATOD:
18651            P0,     BUSDIN_CUA=[I],                 ;PUT CUA REG (HBMUX PORT 2)
18652            NEXT,   J/GETPROCDAT                    ;ONTO BUSDIN
     (7257)  DCS[0,00,0,0,0,0,0]  BM[0000,,00,00,,01,01,,000,,000,,,0,0,0,,0,,0,,1,1001,,,0,,0000,0,,,11,000,,,011,111,011]
18653
18654    |       -       -       -       -       -       -       -       -       -       -       -       -
18655
18656    7373:
18657    GETPROCDAT:
18658            P3,     CSPD[03]_BUSDIN,                ;GET PREVIOUSLY ENABLED PROC DATA
18659            NEXT,   J/GETMSKPROCDAT                 ;
     (7373)  DCS[0,00,0,0,0,0,0]  BM[0000,,10,00,,00,00,,000,,000,,,0,0,0,,0,,0,,0,1100,,,1,,0000,0,,,11,000,,,010,110,000]
18660
18661    7260:  ;(FREE)
18662    GETMSKPROCDAT:
18663            P2=T,   D_CSPD(D04), D[C]_D,            ;GET MASK VALUE
18664            P3,     ASPLO[17]_D,                    ;INTO ALU SIDE
18665            NEXT,   J/MSKPROCDAT                    ;
     (7260)  DCS[0,00,0,0,0,0,0]  BM[1010,,10,00,,00,01,,011,,000,,,0,1,0,,0,,0,,0,1011,,,0,,0001,0,,,11,000,,,010,110,001]
18666
18667    7261:  ;(FREE)
18668    MSKPROCDAT:
18669            P2=T,   D_ASPLO[17]=AND=CSPD(D03), SAVE=D[C],   ;MASK OUT UNWANTED BITS
18670            P3,     ASPLO[17]_D,                    ;AND WRITE BACK
18671            NEXT,   J/CMPPROCDAT                    ;
     (7261)  DCS[0,00,0,0,0,0,0]  BM[1011,,10,00,,10,01,,011,,111,,,0,1,0,,0,,0,,0,1100,,,0,,0001,0,,,11,000,,,010,110,010]
18672
18673    7262:  ;(FREE)
18674    CMPPROCDAT:
18675            P2=T,   D_ASPLO[17]=XOR=CSPD(D02), SAVE=D[C],   ;COMPARE OBTAINED, EXPECTED BITWISE
18676            NEXT,   J/RESETPROCDAT                  ;
     (7262)  DCS[0,00,0,0,0,0,0]  BM[0110,,10,00,,10,01,,011,,111,,,0,1,0,,0,,0,,0,1101,,,0,,0000,0,,,11,000,,,010,110,011]
18677
18678    7263:  ;(FREE)
18679    RESETPROCDAT:
18680            P0,     BUSDIN_EMIT=[I],                ;RESET PROC UCON
18681                    EM=CLK=IR[15=00],               ;
18682            NEXT,   J/BUTD=IS=ZERO                  ;AND GO TEST D<15:00>
     (7263)  DCS[0,00,0,0,0,0,0]  BM[0000,,00,00,,00,01,,000,,100,,,0,0,0,,0,,0,,1,1001,,,0,,0000,0,,,11,000,,,011,100,001]
18683
18684
18685
18686
18687
18688    ;.PAGE**********************************************************************************************************
18689
18690
18691    .TOC    *       SUBR FOR LOADING FPS<3:0> (VIA BUTA(DIAGNOSE))
18692
18693    |
18694    |       THIS SUBROUTINE LOADS CSP(16)<3:0> => FPS<3:0> VIA THE BUTA(DIAGNOSE) FUNCTION.
18695    |
18696    |       REQUIREMENTS FOR ENTRY:
18697    |
```



```
18698    |               (1)  LOADING SR MUST BE SETUP
18699    |               (2)  CSP(16) SETUP WITH BITS TO LOAD
18700    |               (3)  CSP(00) CONTAINS RETURN MICROADDRESS IN BITS<14:03>
18701    |
18702
18703    7264:  ;(FREE)
18704    LOADFPSCC:
18705            P2=T,   SR_ALL=ONES,                    ;SET BIT<00> FOR JAMUPP EXPECTED
18706            P3,     BUTA(DIAGNOSE),                 ;START THE XFR TO BM SEQUENCE
18707            NEXT,   J/LOADFPSCC02                   ;
     (7264)  DCS[0,00,0,0,0,0,0]  BM[1111,,00,00,,11,01,,101,,000,,,0,0,1,,0,,0,,,0,0000,,,0,,0000,0,,,11,011],,,010,110,101]
18708
18709    7265:  ;(FREE)
18710    LOADFPSCC02:
18711            NEXT,   XFR=TO=BM(LOADMZW4)             ;LOAD PAGE, POINT UPF AT BM CODE
     (7265)  DCS[0,00,0,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,100,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,100,,,011,01[,000]
18712
18713    ;LOADMZW4:                                      ;(THIS WORD ACTUALLY COMES OUT OF BM ROMS)
18714    |       P2=T,   D_CSPB(16),                      ;PUT "MD" = CSP(16) INTO D
18715    |       NEXT,   J/LOADMZW5                       ;
18716    |
18717    ;LOADMZW5:
18718    |       P2,     FPS(CC)_D[3=0],                 ;LOAD FPS<3:0>
18719    |       P3,     BUTA(DIAGNOSE),                 ;BEGIN XFR SEQUENCE BACK TO DCS ROMS
18720    |       NEXT,   J/MZERO2                        ;
18721    |
18722    ;MZERO2:
18723    |       NEXT,   GOTO=PAGE(4),                   ;
18724    |               J/XXXXXX                        ;
18725    |
18726    ;(CONTROL NOW RETURNS TO DCS AT "JAMUPP001" WORD)
18727    ;4777:
18728    ;JAMUPP00:
18729    |       *** SEE FLOWS ON SUBSEQUENT PAGE ***
18730
18731
18732
18733
18734
18735    ;.PAGE**********************************************************************************************************
18736
18737
18738    .TOC    *       SUBP TO COPY D=REGISTER TO DBUF TO IR
18739
18740    |
18741    |       SUBROUTINE TO COPY D=REGISTER TO DBUF TO IR
18742    |               AND LEAVE PROCESSOR UCON "CLK IR" AND "BUSDIN_EMIT" ACTIVE
18743    |
18744
18745
18746    7266:  ;(FREE)
18747    SRINTOIR:
18748            P2=T,   D_SR, SAVE=D[C],                ;COPY SR TO D
```

```
		NEXT,	J/DINTOIR				;
	(7266)	DCS[0,00,0,0,0,0]  BM[1111..00,00..00,00..000..111...0,1,0..0..0...0,0000...0..0000,0...11,000...010,110,111]

	7267:	;(FREE)
	DINTOIR:
		P2-U,	IR_DBUF-[I],				;JUST HAPPENS - DON'T CARE
		P3-U,	DBUF_D-[I],				;TRANSFER D => DBUF
		NEXT,	J/DBUFINTOIR				;
	(7267)	DCS[0,00,0,0,0,0]  BM[0100..00,00..00,01..000..100...0,0,0..0..0...1,1011...0..0000,0...11,000...010,111,000]

	7270:	;(FREE)
	DBUFINTOIR:
		EMITC,	EMIT/125200,				;JUNK ON EMIT: INSTRS/E88/405 PATTERN
		P2-U,	IR_DBUF,				;TRANSFER DBUF => BUSDIN => IR
		P3-U,	DBUF_D,					;JUST HAPPENS - DON'T CARE
		NEXT,	J/RESETUCONP				;
	(7270)	DCS[0,00,0,0,0,0]  BM[1010..00,10..10,10..000..000...0,0,0..0..0...1,1010...0..0000,0...11,000...010,111,001]

	7271:	;(FREE)
	RESETUCONP:
		SELECT,	UCON-PROC,				;RESET TO PROCESSOR UCON;
		ENABLE,	EN-CLK-IR[15-00],			;  ENABLE CLOCKING IR
			BUSDIN_EMIT[15-00],			;  PUT EMIT ONTO BUSDIN
		P0,	SET-UCON-CONTROL,			;WRITE CONTROLS
		NEXT,	BUTA(RETURN),				;RETURN TO CALLER
			J/BUTERROR7				;FATAL ERROR IF GO HERE
	(7271)	DCS[0,00,0,0,0,0]  BM[0000..00,00..00,01..000..100...0,0,0..0..0...1,1001...0..0000,0...11,111...011,111,110]


	; - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

	7272:	;(FREE)
	SRINTOIRS:
		P2-T,	D_SR, SAVE-D[C],			;COPY SR TO D
		NEXT,	J/DINTOIRS				;
	(7272)	DCS[0,00,0,0,0,0]  BM[1111..00,00..00,00..000..111...0,1,0..0..0...0,0000...0..0000,0...11,000...010,111,011]

	7273:	;(FREE)
	DINTOIRS:
		P2-U,	IR_DBUF-[I],				;JUST HAPPENS - DON'T CARE
		P3-U,	DBUF_D-[I],				;TRANSFER D => DBUF
		NEXT,	J/DBUFINTOIRS				;
	(7273)	DCS[0,00,0,0,0,0]  BM[0100..00,00..00,01..000..100...0,0,0..0..0...1,1011...0..0000,0...11,000...010,111,100]

	7274:	;(FREE)
	DBUFINTOIRS:
		EMITC,	EMIT/125200,				;JUNK ON EMIT: INSTRS/E88/405 PATTERN
		P2-U,	IR_DBUF,				;TRANSFER DBUF => BUSDIN => IR
		P3-U,	DBUF_D,					;JUST HAPPENS - DON'T CARE
		NEXT,	J/RESETUCONPS				;
	(7274)	DCS[0,00,0,0,0,0]  BM[1010..00,10..10,10..000..000...0,0,0..0..0...1,1010...0..0000,0...11,000...010,111,101]

	7275:	;(FREE)
	RESETUCONPS:
```



```
		SELECT,	UCON-PROC,				;RESET TO PROCESSOR UCON;
		ENABLE,	EN-CLK-IR[15-00],			;  ENABLE CLOCKING IR
			BUSDIN_EMIT[15-00],			;  PUT EMIT ONTO BUSDIN
		P0,	SET-UCON-CONTROL,			;WRITE CONTROLS
		NEXT,	J/BUTINSTRS				;AND EXIT TO DO INSTR-S BUT INTO TARGET AREA
	(7275)	DCS[0,00,0,0,0,0]  BM[0000..00,00..00,01..000..100...0,0,0..0..0...1,1001...0..0000,0...11,000...011,000,001]





	;.PAGE=========================================================================================

	.TOC	*	JAM UPP SERVICE SUBROUTINE

	;
	;	IF A JAMUPP TO 0777/4777 OCCURS, ACTION DEPENDS UPON SR<00> CONTENTS:
	;
	;		SR<00>=1 => RESET SR TO (000000), DO A BUTA[RETURN] TO ADDRESS IN CSP(00)
	;
	;		SR<00>=0 => SIGNAL ERROR, LOG STATUS, RETURN VIA "CUA" CONTENTS
	;			ERROR-CODE := INDICATES LAST TEST EXECUTED
	;

	4777:
	JAMUPP001:
		P1-T,	SR_D,					;SAVE CURRENT D CONTENTS IN SR
		NEXT,	BUTR(SR00),				;TEST SR<00> FLAG  (PREV SR00 CONTENTS)
			;IF SR<00>=1, EXPECTED JAM, GOTO "JANUPP002B"
			;IF SR<00>=0, NO JAM WAS EXPECTED, FORCE ERROR AT "JANUPP003"
			J/JAMUPP003				;
	(4777)	DCS[0,00,0,0,0,0]  BM[1111..00,00..01,01..000..000...1,0,1..0..0...0,0000...0..0000,0...00,000...111,101,110]

	4757:
	JAMUPP002B:
		P2-T,	D_CSPD(D00), SAVE-D[C],			;GET RETURN ADDRESS
		NEXT,	J/JAMUPP002C				;
	(4757)	DCS[0,00,0,0,0,0]  BM[1010..10,00..00,00..000..111...0,1,0..0..0...0,1111...0..0000,0...11,000...100,011,100]

	4434:	;(FREE)
	JAMUPP002C:
		P0,	RETURN_D[14-03], PAGE(7),		;PUT RETURN ADDRESS INTO REG
		P2-T,	D_SR, SAVE-D[C],			;RESTORE OLD D
		NEXT,	J/JAMUPP002D				;

	(4434)	DCS[0,00,0,0,0,0]  BM[1111..00,00..00,00..000..111...0,1,0..0..0...0,0000...0..0000,0...11,101...010,001,011]

	7213:	;(FREE)
	JAMUPP002D:
		P2-T,	SR_ZERO,				;ZERO OUT SR FOR RETURN
		NEXT,	BUTA(RETURN),				;NOW RETURN
			J/BUTERROR7				;ERROR IF GO HERE
```

```
      (7213)  DCS[0,00,0,0,0,0]  BM[0011,,00,00,,00,00,,000,,000,,,0,0,1,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
18845
18846     4756:
18847     JAMUPP003:
18848             SELECT, BUSDIN_CUA=(I),                         ;ENABLE READ JAMUPP MICROADDR
18849             NEXT,   BUT0(ERROR),                            ;WAS ERROR PREVIOUSLY SET ?
18850                     J/JAMUPP005                             ;  YES/JAMUPP004, NO/JAMUPP005
      (4756)  DCS[0,00,0,1,0,0]  BM[0000,,00,00,,01,01,,000,,000,,,0,0,0,,0,,0,,,1,1001,,,0,,0000,0,,,11,000,,,111,100,111]
18851
18852     ;* ENTER HERE IF "ERROR" WAS PREVIOUSLY SET (INHIBIT LOG)
18853     4746:
18854     JAMUPP004:
18855             P3,     CSPD[00]_LOG=CUA,                       ;GET RETURN LOC <= SAVED CUA
18856             P3=T,   BUTA(CUA=TRACK),                        ;START CUA GOING AGAIN
18857             NEXT,   J/JAMUPP002B                            ;INHIBIT LOG IF PREV ERROR
      (4746)  DCS[0,00,0,0,0,0]  BM[0000,,10,00,,00,00,,000,,000,,,1,0,0,,0,,0,,,0,1111,,,1,,0000,0,,,11,001,,,111,101,111]
18858
18859     ;* ENTER HERE IF "ERROR" NOT YET SET (LOG FIRST MODE)
18860     4747:
18861     JAMUPP005:
18862             P0,     DCS=CTR(C0,);                           ;FORCE ERROR WITH PREV ERROR-CODE/ENUA, TNUA=(4747)
18863             P3,     CSPD[00]_LOG=CUA,                       ;RETURN LOC <= RETURN ADDR FROM CUA
18864             NEXT,   J/JAMUPP006                             ;AND GO LOG REGISTERS THIS TIME
      (4747)  DCS[0,00,1,0,0,0]  BM[1111,,10,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,1111,,,1,,0000,0,,,11,000,,,100,011,101]
18865
18866     4435:  ;(FREE)
18867     JAMUPP006:
18868             SELECT, BUSDIN_SERVICE=(I),                     ;SERVICE REGISTER
18869             NEXT,   J/JAMUPP007                             ;
      (4435)  DCS[0,00,0,0,0,0]  BM[0100,,01,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,1,1001,,,0,,0000,0,,,11,000,,,100,011,110]
18870
18871     4436:  ;(FREE)
18872     JAMUPP007:
18873             P3,     CSPD[01]_LOG=SERVICE,                   ;LOG SERVICE INFO REGISTER
18874             NEXT,   J/JAMUPP010                             ;
      (4436)  DCS[0,00,0,0,0,0]  BM[0000,,10,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,1110,,,1,,0000,0,,,11,000,,,100,011,111]
18875
18876     4437:  ;(FREE)
18877     JAMUPP010:
18878             SELECT, BUSDIN_JAM=(I),                         ;JAM REGISTER
18879             NEXT,   J/JAMUPP011                             ;
      (4437)  DCS[0,00,0,0,0,0]  BM[1100,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,1,1001,,,0,,0000,0,,,11,000,,,100,100,000]
18880
18881     4440:  ;(FREE)
18882     JAMUPP011:
18883             P3,     CSPD[02]_LOG=JAM,                       ;LOG JAMUPP CAUSE INFO
18884                     BUTA(CUA=TRACK),                        ;RESET CUA TO TRACK, IN CASE ANOTHER JAM COMES
18885             NEXT,   J/JAMUPP002B                            ;GO TO TOP OF THIS PAGE, AND RETURN INLINE
      (4440)  DCS[0,00,0,0,0,0]  BM[0000,,10,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,1101,,,1,,0000,0,,,11,001,,,111,101,111]
18886
18887
18888
18889
18890
```



```
18891     1.PAGE;*************************************************************************************************************
18892
18893
18894     .TOC    *   MICROBRANCH (BUT) TAKEOFF WORDS
18895
18896     ;*************************************************************************************
18897     ;*                                                                                   *
18898     ;*                                                  UWORDS: 000 + 004                *
18899     ;*                                                                                   *
18900     ;*     FUNCTION:       THESE WORDS ARE THE INACTIVE-"BUT" [BRANCHING TYPE]           *
18901     ;*                     "TAKEOFF", OR SUBROUTINE MICROWORDS.  ANY TEST WHICH          *
18902     ;*                     REQUIRES A SPECIFIC "BUT" CONDITION TO BE TESTED WILL USE     *
18903     ;*                     ONE OF THESE MICROWORDS AS A TAKEOFF POINT INTO THE "BUT"     *
18904     ;*                     TARGET TABLE" [DESCRIBED NEXT], WHERE AN ENUA:TNUA            *
18905     ;*                     COMPARISON WILL HAVE BEEN PREVIOUSLY ENABLED [VIA SET-        *
18906     ;*                     TING THE DIAGNOSTIC COUNTER TO THE APPROPRIATE VALUE.         *
18907     ;*                                                                                   *
18908     ;*************************************************************************************
18909
18910
18911
18912
18913     ;*** BUT 00 ***
18914     ;FULL WIDTH IS BUT(BR<3:0>)
18915     7276:  ;(FREE)
18916     BUTBR3=0:
18917             NEXT,   BUT(BR3=0),                             ;TO (400)=(417), W4
18918                     J/ETARGET400                            ;NO MASK
      (7276)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,00,000,,,100,000,000]
18919
18920
18921     ;*** BUT 01 ***
18922     ;FULL WIDTH IS BUT(IR<15:12>)
18923     7300:  ;(FREE)
18924     BUTIR15=12:
18925             NEXT,   BUT(IR15=12),                           ;TO (400)=(417), W4
18926                     J/ETARGET400                            ;NO MASK
      (7300)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,00,001,,,100,000,000]
18927
18928
18929     ;*** BUT 07 ***
18930     ;FULL WIDTH IS BUT(INSTR 5)
18931     7301:  ;(FREE)
18932     BUTINSTR5:
18933             NEXT,   BUT(INSTR5),                            ;TO (400)=(437), W5
18934                     J/ETARGET400                            ;NO MASK

      (7301)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,00,010,,,100,000,000]
18935
18936
18937     ;*** BUT 03 ***
18938     ;FULL WIDTH IS BUT(IR11:FLTPT<3:0>)
18939     7302:  ;(FREE)
18940     BUTIR11:FLTPT3=0:
18941             NEXT,   BUT(IR11:FLTPT3=0),                     ;TO (400)=(437), W5
```

```
18942                   J/ZTARGET400                            ;NO MASK
      (7302)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,00,011,,,100,000,000]
18943
18944
18945   ;*** BUT 04 ***
18946   ;FULL WIDTH IS BUT[IR<9:6>]
18947   7304:  ;(FREE)
18948   BUT[R9=6:
18949           NEXT,   BUT[IR9=6),                             ;TO (408)=(417), W6
18950                   J/ZTARGET400                            ;NO MASK
      (7304)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,00,100,,,100,000,000]
18951
18952
18953   ;*** BUT 05 ***
18954   ;FULL WIDTH IS BUT[MOV-DR7$IR<5:3>]
18955   7305:  ;(FREE)
18956   BUTMOVDR7IR5=3:
18957           NEXT,   BUT(MOV-DR7$IR5=3),                     ;TO (400)=(417), W6
18958                   J/ZTARGET400                            ;NO MASK
      (7305)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,00,101,,,100,000,000]
18959
18960
18961   ;*** BUT 06 ***
18962   ;FULL WIDTH IS BUT[INSTR 1]          *** N.B.:  THIS BUT IS ALSO ACTIVE ***
18963   7306:  ;(FREE)
18964   BUTINSTR1:
18965           NEXT,   BUTA(INSTR1),                           ;TO (408)=(777), W8
18966                   J/ZTARGET400                            ;NO MASK
      (7306)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,00,110,,,100,000,000]
18967
18968
18969   ;*** BUT 07 ***
18970   ;FULL WIDTH IS BUT[BG=SERV=H+FP=SERV=H$D[C]$FPRET<1:0>]
18971   7307:  ;(FREE)
18972   BUTBGFPSERV:
18973           NEXT,   BUT[BGSERV=FPSERV$D[C]$FPRET),          ;TO (407),(417), W1
18974                   J/ZTARGET407                            ;MASK OUT D[C], FPRET<1:0>
      (7307)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,00,111,,,100,000,111]
18975
18976   7310:  ;(FREE)
18977   BUTD[C]C:
18978           NEXT,   BUT(BGSERV=FPSERV$D[C]$FPRET),          ;TO (415),(417), W1
18979                   J/ZTARGET413                            ;MASK OUT BG=SERV=H+FP=SERV=H, FPRET<1:0>
      (7310)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,00,111,,,100,001,011]
18980
18981
18982   ;*** BUT10 ***
18983   ;FULL WIDTH IS BUT[COUT07$DOUT07$FP505]
18984   7311:  ;(FREE)
18985   BUTCOUT7DOUT71
18986           NEXT,   BUT(COUT07$DOUT07$FP505),               ;TO (401),(403),(405),(407), W2
18987                   J/ZTARGET401                            ;MASK OUT FP504
      (7311)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,01,000,,,100,000,001]
18988
```



```
18989   7312:  ;(FREE)
18990   BUTFP505:
18991           NEXT,   BUT(COUT07$DOUT07$FP505),               ;TO (406),(407), W1
18992                   J/ZTARGET406                            ;MASK OUT COUT7, DOUT7
      (7312)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,01,000,,,100,000,110]
18993
18994
18995   ;*** BUT 11 ***
18996   ;FULL WIDTH IS BUT[DM08$MD0$BYTE]
18997   7313:  ;(FREE)
18998   BUTDM8$MDBYTE:
18999           NEXT,   BUT(DM08$MD0$BYTE),                     ;TO (400)=(407), W3
19000                   J/ZTARGET400                            ;NO MASK
      (7313)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,01,001,,,100,000,000]
19001
19002
19003   ;*** BUT 12 ***
19004   ;FULL WIDTH IS BUT[GD<3:2>]
19005   7314:  ;(FREE)
19006   BUTGD3-2:
19007           NEXT,   BUT(GD3=2),                             ;TO (400)=(403), W2
19008                   J/ZTARGET400                            ;NO MASK
      (7314)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,01,010,,,100,000,000]
19009
19010
19011   ;*** BUT 13 ***
19012   ;FULL WIDTH IS BUT[SR<1:0>$COUNT-1S-377]                ;*** N.B.:  THIS BUT IS ALSO ACTIVE  ***
19013   7315:  ;(FREE)
19014   BUTSR1-0:
19015           NEXT,   BUTA(SR1-0$COUNT-1S-377),               ;TO (401),(403),(405),(407), W2
19016                   J/ZTARGET401                            ;MASK OUT COUNT-1S-377
      (7315)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,01,011,,,100,000,001]
19017
19018
19019   ;*** BUT 14 ***
19020   ;FULL WIDTH IS BUT[BG-SERVCE-L$NFSS$MULTIPLE]
19021   7316:  ;(FREE)
19022   BUTBGSERVL:
19023           NEXT,   BUT(BG-SERVCE-L$NFSS$MULTIPLE),         ;TO (403),(407), W1
19024                   J/ZTARGET403                            ;MASK OUT NFSS, MULTIPLE
      (7316)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,01,100,,,100,000,011]
19025
19026   7317:  ;(FREE)
19027   BUTMMASKPS[7]:
19028           SETUP,  TEST(MASKED-PS[7]),                     ;SELECT MULTIPLE BUT
19029           NEXT,   J/BUTMNXT000                            ;NEXT

      (7317)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,000,,,011,010,000]
19030
19031   7320:  ;(FREE)
19032   BUTMNXT000:
19033           SETUP,  TEST(MASKED-PS[7]),                     ;SELECT MULTIPLE BUT
19034           NEXT,   BUT(BG-SERVCE-L$NFSS$MULTIPLE),         ;TO (406),(407), W1
19035                   J/ZTARGET406                            ;MASK OUT BG-SERVCE-L, NFSS
      (7320)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,01,100,,,100,000,110]
```

```
19036
19037    7321:  ;(FREE)
19038    BUTMD00:
19039            SETUP,  TEST(D00),                          ;SELECT MULTIPLE BUT
19040            NEXT,   J/BUTMNXT001                        ;NEXT
     (7321)  DCS[0,00,0,0,0,0]  BM[0000..00.00..00.00..000..001...0.0.0..0..0...0.0000...0..0000,0...13,000...011,010,010]
19041
19042    7322:  ;(FREE)
19043    BUTMNXT001:
19044            SETUP,  TEST(D00),                          ;SELECT MULTIPLE BUT
19045            NEXT,   BUT(BG-SERVCE-L&MFSS&MULTIPLE1),    ;TO (405),(407), W1
19046                    J/ZTARGET406                        ;MASK OUT BG-SERVCE-L, MFSS
     (7322)  DCS[0,00,0,0,0,0]  BM[0000..00.00..00.00..000..001...0.0.0..0..0...0.0000...0..0000,0...01,100...100,000,110]
19047
19048    7323:  ;(FREE)
19049    BUTMPS[N]:
19050            SETUP,  TEST(PS[N]),                        ;SELECT MULTIPLE BUT
19051            NEXT,   J/BUTMNXT003                        ;NEXT
     (7323)  DCS[0,00,0,0,0,0]  BM[0000..00.00..00.00..000..010...0.0.0..0..0...0.0000...0..0000,0...11,000...011,010,100]
19052
19053    7324:  ;(FREE)
19054    BUTMNXT002:
19055            SETUP,  TEST(PS[N]),                        ;SELECT MULTIPLE BUT
19056            NEXT,   BUT(BG-SERVCE-L&MFSS&MULTIPLE),     ;TO (406),(407), W1
19057                    J/ZTARGET406                        ;MASK OUT BG-SERVCE-L, MFSS
     (7324)  DCS[0,00,0,0,0,0]  BM[0000..00.00..00.00..000..010...0.0.0..0..0...0.0000...0..0000,0...01,100...100,000,110]
19058
19059    7325:  ;(FREE)
19060    BUTMFLAG7:
19061            SETUP,  TEST(FLAG7),                        ;SELECT MULTIPLE BUT
19062            NEXT,   J/BUTMNXT003                        ;NEXT
     (7325)  DCS[0,00,0,0,0,0]  BM[0000..00.00..00.00..000..011...0.0.0..0..0...0.0000...0..0000,0...11,000...011,010,110]
19063
19064    7326:  ;(FREE)
19065    BUTMNXT003:
19066            SETUP,  TEST(FLAG7),                        ;SELECT MULTIPLE BUT
19067            NEXT,   BUT(BG-SERVCE-L&MFSS&MULTIPLE),     ;TO (406),(407), W1
19068                    J/ZTARGET406                        ;MASK OUT BG-SERVCE-L, MFSS
     (7326)  DCS[0,00,0,0,0,0]  BM[0000..00.00..00.00..000..011...0.0.0..0..0...0.0000...0..0000,0...01,100...100,000,110]
19069
19070    7327:  ;(FREE)
19071    BUTMEXFLAG1:
19072            SETUP,  TEST(EXFLAG1),                      ;SELECT MULTIPLE BUT
19073            NEXT,   J/BUTMNXT004                        ;NEXT
     (7327)  DCS[0,00,0,0,0,0]  BM[0000..00.00..00.00..000..100...0.0.0..0..0...0.0000...0..0000,0...11,000...011,011,000]
19074
19075    7330:  ;(FREE)
19076    BUTMNXT004:
19077            SETUP,  TEST(EXFLAG1),                      ;SELECT MULTIPLE BUT
19078            NEXT,   BUT(BG-SERVCE-L&MFSS&MULTIPLE),     ;TO (406),(407), W1
19079                    J/ZTARGET406                        ;MASK OUT BG-SERVCE-L, MFSS
     (7330)  DCS[0,00,0,0,0,0]  BM[0000..00.00..00.00..000..100...0.0.0..0..0...0.0000...0..0000,0...01,100...100,000,110]
19080
19081    7331:  ;(FREE)
```



```
19082    BUTMFLTPTS:
19083            SETUP,  TEST(FLTPTS),                       ;SELECT MULTIPLE BUT
19084            NEXT,   J/BUTMNXT005                        ;NEXT
     (7331)  DCS[0,00,0,0,0,0]  BM[0000..00.00..00.00..000..101...0.0.0..0..0...0.0000...0..0000,0...11,000...011,011,010]
19085
19086    7332:  ;(FREE)
19087    BUTMNXT005:
19088            SETUP,  TEST(FLTPTS),                       ;SELECT MULTIPLE BUT
19089            NEXT,   BUT(BG-SERVCE-L&MFSS&MULTIPLE),     ;TO (406),(407), W1
19090                    J/ZTARGET406                        ;MASK OUT BG-SERVCE-L, MFSS
     (7332)  DCS[0,00,0,0,0,0]  BM[0000..00.00..00.00..000..101...0.0.0..0..0...0.0000...0..0000,0...01,100...100,000,110]
19091
19092    7333:  ;(FREE)
19093    BUTMEXFLAG2:
19094            SETUP,  TEST(EXFLAG2),                      ;SELECT MULTIPLE BUT
19095            NEXT,   J/BUTMNXT006                        ;NEXT
     (7333)  DCS[0,00,0,0,0,0]  BM[0000..00.00..00.00..000..110...0.0.0..0..0...0.0000...0..0000,0...11,000...011,011,100]
19096
19097    7334:  ;(FREE)
19098    BUTMNXT006:
19099            SETUP,  TEST(FXFLAG2),                      ;SELECT MULTIPLE BUT
19100            NEXT,   BUT(BG-SERVCE-L&MFSS&MULTIPLE),     ;TO (406),(407), W1
19101                    J/ZTARGET406                        ;MASK OUT BG-SERVCE-L, MFSS
     (7334)  DCS[0,00,0,0,0,0]  BM[0000..00.00..00.00..000..110...0.0.0..0..0...0.0000...0..0000,0...01,100...100,000,110]
19102
19103
19104    7335:  ;(FREE)
19105    BUTMINITJAM:
19106            SETUP,  TEST(INIT-JAM),                     ;SELECT MULTIPLE BUT
19107            NEXT,   J/BUTMNXT007                        ;NEXT
     (7335)  DCS[0,00,0,0,0,0]  BM[0000..00.00..00.00..000..111...0.0.0..0..0...0.0000...0..0000,0...11,000...011,100,000]
19108
19109    7340:  ;(FREE)
19110    BUTMNXT007:
19111            SETUP,  TEST(INIT-JAM),                     ;SELECT MULTIPLE BUT
19112            NEXT,   BUT(BG-SERVCE-L&MFSS&MULTIPLE),     ;TO (406),(407), W1
19113                    J/ZTARGET406                        ;MASK OUT BG-SERVCE-L, MFSS
     (7340)  DCS[0,00,0,0,0,0]  BM[0000..00.00..00.00..000..111...0.0.0..0..0...0.0000...0..0000,0...01,100...100,000,110]
19114
19115    ;*** BUT 15 ***
19116    ;FULL WIDTH IS BUT[0<14:00>=D&D15]
19117    7341:  ;(FREE)
19118    BUTD-IS-ZERO:
19119            NEXT,   BUT(D14-00&D&D15),                  ;TO (400)-(403),W2
19120                    J/ZTARGET400                        ;NO MASK
     (7341)  DCS[0,00,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0..0..0...0.0000...0..0000,0...01,101...100,000,000]

19121
19122
19123    ;*** BUT 16 ***
19124    ;FULL WIDTH IS BUT(IR11&PS15]
19125    7342:  ;(FREE)
19126    BUTIR11S:
19127            NEXT,   BUT(IR11&PS15),                     ;TO (401),(403), W1
19128                    J/ZTARGET401                        ;MASK OUT PS15
```

```
      (7342)  DCS[0,00,0,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0..0..0...0.0000...0..0000.0...01.110...100.000.001]
19129
19130     7343:  ;(FREE)
19131     BUTP819:
19132             NEXT,   BUT(IR11*PP815),                        ;TO (402),(403), M1
19133                     J/ZTARGET402                            ;MASK OUT IR11
      (7343)  DCS[0,00,0,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0..0..0...0.0000...0..0000.0...01.110...100.000.010]
19134
19135
19136     ;*** BUT 17 ***
19137     ;FULL WIDTH IS BUT(COUNT-I8=377*D[C])          *** N.B.: THIS BUT IS ALSO ACTIVE  ***
19138     7344:  ;(FREE)
19139     BUTDIC]A:
19140             NEXT,   BUTA(COUNT-I8-377*D[C]),                ;TO (402),(403), M1
19141                     J/ZTARGET402                            ;MASK OUT COUNT-I8=377
      (7344)  DCS[0,00,0,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0..0..0...0.0000...0..0000.0...01.111...100.000.010]
19142
19143
19144     ;*** BUT 20 ***
19145     ;FULL WIDTH IS BUT(PREFETCH-L*SERVICE)          *** N.B.: THIS BUT IS ALSO ACTIVE ***
19146     7345:  ;(FREE)
19147     BUTSERVICE:
19148             NEXT,   BUTA(PREFETCH-L*SERVICE),               ;TO (402),(403), M1
19149                     J/ZTARGET402                            ;MASK OUT PREFETCH-L
      (7345)  DCS[0,00,0,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0..0..0...0.0000...0..0000.0...10.000...100.000.010]
19150
19151
19152     ;*** BUT 21 ***
19153     ;FULL WIDTH IS BUT(VECTOR-LOAD*DR6/7L]
19154     7346:  ;(FREE)
19155     BUTVECTLOAD:
19156             NEXT,   BUT(VECTOR-LOAD*DR6-7L),                ;TO (401), (403), M1
19157                     J/ZTARGET401                            ;MASK OUT DR6/7-L
      (7346)  DCS[0,00,0,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0..0..0...0.0000...0..0000.0...10.001...100.000.001]
19158
19159     7347:  ;(FREE)
19160     BUTDR6-7L:
19161             NEXT,   BUT(VECTOR-LOAD*DR6-7L),                ;TO (402), (403), M1
19162                     J/ZTARGET402                            ;MASK OUT VECTOR-LOAD
      (7347)  DCS[0,00,0,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0..0..0...0.0000...0..0000.0...10.001...100.000.010]
19163
19164
19165     ;*** BUT 22 ***
19166     ;THIS IS AN ACTIVE BUT - NO BRANCH MODIFICATION
19167     ;[THIS BUT IS NOT USED WITH THE TARGET TABLE]
19168
19169
19170     ;*** BUT 23 ***
19171     ;FULL WIDTH IS BUT[D[C]*BA00]
19172     7349:  ;(FREE)
19173     BUTD[C]B:
19174             NEXT,   BUT(D[C]*BA00),                         ;TO (401),(403), M1
19175                     J/ZTARGET401                            ;MASK OUT BA00
      (7350)  DCS[0,00,0,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0..0..0...0.0000...0..0000.0...10.011...100.000.001]
```



```
19176
19177     7351:  ;(FREE)
19178     BUTBA00:
19179             NEXT,   BUT(D[C]*BA00),                         ;TO (402),(403), M1
19180                     J/ZTARGET402                            ;MASK OUT D[C]
      (7351)  DCS[0,00,0,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0..0..0...0.0000...0..0000.0...10.011...100.000.010]
19181
19182
19183     ;*** BUT 24 ***
19184     ;FULL WIDTH IS BUT[OTHER-JAM*FP-PROC]
19185     7352:  ;(FREE)
19186     BUTOTHERJAM:
19187             NEXT,   BUT(OTHER-JAM*FP-PROC),                 ;TO (401),(403), M1
19188                     J/ZTARGET401                            ;MASK OUT FLTPT-PROC-H
      (7352)  DCS[0,00,0,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0..0..0...0.0000...0..0000.0...10.100...100.000.001]
19189
19190     7354:  ;(FREE)
19191     BUTFPPROC:
19192             NEXT,   BUT(OTHER-JAM*FP-PROC),                 ;TO (402),(403), M1
19193                     J/ZTARGET402                            ;MASK OUT OTHER-JAM
      (7354)  DCS[0,00,0,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0..0..0...0.0000...0..0000.0...10.100...100.000.010]
19194
19195
19196     ;*** BUT 25 ***
19197     ;FULL WIDTH IS BUT[COUNT-I8-377]                *** N.B.: THIS BUT IS ALSO ACTIVE ***
19198     ;[THIS BUT IS NOT USED WITH THE TARGET TABLE]
19199
19200
19201     ;*** BUT 26 ***
19202     ;FULL WIDTH IS BUT[INTR-HIGH*INSTR-BRANCH-L]
19203     7355:  ;(FREE)
19204     BUTINTRHIGH:
19205             NEXT,   BUT(INTR-HIGH*INSTR-BRANCH-L),          ;TO (401),(403), M1
19206                     J/ZTARGET401                            ;MASK OUT INSTR-BRANCH-L
      (7355)  DCS[0,00,0,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0..0..0...0.0000...0..0000.0...10.110...100.000.001]
19207
19208     7356:  ;(FREE)
19209     BUTINSTRBRANCH:
19210             NEXT,   BUT(INTR-HIGH*INSTR-BRANCH-L),          ;TO (402),(403), M1
19211                     J/ZTARGET402                            ;MASK OUT INTR-HIGH-H
      (7356)  DCS[0,00,0,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0..0..0...0.0000...0..0000.0...10.110...100.000.010]
19212
19213
19214     ;*** BUT 27 ***
19215     ;FULL WIDTH IS BUT(PREFETCH-JAM*FP-FD]
19216     7360:  ;(FREF)

19217     BUTPREFETCHJAM:
19218             NEXT,   BUT(PREFETCH-JAM*FP-FD),                ;TO (401),(403), M1
19219                     J/ZTARGET401                            ;MASK OUT FLTPT-FD-H
      (7360)  DCS[0,00,0,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0..0..0...0.0000...0..0000.0...10.111...100.000.001]
19220
19221     7371:  ;(FREE)
19222     BUTFPFD:
19223             NEXT,   BUT(PREFETCH-JAM*FP-FD),                ;TO (402),(403), M1
```

```
19224                   J/ZTARGET402                            ;MASK OUT PREFETCH-JAM-H
      (7371)  DCS[0,00,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0...0...0.0000...0..0000.0...10.111...100.000.010]
19225
19226
19227
19228
19229
19230   1.PAGE=========================================================================================================
19231
19232
19233   .TOC    *   MICROBRANCH (BUT) TARGET WORDS
19234
19235   !*************************************************************************************
19236   !*                                                                                   *
19237   !*                                                UWORDS: 000 + 400                  *
19238   !*                                                                                   *
19239   !*     FUNCTION:       TARGET BUT TABLE                                              *
19240   !*                     ALL THE ABOVE BUTS TARGET INTO THIS TABLE OF MICROWORDS,      *
19241   !*                     ALL OF WHICH DO A BUTA(RETURN).  IN THIS MANNER, ANY OF       *
19242   !*                     THE ABOVE BRANCHES MAY BE EXECUTED, AND CONTROL WILL ALWAYS   *
19243   !*                     RETURN TO WHERE THE "BUT TBL" SUBROUTINE WAS CALLED.          *
19244   !*                                                                                   *
19245   !*************************************************************************************
19246
19247
19248
19249   !*** THE TARGET BUT TABLE ***
19250
19251   7400:
19252   ZTARGET400:   NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7400)  DCS[0,00,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0...0...0.0000...0..0000.0...11.111...011.111.110]
19253   7401:
19254   ZTARGET401:   NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7401)  DCS[0,00,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0...0...0.0000...0..0000.0...11.111...011.111.110]
19255   7402:
19256   ZTARGET402:   NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7402)  DCS[0,00,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0...0...0.0000...0..0000.0...11.111...011.111.110]
19257   7403:
19258   ZTARGET403:   NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7403)  DCS[0,00,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0...0...0.0000...0..0000.0...11.111...011.111.110]
19259   7404:
19260   ZTARGET404:   NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7404)  DCS[0,00,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0...0...0.0000...0..0000.0...11.111...011.111.110]
19261   7405:
19262   ZTARGET405:   NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7405)  DCS[0,00,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0...0...0.0000...0..0000.0...11.111...011.111.110]
19263   7406:
19264   ZTARGET406:   NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7406)  DCS[0,00,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0...0...0.0000...0..0000.0...11.111...011.111.110]
19265   7407:
19266   ZTARGET407:   NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7407)  DCS[0,00,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0...0...0.0000...0..0000.0...11.111...011.111.110]
19267   7410:
19268   ZTARGET410:   NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
```



```
      (7410)  DCS[0,00,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0...0...0.0000...0..0000.0...11.111...011.111.110]
19269   7411:
19270   ZTARGET411:   NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7411)  DCS[0,00,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0...0...0.0000...0..0000.0...11.111...011.111.110]
19271   7412:
19272   ZTARGET412:   NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7412)  DCS[0,00,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0...0...0.0000...0..0000.0...11.111...011.111.110]
19273   7413:
19274   ZTARGET413:   NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7413)  DCS[0,00,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0...0...0.0000...0..0000.0...11.111...011.111.110]
19275   7414:
19276   ZTARGET414:   NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7414)  DCS[0,00,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0...0...0.0000...0..0000.0...11.111...011.111.110]
19277   7415:
19278   ZTARGET415:   NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7415)  DCS[0,00,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0...0...0.0000...0..0000.0...11.111...011.111.110]
19279   7416:
19280   ZTARGET416:   NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7416)  DCS[0,00,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0...0...0.0000...0..0000.0...11.111...011.111.110]
19281   7417:
19282   ZTARGET417:   NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7417)  DCS[0,00,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0...0...0.0000...0..0000.0...11.111...011.111.110]
19283   7420:
19284   ZTARGET420:   NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7420)  DCS[0,00,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0...0...0.0000...0..0000.0...11.111...011.111.110]
19285   7421:
19286   ZTARGET421:   NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7421)  DCS[0,00,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0...0...0.0000...0..0000.0...11.111...011.111.110]
19287   7422:
19288   ZTARGET422:   NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7422)  DCS[0,00,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0...0...0.0000...0..0000.0...11.111...011.111.110]
19289   7423:
19290   ZTARGET423:   NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7423)  DCS[0,00,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0...0...0.0000...0..0000.0...11.111...011.111.110]
19291   7424:
19292   ZTARGET424:   NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7424)  DCS[0,00,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0...0...0.0000...0..0000.0...11.111...011.111.110]
19293   7425:
19294   ZTARGET425:   NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7425)  DCS[0,00,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0...0...0.0000...0..0000.0...11.111...011.111.110]
19295   7426:
19296   ZTARGET426:   NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7426)  DCS[0,00,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0...0...0.0000...0..0000.0...11.111...011.111.110]
19297   7427:
19298   ZTARGET427:   NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7427)  DCS[0,00,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0...0...0.0000...0..0000.0...11.111...011.111.110]

19299   7430:
19300   ZTARGET430:   NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7430)  DCS[0,00,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0...0...0.0000...0..0000.0...11.111...011.111.110]
19301   7431:
19302   ZTARGET431:   NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7431)  DCS[0,00,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0...0...0.0000...0..0000.0...11.111...011.111.110]
19303   7432:
19304   ZTARGET432:   NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
```

```
      (7432)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19305    7433;
19306    ZTARGET433:   NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7433)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19307    7434;
19308    ZTARGET434:   NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7434)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19309    7435;
19310    ZTARGET435:   NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7435)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19311    7436;
19312    ZTARGET436:   NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7436)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19313    7437;
19314    ZTARGET437:   NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7437)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19315    7440;
19316    ZTARGET440:   NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7440)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19317    7441;
19318    ZTARGET441:   NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7441)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19319    7442;
19320    ZTARGET442:   NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7442)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19321    7443;
19322    ZTARGET443:   NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7443)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19323    7444;
19324    ZTARGET444:   NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7444)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19325    7445;
19326    ZTARGET445:   NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7445)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19327    7446;
19328    ZTARGET446:   NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7446)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19329    7447;
19330    ZTARGET447:   NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7447)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19331    7450;
19332    ZTARGET450:   NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7450)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19333    7451;
19334    ZTARGET451:   NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7451)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19335    7452;
19336    ZTARGET452:   NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7452)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19337    7453;
19338    ZTARGET453:   NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7453)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19339    7454;
19340    ZTARGET454:   NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
```



```
      (7454)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19341    7455;
19342    ZTARGET455:   NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7455)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19343    7456;
19344    ZTARGET456:   NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7456)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19345    7457;
19346    ZTARGET457:   NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7457)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19347    7460;
19348    ZTARGET460:   NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7460)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19349    7461;
19350    ZTARGET461:   NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7461)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19351    7462;
19352    ZTARGET462:   NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7462)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19353    7463;
19354    ZTARGET463:   NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7463)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19355    7464;
19356    ZTARGET464:   NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7464)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19357    7465;
19358    ZTARGET465:   NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7465)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19359    7466;
19360    ZTARGET466:   NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7466)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19361    7467;
19362    ZTARGET467:   NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7467)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19363    7470;
19364    ZTARGET470:   NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7470)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19365    7471;
19366    ZTARGET471:   NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7471)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19367    7472;
19368    ZTARGET472:   NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7472)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19369    7473;
19370    ZTARGET473:   NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7473)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]

19371    7474;
19372    ZTARGET474:   NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7474)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19373    7475;
19374    ZTARGET475:   NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7475)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19375    7476;
19376    ZTARGET476:   NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
```

```
      (7476)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,0000...0,,0000,0,,,11,111,,,011,111,110]
19377    7477;
19378    ZTARGET477:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7477)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,0000...0,,0000,0,,,11,111,,,011,111,110]
19379    7500;
19380    ZTARGET500:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7500)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,0000...0,,0000,0,,,11,111,,,011,111,110]
19381    7501;
19382    ZTARGET501:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7501)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,0000...0,,0000,0,,,11,111,,,011,111,110]
19383    7502;
19384    ZTARGET502:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7502)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,0000...0,,0000,0,,,11,111,,,011,111,110]
19385    7503;
19386    ZTARGET503:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7503)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,0000...0,,0000,0,,,11,111,,,011,111,110]
19387    7504;
19388    ZTARGET504:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7504)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,0000...0,,0000,0,,,11,111,,,011,111,110]
19389    7505;
19390    ZTARGET505:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7505)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,0000...0,,0000,0,,,11,111,,,011,111,110]
19391    7506;
19392    ZTARGET506:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7506)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,0000...0,,0000,0,,,11,111,,,011,111,110]
19393    7507;
19394    ZTARGET507:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7507)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,0000...0,,0000,0,,,11,111,,,011,111,110]
19395    7510;
19396    ZTARGET510:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7510)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,0000...0,,0000,0,,,11,111,,,011,111,110]
19397    7511;
19398    ZTARGET511:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7511)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,0000...0,,0000,0,,,11,111,,,011,111,110]
19399    7512;
19400    ZTARGET512:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7512)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,0000...0,,0000,0,,,11,111,,,011,111,110]
19401    7513;
19402    ZTARGET513:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7513)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,0000...0,,0000,0,,,11,111,,,011,111,110]
19403    7514;
19404    ZTARGET514:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7514)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,0000...0,,0000,0,,,11,111,,,011,111,110]
19405    7515;
19406    ZTARGET515:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7515)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,0000...0,,0000,0,,,11,111,,,011,111,110]
19407    7516;
19408    ZTARGET516:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7516)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,0000...0,,0000,0,,,11,111,,,011,111,110]
19409    7517;
19410    ZTARGET517:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7517)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,0000...0,,0000,0,,,11,111,,,011,111,110]
19411    7520;
19412    ZTARGET520:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
```



```
      (7520)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,0000...0,,0000,0,,,11,111,,,011,111,110]
19413    7521;
19414    ZTARGET521:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7521)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,0000...0,,0000,0,,,11,111,,,011,111,110]
19415    7522;
19416    ZTARGET522:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7522)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,0000...0,,0000,0,,,11,111,,,011,111,110]
19417    7523;
19418    ZTARGET523:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7523)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,0000...0,,0000,0,,,11,111,,,011,111,110]
19419    7524;
19420    ZTARGET524:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7524)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,0000...0,,0000,0,,,11,111,,,011,111,110]
19421    7525;
19422    ZTARGET525:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7525)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,0000...0,,0000,0,,,11,111,,,011,111,110]
19423    7526;
19424    ZTARGET526:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7526)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,0000...0,,0000,0,,,11,111,,,011,111,110]
19425    7527;
19426    ZTARGET527:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7527)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,0000...0,,0000,0,,,11,111,,,011,111,110]
19427    7530;
19428    ZTARGET530:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7530)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,0000...0,,0000,0,,,11,111,,,011,111,110]
19429    7531;
19430    ZTARGET531:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7531)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,0000...0,,0000,0,,,11,111,,,011,111,110]
19431    7532;
19432    ZTARGET532:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7532)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,0000...0,,0000,0,,,11,111,,,011,111,110]
19433    7533;
19434    ZTARGET533:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7533)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,0000...0,,0000,0,,,11,111,,,011,111,110]
19435    7534;
19436    ZTARGET534:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7534)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,0000...0,,0000,0,,,11,111,,,011,111,110]
19437    7535;
19438    ZTARGET535:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7535)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,0000...0,,0000,0,,,11,111,,,011,111,110]
19439    7536;
19440    ZTARGET536:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7536)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,0000...0,,0000,0,,,11,111,,,011,111,110]
19441    7537;
19442    ZTARGET537:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7537)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,0000...0,,0000,0,,,11,111,,,011,111,110]

19443    7540;
19444    ZTARGET540:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7540)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,0000...0,,0000,0,,,11,111,,,011,111,110]
19445    7541;
19446    ZTARGET541:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7541)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,0000...0,,0000,0,,,11,111,,,011,111,110]
19447    7542;
19448    ZTARGET542:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
```

```
      (7542)  DCS[0,00,0,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19449     7543;
19450     ZTARGET543:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7543)  DCS[0,00,0,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19451     7544;
19452     ZTARGET544:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7544)  DCS[0,00,0,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19453     7545;
19454     ZTARGET545:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7545)  DCS[0,00,0,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19455     7546;
19456     ZTARGET546:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7546)  DCS[0,00,0,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19457     7547;
19458     ZTARGET547:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7547)  DCS[0,00,0,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19459     7548;
19460     ZTARGET550:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7550)  DCS[0,00,0,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19461     7551;
19462     ZTARGET551:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7551)  DCS[0,00,0,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19463     7552;
19464     ZTARGET552:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7552)  DCS[0,00,0,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19465     7553;
19466     ZTARGET553:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7553)  DCS[0,00,0,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19467     7554;
19468     ZTARGET554:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7554)  DCS[0,00,0,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19469     7555;
19470     ZTARGET555:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7555)  DCS[0,00,0,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19471     7556;
19472     ZTARGET556:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7556)  DCS[0,00,0,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19473     7557;
19474     ZTARGET557:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7557)  DCS[0,00,0,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19475     7560;
19476     ZTARGET560:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7560)  DCS[0,00,0,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19477     7561;
19478     ZTARGET561:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7561)  DCS[0,00,0,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19479     7562;
19480     ZTARGET562:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7562)  DCS[0,00,0,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19481     7563;
19482     ZTARGET563:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7563)  DCS[0,00,0,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19483     7564;
19484     ZTARGET564:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
```



```
      (7564)  DCS[0,00,0,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19485     7565;
19486     ZTARGET565:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7565)  DCS[0,00,0,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19487     7566;
19488     ZTARGET566:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7566)  DCS[0,00,0,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19489     7567;
19490     ZTARGET567:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7567)  DCS[0,00,0,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19491     7570;
19492     ZTARGET570:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7570)  DCS[0,00,0,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19493     7571;
19494     ZTARGET571:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7571)  DCS[0,00,0,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19495     7572;
19496     ZTARGET572:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7572)  DCS[0,00,0,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19497     7573;
19498     ZTARGET573:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7573)  DCS[0,00,0,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19499     7574;
19500     ZTARGET574:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7574)  DCS[0,00,0,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19501     7575;
19502     ZTARGET575:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7575)  DCS[0,00,0,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19503     7576;
19504     ZTARGET576:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7576)  DCS[0,00,0,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19505     7577;
19506     ZTARGET577:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7577)  DCS[0,00,0,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19507     7600;
19508     ZTARGET600:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7600)  DCS[0,00,0,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19509     7601;
19510     ZTARGET601:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7601)  DCS[0,00,0,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19511     7602;
19512     ZTARGET602:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7602)  DCS[0,00,0,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19513     7603;
19514     ZTARGET603:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7603)  DCS[0,00,0,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19515     7604;
19516     ZTARGET604:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7604)  DCS[0,00,0,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19517     7605;
19518     ZTARGET605:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7605)  DCS[0,00,0,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000,,,0,0,0,,0,,0,,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19519     7606;
19520     ZTARGET606:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
```

```
      (7606)  DCS[0,00,0,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0..0..0...0.0000...0..0000.0...11.111...011.111.110]
19521     7607;
19522     ZTARGET607:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7607)  DCS[0,00,0,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0..0..0...0.0000...0..0000.0...11.111...011.111.110]
19523     7610;
19524     ZTARGET610:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7610)  DCS[0,00,0,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0..0..0...0.0000...0..0000.0...11.111...011.111.110]
19525     7611;
19526     ZTARGET611:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7611)  DCS[0,00,0,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0..0..0...0.0000...0..0000.0...11.111...011.111.110]
19527     7612;
19528     ZTARGET612:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7612)  DCS[0,00,0,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0..0..0...0.0000...0..0000.0...11.111...011.111.110]
19529     7613;
19530     ZTARGET613:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7613)  DCS[0,00,0,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0..0..0...0.0000...0..0000.0...11.111...011.111.110]
19531     7614;
19532     ZTARGET614:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7614)  DCS[0,00,0,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0..0..0...0.0000...0..0000.0...11.111...011.111.110]
19533     7615;
19534     ZTARGET615:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7615)  DCS[0,00,0,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0..0..0...0.0000...0..0000.0...11.111...011.111.110]
19535     7616;
19536     ZTARGET616:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7616)  DCS[0,00,0,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0..0..0...0.0000...0..0000.0...11.111...011.111.110]
19537     7617;
19538     ZTARGET617:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7617)  DCS[0,00,0,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0..0..0...0.0000...0..0000.0...11.111...011.111.110]
19539     7620;
19540     ZTARGET620:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7620)  DCS[0,00,0,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0..0..0...0.0000...0..0000.0...11.111...011.111.110]
19541     7621;
19542     ZTARGET621:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7621)  DCS[0,00,0,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0..0..0...0.0000...0..0000.0...11.111...011.111.110]
19543     7622;
19544     ZTARGET622:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7622)  DCS[0,00,0,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0..0..0...0.0000...0..0000.0...11.111...011.111.110]
19545     7623;
19546     ZTARGET623:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7623)  DCS[0,00,0,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0..0..0...0.0000...0..0000.0...11.111...011.111.110]
19547     7624;
19548     ZTARGET624:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7624)  DCS[0,00,0,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0..0..0...0.0000...0..0000.0...11.111...011.111.110]
19549     7625;
19550     ZTARGET625:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7625)  DCS[0,00,0,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0..0..0...0.0000...0..0000.0...11.111...011.111.110]
19551     7626;
19552     ZTARGET626:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7626)  DCS[0,00,0,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0..0..0...0.0000...0..0000.0...11.111...011.111.110]
19553     7627;
19554     ZTARGET627:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7627)  DCS[0,00,0,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0..0..0...0.0000...0..0000.0...11.111...011.111.110]
19555     7630;
19556     ZTARGET630:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
```



```
      (7630)  DCS[0,00,0,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0..0..0...0.0000...0..0000.0...11.111...011.111.110]
19557     7631;
19558     ZTARGET631:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7631)  DCS[0,00,0,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0..0..0...0.0000...0..0000.0...11.111...011.111.110]
19559     7632;
19560     ZTARGET632:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7632)  DCS[0,00,0,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0..0..0...0.0000...0..0000.0...11.111...011.111.110]
19561     7633;
19562     ZTARGET633:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7633)  DCS[0,00,0,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0..0..0...0.0000...0..0000.0...11.111...011.111.110]
19563     7634;
19564     ZTARGET634:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7634)  DCS[0,00,0,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0..0..0...0.0000...0..0000.0...11.111...011.111.110]
19565     7635;
19566     ZTARGET635:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7635)  DCS[0,00,0,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0..0..0...0.0000...0..0000.0...11.111...011.111.110]
19567     7636;
19568     ZTARGET636:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7636)  DCS[0,00,0,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0..0..0...0.0000...0..0000.0...11.111...011.111.110]
19569     7637;
19570     ZTARGET637:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7637)  DCS[0,00,0,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0..0..0...0.0000...0..0000.0...11.111...011.111.110]
19571     7640;
19572     ZTARGET640:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7640)  DCS[0,00,0,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0..0..0...0.0000...0..0000.0...11.111...011.111.110]
19573     7641;
19574     ZTARGET641:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7641)  DCS[0,00,0,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0..0..0...0.0000...0..0000.0...11.111...011.111.110]
19575     7642;
19576     ZTARGET642:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7642)  DCS[0,00,0,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0..0..0...0.0000...0..0000.0...11.111...011.111.110]
19577     7643;
19578     ZTARGET643:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7643)  DCS[0,00,0,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0..0..0...0.0000...0..0000.0...11.111...011.111.110]
19579     7644;
19580     ZTARGET644:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7644)  DCS[0,00,0,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0..0..0...0.0000...0..0000.0...11.111...011.111.110]
19581     7645;
19582     ZTARGET645:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7645)  DCS[0,00,0,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0..0..0...0.0000...0..0000.0...11.111...011.111.110]
19583     7646;
19584     ZTARGET646:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7646)  DCS[0,00,0,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0..0..0...0.0000...0..0000.0...11.111...011.111.110]
19585     7647;
19586     ZTARGET647:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7647)  DCS[0,00,0,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0..0..0...0.0000...0..0000.0...11.111...011.111.110]
19587     7650;
19588     ZTARGET650:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7650)  DCS[0,00,0,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0..0..0...0.0000...0..0000.0...11.111...011.111.110]
19589     7651;
19590     ZTARGET651:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7651)  DCS[0,00,0,0,0,0,0]  BM[0000..00.00..00.00..000..000...0.0.0..0..0...0.0000...0..0000.0...11.111...011.111.110]
19591     7652;
19592     ZTARGET652:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
```

```
      (7652)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19593     7653;
19594     ZTARGET653:    NEXT,   BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7653)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19595     7654;
19596     ZTARGET654:    NEXT,   BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7654)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19597     7655;
19598     ZTARGET655:    NEXT,   BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7655)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19599     7656;
19600     ZTARGET656:    NEXT,   BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7656)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19601     7657;
19602     ZTARGET657:    NEXT,   BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7657)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19603     7660;
19604     ZTARGET660:    NEXT,   BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7660)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19605     7661;
19606     ZTARGET661:    NEXT,   BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7661)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19607     7662;
19608     ZTARGET662:    NEXT,   BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7662)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19609     7663;
19610     ZTARGET663:    NEXT,   BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7663)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19611     7664;
19612     ZTARGET664:    NEXT,   BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7664)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19613     7665;
19614     ZTARGET665:    NEXT,   BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7665)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19615     7666;
19616     ZTARGET666:    NEXT,   BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7666)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19617     7667;
19618     ZTARGET667:    NEXT,   BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7667)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19619     7670;
19620     ZTARGET670:    NEXT,   BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7670)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19621     7671;
19622     ZTARGET671:    NEXT,   BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7671)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19623     7672;
19624     ZTARGET672:    NEXT,   BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7672)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19625     7673;
19626     ZTARGET673:    NEXT,   BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7673)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19627     7674;
19628     ZTARGET674:    NEXT,   BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
```



```
      (7674)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19629     7675;
19630     ZTARGET675:    NEXT,   BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7675)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19631     7676;
19632     ZTARGET676:    NEXT,   BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7676)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19633     7677;
19634     ZTARGET677:    NEXT,   BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7677)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19635     7700;
19636     ZTARGET700:    NEXT,   BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7700)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19637     7701;
19638     ZTARGET701:    NEXT,   BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7701)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19639     7702;
19640     ZTARGET702:    NEXT,   BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7702)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19641     7703;
19642     ZTARGET703:    NEXT,   BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7703)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19643     7704;
19644     ZTARGET704:    NEXT,   BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7704)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19645     7705;
19646     ZTARGET705:    NEXT,   BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7705)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19647     7706;
19648     ZTARGET706:    NEXT,   BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7706)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19649     7707;
19650     ZTARGET707:    NEXT,   BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7707)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19651     7710;
19652     ZTARGET710:    NEXT,   BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7710)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19653     7711;
19654     ZTARGET711:    NEXT,   BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7711)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19655     7712;
19656     ZTARGET712:    NEXT,   BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7712)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19657     7713;
19658     ZTARGET713:    NEXT,   BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7713)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]

19659     7714;
19660     ZTARGET714:    NEXT,   BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7714)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19661     7715;
19662     ZTARGET715:    NEXT,   BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7715)  DCS[0,00,0,0,0,0]  BM[0000,,00,00,,00,00,,000,,000...0,0,0,,0,,0...0,,0,0000,,,0,,0000,0,,,11,111,,,011,111,110]
19663     7716;
19664     ZTARGET716:    NEXT,   BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
```

```
      (7716)  DCS[0,00,0,0,0,0]  BM[0000..00,00..00,00..000..000...0,0,0..0..0...0,0000...0..0000,0...11,111...011,111,110]
19665     7717;
19666     ZTARGET7717:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7717)  DCS[0,00,0,0,0,0]  BM[0000..00,00..00,00..000..000...0,0,0..0..0...0,0000...0..0000,0...11,111...011,111,110]
19667     7720;
19668     ZTARGET7720:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7720)  DCS[0,00,0,0,0,0]  BM[0000..00,00..00,00..000..000...0,0,0..0..0...0,0000...0..0000,0...11,111...011,111,110]
19669     7721;
19670     ZTARGET7721:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7721)  DCS[0,00,0,0,0,0]  BM[0000..00,00..00,00..000..000...0,0,0..0..0...0,0000...0..0000,0...11,111...011,111,110]
19671     7722;
19672     ZTARGET7722:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7722)  DCS[0,00,0,0,0,0]  BM[0000..00,00..00,00..000..000...0,0,0..0..0...0,0000...0..0000,0...11,111...011,111,110]
19673     7723;
19674     ZTARGET7723:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7723)  DCS[0,00,0,0,0,0]  BM[0000..00,00..00,00..000..000...0,0,0..0..0...0,0000...0..0000,0...11,111...011,111,110]
19675     7724;
19676     ZTARGET7724:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7724)  DCS[0,00,0,0,0,0]  BM[0000..00,00..00,00..000..000...0,0,0..0..0...0,0000...0..0000,0...11,111...011,111,110]
19677     7725;
19678     ZTARGET7725:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7725)  DCS[0,00,0,0,0,0]  BM[0000..00,00..00,00..000..000...0,0,0..0..0...0,0000...0..0000,0...11,111...011,111,110]
19679     7726;
19680     ZTARGET7726:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7726)  DCS[0,00,0,0,0,0]  BM[0000..00,00..00,00..000..000...0,0,0..0..0...0,0000...0..0000,0...11,111...011,111,110]
19681     7727;
19682     ZTARGET7727:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7727)  DCS[0,00,0,0,0,0]  BM[0000..00,00..00,00..000..000...0,0,0..0..0...0,0000...0..0000,0...11,111...011,111,110]
19683     7730;
19684     ZTARGET7730:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7730)  DCS[0,00,0,0,0,0]  BM[0000..00,00..00,00..000..000...0,0,0..0..0...0,0000...0..0000,0...11,111...011,111,110]
19685     7731;
19686     ZTARGET7731:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7731)  DCS[0,00,0,0,0,0]  BM[0000..00,00..00,00..000..000...0,0,0..0..0...0,0000...0..0000,0...11,111...011,111,110]
19687     7732;
19688     ZTARGET7732:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7732)  DCS[0,00,0,0,0,0]  BM[0000..00,00..00,00..000..000...0,0,0..0..0...0,0000...0..0000,0...11,111...011,111,110]
19689     7733;
19690     ZTARGET7733:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7733)  DCS[0,00,0,0,0,0]  BM[0000..00,00..00,00..000..000...0,0,0..0..0...0,0000...0..0000,0...11,111...011,111,110]
19691     7734;
19692     ZTARGET7734:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7734)  DCS[0,00,0,0,0,0]  BM[0000..00,00..00,00..000..000...0,0,0..0..0...0,0000...0..0000,0...11,111...011,111,110]
19693     7735;
19694     ZTARGET7735:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7735)  DCS[0,00,0,0,0,0]  BM[0000..00,00..00,00..000..000...0,0,0..0..0...0,0000...0..0000,0...11,111...011,111,110]
19695     7736;
19696     ZTARGET7736:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7736)  DCS[0,00,0,0,0,0]  BM[0000..00,00..00,00..000..000...0,0,0..0..0...0,0000...0..0000,0...11,111...011,111,110]
19697     7737;
19698     ZTARGET7737:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7737)  DCS[0,00,0,0,0,0]  BM[0000..00,00..00,00..000..000...0,0,0..0..0...0,0000...0..0000,0...11,111...011,111,110]
19699     7740;
19700     ZTARGET7740:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
```



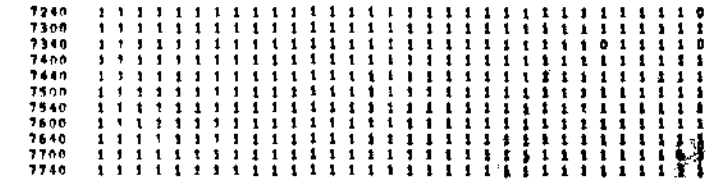
```
      (7740)  DCS[0,00,0,0,0,0]  BM[0000..00,00..00,00..000..000...0,0,0..0..0...0,0000...0..0000,0...11,111...011,111,110]
19701     7741;
19702     ZTARGET7741:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7741)  DCS[0,00,0,0,0,0]  BM[0000..00,00..00,00..000..000...0,0,0..0..0...0,0000...0..0000,0...11,111...011,111,110]
19703     7742;
19704     ZTARGET7742:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7742)  DCS[0,00,0,0,0,0]  BM[0000..00,00..00,00..000..000...0,0,0..0..0...0,0000...0..0000,0...11,111...011,111,110]
19705     7743;
19706     ZTARGET7743:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7743)  DCS[0,00,0,0,0,0]  BM[0000..00,00..00,00..000..000...0,0,0..0..0...0,0000...0..0000,0...11,111...011,111,110]
19707     7744;
19708     ZTARGET7744:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7744)  DCS[0,00,0,0,0,0]  BM[0000..00,00..00,00..000..000...0,0,0..0..0...0,0000...0..0000,0...11,111...011,111,110]
19709     7745;
19710     ZTARGET7745:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7745)  DCS[0,00,0,0,0,0]  BM[0000..00,00..00,00..000..000...0,0,0..0..0...0,0000...0..0000,0...11,111...011,111,110]
19711     7746;
19712     ZTARGET7746:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7746)  DCS[0,00,0,0,0,0]  BM[0000..00,00..00,00..000..000...0,0,0..0..0...0,0000...0..0000,0...11,111...011,111,110]
19713     7747;
19714     ZTARGET7747:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7747)  DCS[0,00,0,0,0,0]  BM[0000..00,00..00,00..000..000...0,0,0..0..0...0,0000...0..0000,0...11,111...011,111,110]
19715     7750;
19716     ZTARGET7750:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7750)  DCS[0,00,0,0,0,0]  BM[0000..00,00..00,00..000..000...0,0,0..0..0...0,0000...0..0000,0...11,111...011,111,110]
19717     7751;
19718     ZTARGET7751:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7751)  DCS[0,00,0,0,0,0]  BM[0000..00,00..00,00..000..000...0,0,0..0..0...0,0000...0..0000,0...11,111...011,111,110]
19719     7752;
19720     ZTARGET7752:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7752)  DCS[0,00,0,0,0,0]  BM[0000..00,00..00,00..000..000...0,0,0..0..0...0,0000...0..0000,0...11,111...011,111,110]
19721     7753;
19722     ZTARGET7753:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7753)  DCS[0,00,0,0,0,0]  BM[0000..00,00..00,00..000..000...0,0,0..0..0...0,0000...0..0000,0...11,111...011,111,110]
19723     7754;
19724     ZTARGET7754:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7754)  DCS[0,00,0,0,0,0]  BM[0000..00,00..00,00..000..000...0,0,0..0..0...0,0000...0..0000,0...11,111...011,111,110]
19725     7755;
19726     ZTARGET7755:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7755)  DCS[0,00,0,0,0,0]  BM[0000..00,00..00,00..000..000...0,0,0..0..0...0,0000...0..0000,0...11,111...011,111,110]
19727     7756;
19728     ZTARGET7756:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7756)  DCS[0,00,0,0,0,0]  BM[0000..00,00..00,00..000..000...0,0,0..0..0...0,0000...0..0000,0...11,111...011,111,110]
19729     7757;
19730     ZTARGET7757:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7757)  DCS[0,00,0,0,0,0]  BM[0000..00,00..00,00..000..000...0,0,0..0..0...0,0000...0..0000,0...11,111...011,111,110]
19731     7760;
19732     ZTARGET7760:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7760)  DCS[0,00,0,0,0,0]  BM[0000..00,00..00,00..000..000...0,0,0..0..0...0,0000...0..0000,0...11,111...011,111,110]
19733     7761;
19734     ZTARGET7761:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
      (7761)  DCS[0,00,0,0,0,0]  BM[0000..00,00..00,00..000..000...0,0,0..0..0...0,0000...0..0000,0...11,111...011,111,110]
19735     7762;
19736     ZTARGET7762:    NEXT,   BUTA(RETURN), J/BUTERROR?       ;COMPARE MICROADDRESS, THEN RETURN
```

```
      (7762)  DC8[0,00,0,0,0,0]  BM[0000..00,00..00,00,,000,,000,..0,0,0,,0,,0...0.0000...0..0000,0...11,111...011,111,110]
19737    7763;
19738    ZTARGET7763:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7763)  DC8[0,00,0,0,0,0]  BM[0000..00,00..00,00,,000,,000,..0,0,0,,0,,0...0.0000...0..0000,0...11,111...011,111,110]
19739    7764;
19740    ZTARGET7764:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7764)  DC8[0,00,0,0,0,0]  BM[0000..00,00..00,00,,000,,000,..0,0,0,,0,,0...0.0000...0..0000,0...11,111...011,111,110]
19741    7765;
19742    ZTARGET7765:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7765)  DC8[0,00,0,0,0,0]  BM[0000..00,00..00,00,,000,,000,..0,0,0,,0,,0...0.0000...0..0000,0...11,111...011,111,110]
19743    7766;
19744    ZTARGET7766:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7766)  DC8[0,00,0,0,0,0]  BM[0000..00,00..00,00,,000,,000,..0,0,0,,0,,0...0.0000...0..0000,0...11,111...011,111,110]
19745    7767;
19746    ZTARGET7767:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7767)  DC8[0,00,0,0,0,0]  BM[0000..00,00..00,00,,000,,000,..0,0,0,,0,,0...0.0000...0..0000,0...11,111...011,111,110]
19747    7770;
19748    ZTARGET7770:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7770)  DC8[0,00,0,0,0,0]  BM[0000..00,00..00,00,,000,,000,..0,0,0,,0,,0...0.0000...0..0000,0...11,111...011,111,110]
19749    7771;
19750    ZTARGET7771:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7771)  DC8[0,00,0,0,0,0]  BM[0000..00,00..00,00,,000,,000,..0,0,0,,0,,0...0.0000...0..0000,0...11,111...011,111,110]
19751    7772;
19752    ZTARGET7772:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7772)  DC8[0,00,0,0,0,0]  BM[0000..00,00..00,00,,000,,000,..0,0,0,,0,,0...0.0000...0..0000,0...11,111...011,111,110]
19753    7773;
19754    ZTARGET7773:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7773)  DC8[0,00,0,0,0,0]  BM[0000..00,00..00,00,,000,,000,..0,0,0,,0,,0...0.0000...0..0000,0...11,111...011,111,110]
19755    7774;
19756    ZTARGET7774:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7774)  DC8[0,00,0,0,0,0]  BM[0000..00,00..00,00,,000,,000,..0,0,0,,0,,0...0.0000...0..0000,0...11,111...011,111,110]
19757    7775;
19758    ZTARGET7775:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7775)  DC8[0,00,0,0,0,0]  BM[0000..00,00..00,00,,000,,000,..0,0,0,,0,,0...0.0000...0..0000,0...11,111...011,111,110]
19759    7776;
19760    ZTARGET7776:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7776)  DC8[0,00,0,0,0,0]  BM[0000..00,00..00,00,,000,,000,..0,0,0,,0,,0...0.0000...0..0000,0...11,111...011,111,110]
19761    7777;
19762    ZTARGET7777:    NEXT,    BUTA(RETURN), J/BUTERROR?        ;COMPARE MICROADDRESS, THEN RETURN
      (7777)  DC8[0,00,0,0,0,0]  BM[0000..00,00..00,00,,000,,000,..0,0,0,,0,,0...0.0000...0..0000,0...11,111...011,111,110]
19763
19764
19765    |*****************************************************************************************
19766    .TOC    * END OF KD11-K MICRODIAGNOSTIC CODE
19767    |*****************************************************************************************
19768
19769    .END
```



```
4000   1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
4040   1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
4100   1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
4140   1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
4200   1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
4240   1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
4300   1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
4340   1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
4400   1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
4440   1 0 0 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
4500   1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
4540   1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
4600   1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
4640   1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
4700   1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
4740   1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
5000   1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
5040   1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
5100   1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
5140   1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
5200   1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
5240   1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
5300   1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
5340   1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
5400   1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
5440   1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 1 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1
5500   1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
5540   1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
5600   1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
5640   1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
5700   1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
5740   1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
6000   1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
6040   1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
6100   1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
6140   1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
6200   1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
6240   1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
6300   1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
6340   1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
6400   1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
6440   1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
6500   1 1 1 1 1 1 1 0 1 1 1 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
6540   1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
6600   1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

6640   1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
6700   1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
6740   1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
7000   1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
7040   1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
7100   1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
7140   1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
7200   1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
```



LINE	LOCN	SYMBOL	LINE	LOCN	SYMBOL	LINE	LOCN	SYMBOL	LINE	LOCN	SYMBOL
8399	7025	AGAINBRD350	14400	7189	ALI-537A	18019	6731	ALUGW763A	5887	4003	ALU114A
5946	4610	ALU115A1	6028	4015	ALU115D1	6141	4622	ALU115C1	6171	4026	ALU115D1
6242	4634	ALU116A1	6351	4040	ALU116B	6366	4006	ALU116C1	6489	4050	ALU116D
6507	4666	ALU117A1	6577	4057	ALU117B1	6660	4663	ALU117C1	6697	4664	ALU120A1
6752	4072	ALU120B1	6839	4706	ALU121B1	6913	4107	ALU121B1	6994	4670	ALU121C1
7064	4121	ALH121D1	7155	4672	ALU122A1	8041	7007	ALUCARRY1	8050	7017	ALUCARRY1A
8058	7020	ALUCARRY2	8067	7021	ALUCARRYBA	14347	7145	AR11-537A	14298	6451	AR3-537A
14323	7141	AR7-537A	7290	4640	ARITH130A1	7341	4140	ARITH130B1	7399	4520	ARITH131A1
7450	4150	ARITH131B1	7509	4522	ARITH132A1	7554	4107	ARITH132B1	10856	6114	ASIDE410C
11048	6525	AASERTFOV500	15612	5413	BC1FCN622A	15369	5670	BCERC620A	15641	5416	BCERC622B
17504	7204	BDX05	17492	7177	BDX12	14407	4142	BL1-537A	14353	7146	BR11-537A
14305	7134	BR3-537A	14329	7142	BR7-537A	17054	4251	BUMPD762A	18180	4313	BUSFCN710A
16342	4311	BUSFCN711A	16480	4316	BUSFCN713A	16600	7147	BUSFCN713A	16780	7176	BUSFCN720A
16863	7201	BUSFCN721A	16474	4324	BUSFCN722A	17301	4131	BUSFCN730A	17161	4337	BUSFCN730B
17257	4346	BUSFCN730D	17328	4764	BUSFCN731A	17350	4366	BUSFCN731B	17440	4364	BUSFCN731D
17563	4371	BUSFCN740A	17598	4373	BUSFCN740B	17657	4374	BUSFCN740C	17660	4401	BUSFCN740D
19178	7351	BUTBA00	18972	7307	BUTBGPPSERV	19032	7316	BUTBGSSERVL	11873	6217	BUTCLR505A
14517	7136	BUTCOUNT-IS-377	18985	7311	BUTCOUTTDOUT7	19118	7141	BUTD-IS-ZERO	18998	7313	BUTDNSMBYTE
19160	7347	BUTDR6-7L	19139	7344	BUTD[C]A	19173	7360	BUTD[C]B	18977	7310	BUTD[C]C
2880	4376	BUTERRDR4	2886	5376	BUTERROR6	2892	6376	BUTERROR6	2898	7376	BUTERROR7
19222	7371	BUTFPFD	19191	7354	BUTFPPRDC	18990	7312	BUTFPSDS	19006	7314	BUTGD3-2
18964	7306	BUTINSTP1	18932	7301	BUTINSTPS	19209	7356	BUTINSTPRAWCH	19204	7355	BUTINTMHIGH
19126	7342	BUTIR11B	18940	7302	BUTIR11FLTPT3-0	18924	7300	BUTIR15-12	18948	7304	BUTIR9-6
19038	7321	BUTMD00	19071	7327	BUTNEXFLAG1	19093	7333	BUTNEXFLAG2	19060	7325	BUTNFLAG7
19082	7331	BUTNFLTPT8	19105	7336	BUTNINITJAM	19087	7317	BUTNMASKPS[T]	19032	7320	BUTNNXT000
19043	7322	BUTNNXT001	19054	7324	BUTNNXT002	19085	7326	BUTNNXT003	19076	7330	BUTNNXT004
19087	7332	BUTNNXT005	19098	7334	BUTNNXT006	19110	7340	BUTNNXT007	18956	7305	BUTNOVDR7IRS-3
19049	7323	BUTNPS[N]	19186	7352	BUTOTHERJAM	19217	7360	BUTPREFETCHJAM	19191	7343	BUTPS15
19147	7345	BUTSERVICE	19014	7315	BUTSR1-0	18916	7276	BUTSR3-0	19155	7346	BUTVECTLOAD
10540	7062	BYTE375A	10745	7070	BYTEFIRST410	10764	7367	BYTESECOND410	17509	7205	C17X05
17497	7203	C17X12	10547	7063	CHECK379A	10570	6300	CHECK375B	12293	6254	CINSOTF
18600	7247	CJESERVICETOD	18454	7227	CLEAR-I-O-A	18459	7230	CLEAR-I-O-B	12106	6236	CLEAR506E
15878	7165	CLEAR624	16278	6470	CLEAR710D	17747	4604	CLEAR761A	18464	7231	CLEARIO02
18478	7232	CLEARIO04	18474	7233	CLEARIO05	18638	7255	CLRJAMTOD	18585	7244	CLRSERVICETOD
18674	7262	CMPPROCDAT	15618	5414	CNSTC622A	7299	4135	COMP130A1	7380	4141	COMP130B1
7408	4144	COMP131A1	7459	4151	COMP131B1	8434	4702	COMP350A	8451	4244	COMP350B
8479	4246	COMP350C	8506	4250	COMP350D	8576	4704	COMP351A	8598	4257	COMP351B
8620	4261	COMP351C	8641	4263	COMP351D	8677	4714	COMP352A	8699	4267	COMP352B
8721	4271	COMP352C	8742	4273	COMP352D	8860	6004	COMP361B	8943	6012	COMP362A
8976	6015	COMP362B	9051	6022	COMP363A	9174	6034	COMP365A	9242	6042	COMP366A
9302	6047	COMP366C	9386	6057	COMP370A	9447	6064	COMP370C	9517	6072	COMP371A
10797	6106	COMP410A	10827	6111	COMP410B	10861	6115	COMP410C	10893	6120	COMP410D
10925	6123	COMP410E	11078	6134	COMP500A	11114	6137	COMP500C	11143	6141	COMP500D
11179	6144	COMP500E	13548	6350	COMP533A	13589	6354	COMP533B	13630	6357	COMP534A
13664	6362	COMP534B	13704	6365	COMP534C	13738	6370	COMP534D	13785	6374	COMP534E
13824	6402	COMP534F	13986	6411	COMP535A	13966	6415	COMP535B	11998	6422	COMP536A
14039	6426	COMP536B	14082	6431	COMP536C	14129	6436	COMP536D	14175	6442	COMP536E
14215	6446	COMP536F	14422	4302	COMP537A	15580	5412	COMP621H	17116	4333	COMP730A
17342	4354	COMP731A	17409	4361	COMP731C	14579	4301	COUNTER01	14584	4310	COUNTER02
14592	4526	COUNTER03	14603	4545	COUNTER04	14612	4557	COUNTER05	14621	4534	COUNTER06
14631	4535	COUNTER07	14641	4536	COUNTER08	14652	4529	COUNTER09	14663	4527	COUNTER10
14670	4533	COUNTER10A	14676	4531	COUNTER10B	14683	4547	COUNTER11	14690	4555	COUNTER12

LINE	LOCN	SYMBOL	LINE	LOCN	SYMBOL	LINE	LOCN	SYMBOL	LINE	LOCN	SYMBOL
13851	6404	COUNTERS14G	14359	7147	CR11-537A	14311	7137	CR3-537A	14335	7143	CR7-537A
9574	7035	CSP16XOFFLTTOIRS	9568	7031	CSP16XORSRTOIRS	15352	5372	CSP1L621A	9627	6222	CUA372A
9678	5555	CUA372B	18650	7257	CUATOD	18843	7241	DOS00A	18529	7237	D1106A
18518	7235	D1S12A	18590	7245	DATISERVICETOD	18596	7246	DATOSERVICETOD	18758	7270	DBUFINTOIR
18788	7274	DBUFINTOIRS	17958	4415	DELAY762C	18783	7267	DINTOIR	8389	7363	DINTOIR350
18782	7273	DINTOIRS	8408	4563	DINTOIRS350	18410	7222	DISP002	18416	7223	DISP003
18422	7224	DISP004	18428	7225	DISP005	18434	7226	DISP006	18404	7221	DISPLAY
10718	6531	DNONZERO410	11282	6153	DOFCC503A	11611	6177	DOFCC504A	10608	5440	DOIT376A
12009	7072	DOIT506A	14785	5313	DOIT610A1	14838	5331	DOIT610A2	14898	5327	DOIT610B1
14953	5336	DOIT610B2	15016	5344	DOIT610C1	15067	5352	DOIT610C2	15124	5357	DOIT610D1
15163	5363	DOIT610D2	7613	4163	DOPA133A1	7725	4173	DOPA134A1	7838	4203	DOPA135A1
7944	4212	DOPA136A1	9926	5225	DOWRITE374A1	9946	5230	DOWRITE374A2	9999	5234	DOWRITE374B1
10019	5237	DOWRITE374B2	10072	5243	DOWRITE374C1	10092	5247	DOWRITE374C2	10145	5253	DOWRITE374D1
10165	5256	DOWRITE374D2	10218	5262	DOWRITE374E1	10238	5265	DOWRITE374E2	10291	5271	DOWRITE374F1
10311	5274	DOWRITE374F2	14365	7150	DR11-537A	14327	7140	DR3-537A	14341	7144	DR7-537A
18558	7243	DTOIRB	17836	4411	DW11L762A	18552	7242	DZERO	10728	6533	DZERO410
18538	7240	D(05-00)	18524	7236	D(11-06)	18611	7234	D(15-12)	12482	6270	ENFLAG510D
18230	4431	EOP001	18237	6452	EOP002	18263	6610	EOP003	18250	6611	EOP004
18256	7211	EOP005	18265	6776	EOP006	18278	6612	EOP007	18282	7212	EOP010
14371	7151	ER3-537A	2827	6543	ERROR010	15426	5615	ERROR621A	15803	6567	ERROR624A
8853	6003	EXPEC361B	8969	6014	EXPEC362B	9187	6033	EXPEC365A	9236	6041	EXPEC366A
9295	6046	EXPEC366C	9914	5222	EXPEC374A1	9997	5537	EXPEC374B1	10060	5557	EXPEC374C1
10133	5567	EXPEC374D1	10206	5515	EXPEC374E1	10270	5527	EXPEC374F1	10785	7363	EXPEC410A
10877	6110	EXPEC410B	10851	6113	EXPEC410C	10986	6117	EXPEC410D	10918	6122	EXPEC410E
11108	6136	EXPEC500C	11171	6143	EXPEC500E	11262	6150	EXPEC503A	11414	6163	EXPEC503E
11599	6175	EXPEC504A	11723	6206	EXPEC504E	11560	6205	EXPEC505A	11989	6612	EXPEC506A
12151	6602	EXPEC507A	13893	6407	EXPEC535A	13934	6403	EXPEC535B	13988	6420	EXPEC536A
14026	6424	EXPEC536B	14116	6434	EXPEC536D	15802	5408	EXPEC621E	15528	5407	EXPEC621F
15688	5421	EXPEC623	15846	5430	EXPEC624C	15939	7166	EXPEC701A	15989	6455	EXPEC701C
16052	5437	EXPEC702A	16096	6460	EXPEC702B	16243	6465	EXPEC710C	16272	6467	EXPEC710D
16381	6473	EXPEC711B	16411	6475	EXPEC711C	16562	6501	EXPEC712D	16648	5452	EXPEC713B
16697	6504	EXPEC713D	17156	4336	EXPEC730B	18013	4417	EXPEC762E	18069	4422	EXPEC763A
17953	4414	FILL762C	10526	7046	FIRST376A	11056	6027	FIXPAT500	12448	6267	FLAG510D
12571	7075	FLAGFP802	12577	7076	FLAGFP803	12582	7077	FLAGFP804	12566	7073	FLAGFPSSEQLOD
18619	7252	FLAGFP8TOD	14382	7152	FR3-537A	11992	6231	FUDGEPB506A	12177	6244	FUDGEP8507A
15474	4314	GENADR713A	5954	4010	GETALT[illegible]115A1	15535	5410	GETCUA621F	15695	5422	GETCUA623
15857	5431	GETCUA624C	17109	4332	GETDBUF730A	17335	4353	GETDBUF731A	8110	4222	GETDC320A
8176	4227	GETDC320C	8237	4234	GETDC320E	14784	5314	GETIT610A1	14844	5322	GETIT610A2
14907	5330	GETIT610B1	14962	5337	GETIT610B2	15025	5345	GETIT610C1	15076	5353	GETIT610C2
15133	5360	GETIT610D1	15172	5364	GETIT610D2	17169	4340	GETIT730B	17265	4347	GETIT730D
17376	4357	GETIT731B	17853	7210	GETIT762A	15324	5371	GETJAM620C	15508	5406	GETJAM621E
15665	5420	GETJAM622C	18662	7260	GETNOKPROCDAT	5801	4600	GETONES114A	18657	7373	GETPROCDAT
9934	5226	GETTEM374A1	9954	5231	GETTEM374A2	10007	5235	GETTEM374B1	10027	5240	GETTEM374B2
10080	5244	GETTEM374C1	10100	5250	GETTEM374C2	10153	5254	GETTEM374D1	10173	5257	GETTEM374D2
10226	5263	GETTEM374E1	10246	5266	GETTEM374E2	10299	5272	GETTEM374F1	10319	5275	GETTEM374F2
5894	4004	GETZEROES114A	3056	5003	GOBUT012A	3077	5004	GOBUT012B	3097	5005	GOBUT012C
3116	5006	GOBUT012D	3135	5007	GOBUT012E	3156	5010	GOBUT012F	3177	5011	GOBUT012G
3220	5013	GOBUT013A	3242	5014	GOBUT013B	3261	5015	GOBUT013C	3280	5016	GOBUT013D
3299	5017	GOBUT013E	3320	5020	GOBUT013F	3339	5021	GOBUT013G	3382	5023	GOBUT014A
3403	5024	GOBUT014B	3422	5025	GOBUT014C	3444	5026	GOBUT014D	3488	5030	GOBUT015A
3508	5031	GOBUT015B	3527	5032	GOBUT016C	3570	5034	GOBUT016A	3613	5036	GOBUT017A
3635	5037	GOBUT017B	3677	5041	GOBUT020A	3697	5042	GOBUT020B	3719	5043	GOBUT020C

LINE	LOCN	SYMBOL	LINE	LOCN	SYMBOL	LINE	LOCN	SYMBOL	LINE	LOCN	SYMBOL
3762	5045	GOBUT021A	3783	5046	GOBUT021B	3808	5047	GOBUT021C	3848	5051	GOBUT022A
3892	5053	GOBUT023A	3936	5055	GOBUT024A	3987	5056	GOBUT024B	4001	5060	GOBUT025A
4045	5062	GOBUT026A	4089	5064	GOBUT027A	4131	5066	GOBUT030A	4155	5067	GOBUT030B
4174	5070	GOBUT030C	4197	5071	GOBUT030D	4240	5073	GOBUT031A	4261	5074	GOBUT031B
4304	5076	GOBUT032A	4325	5077	GOBUT032B	4370	5101	GOBUT033A	4414	5103	GOBUT034A
4457	5105	GOBUT035A	4502	5107	GOBUT036A	4546	5111	GOBUT037A	4589	5113	GOBUT040A
4611	5114	GOBUT040B	4654	5116	GOBUT041A	4676	5117	GOBUT041B	4718	5121	GOBUT042A
4739	5122	GOBUT042B	4783	5124	GOBUT043A	4804	5125	GOBUT043B	4848	5127	GOBUT044A
4893	5131	GOBUT045A	4935	5133	GOBUT046A	4989	5134	GOBUT046B	5004	5136	GOBUT047A
5027	5137	GOBUT047B	5047	5140	GOBUT047C	5093	5141	GOBUT050A	5122	5144	GOBUT050B
5626	5210	GOBUT105A1	5674	5213	GOBUT105B1	5899	4005	GOBUT114A	5913	4006	GOBUT114A2
5959	4011	GOBUT115A1	5991	4013	GOBUT115A2	6000	4014	GOBUT115A3	5975	4012	GOBUT115A4
6036	4016	GOBUT115B1	6068	4020	GOBUT115B2	6085	4021	GOBUT115B3	6052	4017	GOBUT115B4
6119	4023	GOBUT115C1	6134	4024	GOBUT115C2	6151	4025	GOBUT115C3	6179	4027	GOBUT115D1
6194	4030	GOBUT115D2	6211	4031	GOBUT115D3	6250	4033	GOBUT116A1	6264	4034	GOBUT116A2
6279	4035	GOBUT116A3	6296	4036	GOBUT116A4	6312	4037	GOBUT116A5	6340	4041	GOBUT116B
6374	4043	GOBUT116C1	6389	4044	GOBUT116C2	6404	4042	GOBUT116C3	6420	4046	GOBUT116C4
6436	4047	GOBUT116C5	6466	4051	GOBUT116D	6510	4053	GOBUT117A1	6535	4054	GOBUT117A2
6542	4055	GOBUT117A3	6557	4056	GOBUT117A4	6585	4060	GOBUT117B1	6600	4061	GOBUT117B2
6617	4062	GOBUT117B3	6647	4064	GOBUT117C1	6663	4065	GOBUT117C2	6700	4067	GOBUT120A1
6715	4070	GOBUT120A2	6732	4071	GOBUT120A3	6760	4073	GOBUT120B1	6775	4074	GOBUT120B2
6792	4075	GOBUT120B3	6807	4076	GOBUT120B4	6847	4103	GOBUT121A1	6862	4104	GOBUT121A2
6878	4105	GOBUT121A3	6893	4106	GOBUT121A4	6921	4110	GOBUT121B1	6936	4111	GOBUT121B2
6953	4112	GOBUT121B3	6968	4113	GOBUT121B4	7002	4115	GOBUT121C1	7017	4116	GOBUT121C2
7031	4117	GOBUT121C3	7048	4120	GOBUT121C4	7076	4122	GOBUT121D1	7091	4123	GOBUT121D2
7108	4124	GOBUT121D3	7123	4125	GOBUT121D4	7163	4127	GOBUT122A1	7179	4130	GOBUT122A2
7195	4131	GOBUT122A3	7215	4133	GOBUT122A4	7306	4136	GOBUT130A1	7321	4137	GOBUT130A2
7357	4142	GOBUT130B1	7372	4143	GOBUT130B2	7418	4146	GOBUT131A1	7430	4147	GOBUT131A2
7466	4152	GOBUT131B1	7481	4153	GOBUT131B2	7518	4155	GOBUT132A1	7534	4156	GOBUT132A2
7563	4160	GOBUT132B1	7579	4161	GOBUT132B2	7630	4165	GOBUT133A2	7605	4170	GOBUT133B2
7750	4175	GOBUT134A2	7798	4200	GOBUT134B2	7863	4205	GOBUT135A2	7911	4210	GOBUT135B2
7969	4214	GOBUT136A2	8017	4217	GOBUT136B2	8125	4223	GOBUT320A	8146	4224	GOBUT320B
8184	4230	GOBUT320C	8204	4231	GOBUT320D	8244	4235	GOBUT320E	8263	4236	GOBUT320F
8429	4243	GOBUT350A	8456	4245	GOBUT350B	8484	4247	GOBUT350C	8512	4251	GOBUT350D
8581	4256	GOBUT351A	8603	4260	GOBUT351B	8628	4262	GOBUT351C	8647	4264	GOBUT351D
8682	4266	GOBUT352A	8704	4270	GOBUT352B	8726	4272	GOBUT352C	8748	4274	GOBUT352D
8880	6005	GOBUT361D	8902	6006	GOBUT361E	8949	6013	GOBUT362A	8996	6016	GOBUT362D
9017	6017	GOBUT362E	9056	6023	GOBUT363A	9076	6024	GOBUT363B	9104	6026	GOBUT364A
9131	6030	GOBUT364B	9194	6035	GOBUT365B	9263	6043	GOBUT366B	9345	6053	GOBUT367B
9391	6060	GOBUT370A	9412	6061	GOBUT370B	9452	6065	GOBUT370C	9474	6066	GOBUT370D
9522	6073	GOBUT371A	9545	6074	GOBUT371B	9821	6101	GOBUT373B	10552	7064	GOBUT375A
10575	5301	GOBUT375B	10626	5303	GOBUT376A	10964	6116	GOBUT410C	10898	6121	GOBUT410D
10930	6124	GOBUT410E	11084	6135	GOBUT500A	11121	6140	GOBUT500C	11149	6142	GOBUT500D
11184	6145	GOBUT500E	11306	6155	GOBUT503A	11325	6156	GOBUT503B	11346	6157	GOBUT503C
11393	6162	GOBUT503D	11366	6160	GOBUT503DA	11440	6165	GOBUT503F	11481	6166	GOBUT503G
11482	6167	GOBUT503H	11502	6170	GOBUT503I	11520	6171	GOBUT503J	11539	6172	GOBUT503K
11636	6201	GOBUT504A	11654	6202	GOBUT504B	11675	6203	GOBUT504C	11702	6205	GOBUT504D
11749	6210	GOBUT504F	11770	6211	GOBUT504G	11791	6212	GOBUT504H	11811	6213	GOBUT504I
11899	6221	GOBUT505B	11919	6223	GOBUT505C	12022	7071	GOBUT506A	12041	6233	GOBUT506B
12062	6234	GOBUT506C	12083	6235	GOBUT506D	12112	6237	GOBUT506E	12189	6246	GOBUT507A
12208	6247	GOBUT507B	12229	6250	GOBUT507C	12250	6251	GOBUT507D	12273	6253	GOBUT507E
12328	6261	GOBUT507F	12381	6264	GOBUT510A	12403	6265	GOBUT510B	12425	6266	GOBUT510C

LINE	LOCN	SYMBOL
12462	6273	GOBUT510D
13555	6391	GOBUT513A
13711	6368	GOBUT514C
13856	6405	GOBUT534G
14044	6427	GOBUT536B
14222	6447	GOBUT536F
15447	5402	GOBUT6218
15586	7160	GOBUT621H
15969	5434	GOPUT701B
17127	4335	GOBUT730A
17270	4350	GOBUT730D
17414	4362	GOBUT731C
17605	4766	GOBUT740B
17771	4606	GOBUT761B
18038	4421	GOBUT762F
10773	7361	GOFOR410
11471	6164	GOPUT503E
5173	5147	GOTEST101A
5342	5166	GOTEST102D
5488	5201	GOTEST103D
5654	5212	GOTEST105B
7619	4164	GOTEST113A1
7844	4204	GOTEST115A1
9967	5232	GOTEST374A2
10259	5267	GOTEST374E2
12783	7115	GOTEST511A3
12839	7137	GOTEST511B3
13100	6315	GOTEST512B1
13204	6325	GOTEST512D1
13346	6335	GOTEST520A
13460	6348	GOTEST520E
14552	5626	GOTEST551C
16113	6463	GOTEST7028
16301	6472	GOTEST710E
16519	5450	GOTEST712E
16796	6506	GOTEST720B
17012	4325	GOTEST722B
13535	6666	INIT533A
14110	6433	INIT5360
13458	6361	INITD534B
13817	6401	INITD534F
14033	6425	INITD536B
14209	6445	INITD536F
2948	7004	INITIALIZE05
2977	4102	INITIALIZE11
18378	7217	INSERT03
10802	6187	INTOIP410A
18835	4434	JAMUPP002C
18861	4747	JAMUPP005
18887	4460	JAMUPP011
12944	7124	KTSRCDST03
12973	7137	KTSRCDST06

LINE	LOCN	SYMBOL
12483	6273	GOBUT510DA
13595	6395	GOBUT533B
13745	6371	GOBUT534D
13912	6412	GOBUT535A
14089	6432	GOBUT536C
14428	4303	GOBUT737A
15465	5403	GOBUT621C
15623	5415	GOBUT622A
16018	8435	GOBUT701D
17175	4341	GOBUT730B
17293	4351	GOBUT736E
17452	4366	GOBUT731D
17633	4375	GOBUT740C
17793	4407	GOBUT761C
18115	4425	GOBUT763B
18019	4420	GOGET761E
11630	6200	GOPUT504A
5263	5156	GOTEST102A
5469	5173	GOTEST103A
5528	5203	GOTEST104A
5762	5215	GOTEST105C
7669	4167	GOTEST133B1
7894	4207	GOTEST135B1
10040	5241	GOTEST374B2
10332	5276	GOTEST374F2
12794	7113	GOTEST511A4
12850	7121	GOTEST511B4
13115	6316	GOTEST512B2
13219	6326	GOTEST512D2
13376	6337	GOTEST520B
14487	4564	GOTEST581A
15949	7171	GOTEST701A
16219	6464	GOTEST710B
16387	6474	GOTEST711B
16567	6502	GOTEST712D
16820	4317	GOTEST720C
17036	4326	GOTEST722C
13576	6352	INIT535B
13542	6347	INITD533A
13698	6662	INITD534C
13899	6410	INITD535A
14075	6852	INITD536C
2931	6532	INITIALIZE01
2953	7005	INITIALIZE06
2988	4001	INITIALIZE12
18384	7220	INSERT04
18643	7256	JANTOD
18841	7213	JAMUPP002D
18867	4435	JAMUPP006
18176	4763	KILL764A
12951	7125	KTSRCDST04
12981	7127	KTSRCDST07

LINE	LOCN	SYMBOL
12804	6276	GOBUT510E
13616	6360	GOBUT534A
13791	6375	GOBUT534E
13982	6416	GOBUT535B
14135	6437	GOBUT536D
15275	5366	GOBUT620A
15463	5404	GOBUT621D
15649	9417	GOBUT622B
16337	6500	GOBUT712C
17198	4342	GOBUT730C
17347	4356	GOBUT731A
17454	4367	GOBUT731E
17607	4767	GOBUT740D
17932	4762	GOBUT762B
18136	4426	GOBUT763C
10032	6112	GOPUT410B
11730	6207	GOPUT504E
5289	5160	GOTEST102B
5438	5175	GOTEST103B
5595	9205	GOTEST104B
5729	5217	GOTEST105D
7731	4174	GOTEST134A1
7980	4213	GOTEST136A1
10113	5251	GOTEST374C2
12761	6550	GOTEST511A1
12817	6984	GOTEST511B1
13048	6311	GOTEST512A1
13140	6321	GOTEST512C1
13285	6331	GOTEST512E1
13403	6341	GOTEST520C
14511	4560	GOTEST551B
15998	6497	GOTEST701C
16248	6466	GOTEST710C
16416	6476	GOTEST711C
16593	6483	GOTEST713B
16901	4321	GOTEST721B
14308	7183	GR3=537A
13987	6686	INIT535A
13902	6353	INITD533B
13732	6367	INITD534D
13960	6414	INITD535B
14122	6418	INITD536D
2977	7001	INITIALIZE03
2958	7006	INITIALIZE07
14180	7131	INITSR537A
18762	7214	INSERTEOPREVNO
10822	4377	JAMUPP001
18857	4786	JAMUPP003
18879	6436	JAMUPP007
12933	7114	KTSRCDST01
12959	7126	KTSRCDST04B
12987	7130	KTSRCDST08

LINE	LOCN	SYMBOL
12838	6277	GOBUT510F
13671	6363	GOBUT534B
13831	6403	GOBUT534F
14005	6423	GOBUT536A
14182	6443	GOBUT536E
15293	5367	GOBUT620B
15557	5411	GOBUT621G
15871	5432	GOBUT624D
16671	6503	GOBUT713C
17218	4443	GOBUT730C1
17382	4360	GOBUT731B
17572	4776	GOBUT740A
17752	4405	GOBUT761A
17987	4416	GOBUT762D
18155	4427	GOBUT763D
11300	6154	GOPUT503A
11879	6220	GOPUT505A
5316	5162	GOTEST102C
5482	5177	GOTEST103C
5606	5207	GOTEST105A
5750	5220	GOTEST105E
7761	4177	GOTEST134B1
8000	4216	GOTEST136B1
10185	5260	GOTEST374D2
12772	7107	GOTEST511A2
12828	7115	GOTEST511B2
13060	6312	GOTEST512A2
13164	6322	GOTEST512C2
13268	6332	GOTEST512E2
13431	6343	GOTEST520D
14528	7157	GOTEST551B1
16074	5444	GOTEST702A
16264	6471	GOTEST710D
16435	6477	GOTEST711D
16702	6505	GOTEST713D
16925	4322	GOTEST721C
14394	7154	HR3=537A
13980	6654	INIT536A
13623	6664	INITD534A
13778	6373	INITD534E
13993	6421	INITD536A
14168	6441	INITD536E
2942	7003	INITIALIZE04
2965	4101	INITIALIZE10
18367	7215	INSERT02
18373	7216	INSERTREVNO
18830	4757	JAMUPP002B
18854	4746	JAMUPP004
18877	4437	JAMUPP010
12938	7123	KTSRCDST02
12967	7135	KTSRCDST05
5252	5154	LOAD00-102A

LINE	LOCN	SYMBOL
5232	5530	LOAD01-102A
14937	5333	LOAD01-610B2
2909	4100	LOAD010
3376	5742	LOAD014A
3604	5732	LOAD017A
3841	5716	LOAD022A
4039	5702	LOAD026A
4297	5666	LOAD032A
4450	5660	LOAD038A
4582	5650	LOAD040A
4841	5632	LOAD044A
8549	4567	LOAD05-351
5116	5143	LOAD050B
5393	5170	LOAD07-103A
14259	6276	LOAD14537A
5398	5171	LOAD17-103A
16170	4307	LOADADR710A
16107	6462	LOADBA702B
9311	5161	LOADD102C
5457	5176	LOADD103C
17096	4330	LOADDATA730A
11276	6152	LOADFCC503A
11255	6147	LOADFP8503A
8384	7357	LOADHI350
11186	6161	LOADIP503D
14926	5331	LOADIP610B2
17435	4363	LOADIR731D
12375	6263	LOADPRIO510A
9224	6037	LOADRE8366A
9907	6103	LOADRF8374
16602	5464	LOADRET713A
5600	5542	LOADSR105A
8844	6002	LOADSR361A
8350	4241	LOADSRD350
13772	6660	LODCNTRE34E
9613	6540	LOOP372A
15317	5370	MASK620C
12696	6232	MF8801
12715	6552	MF8805
18668	7261	MSKPROCDAT
16193	4445	NEXT710A
16789	7364	NEXT720A
8376	7023	NEXTPAT350
18633	7254	ODDJANTOD
7825	4514	OPA135A1
7719	4172	OPB134A1
18612	7251	PRATOD
14887	5325	P&CC-DC610B1
15113	5358	P&CC-DC610D1
15222	7164	P&CCTOSR3-08B
12170	6243	P&LO507A
12601	7101	P&SEQLOD02

LINE	LOCN	SYMBOL
14758	5306	LOAD01-610A1
15000	5342	LOAD01-610C1
2954	5001	LOAD011
14272	4300	LOAD01537A
5217	5151	LOAD02-102A
3885	5714	LOAD023A
4082	5700	LOAD027A
4363	5664	LOAD033A
4494	5654	LOAD036A
4646	5644	LOAD041A
4886	5630	LOAD045A
14764	5307	LOAD05-610A1
8554	4254	LOAD06-351
14932	5332	LOAD07-610B2
5383	5166	LOAD15-103A
9633	6703	LOAD372A
15944	7170	LOADBA701A
5189	5520	LOADD101A
5137	5163	LOADD102B
5483	5200	LOADD103D
17240	4343	LOADDATA730D
11605	6176	LOADFCC504A
11593	6174	LOADFP8504A
8393	7024	LOADHIA3D0
11695	6204	LOADIR504D
14994	5544	LOADIR610C1
17551	4765	LOADIR740A
8841	8001	LOADRE8361A
9332	6051	LOADRE8367A
16164	5472	LOADRET710A
16737	5462	LOADRET720A
5649	5211	LOADSR105B
9340	6052	LOADSR367A
11999	7067	LOADUCONS08A
13812	6400	LODCNTRE34F
17252	4345	MANCLED736D
15933	5476	MASK701A
12692	6302	MF8802
12725	6553	MF8806
14291	6643	NEMCTR537A
16355	4446	NEXT711A
16875	7362	NEXT721A
10737	7065	NEXTPAT410
7659	4166	OPA133B1
7884	4206	OPA135B1
7832	4202	OPB135A1
17847	7207	PRIO6762A
14942	5334	P&CC-DC610B2
15152	5361	P&CC-DC610D2
11977	6226	P&MI506A
11979	6227	P&MID506A
12607	7102	P&SEQLOD03

LINE	LOCN	SYMBOL
14819	5316	LOAD01-610A2
15001	5347	LOAD01-610C2
3049	5002	LOAD012A
3481	5740	LOAD015A
3670	5726	LOAD020A
3928	5706	LOAD024A
4188	5676	LOAD030A
4496	5662	LOAD034A
4629	5656	LOAD037A
4771	5640	LOAD042A
4930	5630	LOAD046A
15048	5346	LOAD05-610C2
14789	5310	LOAD06-610A1
5247	5153	LOAD10-102A
5377	5156	LOAD16-103A
9884	5902	LOAD372B
15993	6486	LOADBA701C
5358	5165	LOADD102A
5404	5172	LOADD103A
5523	5540	LOADD104A
12015	6104	LOADDCS506A
11246	6414	LOADFLAG503A
10704	7264	LOADFP8CC
14478	4305	LOADIN551A
14788	5532	LOADIR610A1
16743	7172	LOADIR720A
17986	4372	LOADIR740B
8937	6011	LOADRE8362A
9375	6035	LOADRE8370A
16335	8470	LOADRET711A
16847	4320	LOADRET721A
5697	5214	LOADSR105C
14162	6650	LODBRR536E
15349	5374	LODBRR621A
15988	6480	LODFLG621A
17446	4348	MANCLED731D
16047	6474	MASK702A
12699	6303	MF8803
12739	6546	MF8&EXPECD
2767	7303	NEXT607
16492	4447	NEXT712A
16985	4524	NEXT722A
11039	6377	NEXTPAT500
7712	4512	OPA134A1
7950	4215	OPA136B1
7938	4542	OPB136A1
14774	5311	P&CC-DC610A1
15005	5342	P&CC-DC610C1
15212	7162	P&CCTOSR3-0
12187	6241	P&MI507A
12164	6242	P&MID507A
12612	7103	P&SEQLOD04

LINE	LOCN	SYMBOL
14882	5324	LOAD01-61081
15108	5510	LOAD01-610D1
3213	5744	LOAD013A
3563	5736	LOAD016A
3754	5720	LOAD021A
3994	5704	LOAD025A
4232	5672	LOAD031A
14266	4277	LOAD03537A
5242	5152	LOAD04-102A
4775	5634	LOAD043A
4997	5550	LOAD047A
5026	5552	LOAD050A
14813	5315	LOAD06-610A2
5368	5167	LOAD13-103A
14252	4574	LOAD16537A
11954	6620	LOADB50SA
16069	5443	LOADBA702A
5284	5157	LOADD102B
5430	5174	LOADD103B
5580	5204	LOADD104B
12184	6245	LOADDCS507A
11585	6622	LOADFLAG504A
10710	7268	LOADFPSCC02
11017	6132	LOADIR500
14876	5534	LOADIR610B1
17091	6754	LOADIR710A
17647	4400	LOADIR740D
9161	6032	LOADRE8368A
9505	6070	LOADRE8371A
16467	5486	LOADRET712A
16958	5486	LOADRET722A
5723	5216	LOADSR105D
14203	6444	LOADSR536F
15747	5425	LODBRR624A
15731	5582	LODIR624A
11270	6191	MASK503A
17642	7202	MASK762A
12706	6304	MF8804
12742	6547	MF8&EXPEC1
2833	5245	NEXT010
16621	7365	NEXT713A
17891	4467	NEXTD762A
8536	4253	NEXTPATA350
7771	4176	OPA134B1
7607	4510	OPB133A1
9665	6100	PAGE372B
14824	5317	P&CC-DC610A2
15056	5350	P&CC-DC610C2
15217	7163	P&CCTOSR3=0AA
11985	6230	P&LO506A
12596	7100	P&SEQLOD
12618	7104	P&SEQLOD08

LINE	LOCN	SYMBOL	LINE	LOCN	SYMBOL	LINE	LOCN	SYMBOL	LINE	LOCN	SYMBOL
12621	7105	PSSEQLOD06	18626	7253	PSTOD	18082	4421	READVEC7743A	8530	4703	RESETIR350
8435	7022	RESETIR350A	8461	7026	RESETIR380B	8469	7037	RESETIR380C	18679	7263	RESETPROCDAT
18765	7211	RESETUCONP	18795	7215	RESETUCONP5	8764	4718	RESTORE01	8776	7030	RESTORE02
8787	7032	RESTORE03	8788	7033	RESTORE04	9743	6882	RETURN373A	3185	5012	SCOPE013
3341	5072	SCOPE013	3452	5027	SCOPE014	3513	5011	SCOPE015	3578	5035	SCOPE016
3642	5040	SCOPE017	3726	5044	SCOPE020	3812	5050	SCOPE021	3986	5082	SCOPE022
3899	5084	SCOPE023	3965	5057	SCOPE024	4008	5061	SCOPE025	4053	5063	SCOPE026
4095	5085	SCOPE027	4204	5072	SCOPE030	4288	5078	SCOPE031	4333	5100	SCOPE032
4378	5102	SCOPE033	4421	5104	SCOPE034	4469	5106	SCOPE039	4510	5110	SCOPE036
4554	5112	SCOPE037	4619	5115	SCOPE040	4689	5120	SCOPE041	4747	5123	SCOPE042
4811	5126	SCOPE043	4855	5130	SCOPE044	4900	5133	SCOPE045	4967	5138	SCOPE048
5056	5141	SCOPE047	5129	5145	SCOPE050	5160	5150	SCOPE101	5350	8185	SCOPE107
5496	5202	SCOPE103	5563	5206	SCOPE104	5758	5221	SCOPE105	5920	4007	SCOPE114A
6092	4022	SCOPE115B	6217	4032	SCOPE118D	6168	4042	SCOPE116C	6474	4082	SCOPE116D
6870	4066	SCOPE117C	6813	4077	SCOPE120B	6976	4114	SCOPE121C	7129	4126	SCOPE121D
7223	4136	SCOPE122A	7380	4144	SCOPE130B	7489	4196	SCOPE131B	7587	4162	SCOPE132B
7693	4171	SCOPE133B	7806	4203	SCOPE134B	7919	4211	SCOPE135B	8025	4220	SCOPE136B
8270	4237	SCOPE320	8520	4252	SCOPE350	8585	4268	SCOPE351	8756	4275	SCOPE352
9453	6075	SCOPE371	9700	5501	SCOPE372B	9828	8102	SCOPE373	9975	5233	SCOPE374A
10048	5242	SCOPE374B	10121	5252	SCOPE374C	10194	5261	SCOPE376D	10287	8270	SCOPE374E
10340	5277	SCOPE374F	10633	5304	SCOPE377A	10917	6128	SCOPE410	11193	6146	SCOPE500F
11550	6173	SCOPE503	11820	6214	SCOPE504	11935	6226	SCOPE505	12121	6240	SCOPE506
12338	6262	SCOPE507	12549	6300	SCOPE510	12800	6301	SCOPE511A	12856	6305	SCOPE511B
13066	6313	SCOPE512A	13170	6323	SCOPE512C	13275	6333	SCOPE512E	13469	6346	SCOPE520
13601	6356	SCOPE533B	13677	6364	SCOPE534B	13752	6372	SCOPE534D	13866	6406	SCOPE534G
13959	6417	SCOPE535B	14052	6430	SCOPE536B	14141	6440	SCOPE536D	14228	6450	SCOPE536F
14434	4304	SCOPE537A	14539	5305	SCOPE551	14693	5323	SCOPE610A	14970	5340	SCOPE610B
15081	4354	SCOPE610C	15180	5365	SCOPE610D	15702	5423	SCOPE623	15884	5224	SCOPE624
16027	5436	SCOPE701	16120	5433	SCOPE702	16210	5445	SCOPE710	16443	5446	SCOPE711
16578	5447	SCOPE712	16711	5451	SCOPE713	16914	8484	SCOPE721	17045	4327	SCOPE722
17299	4352	SCOPE730	17481	4370	SCOPE731	17675	4402	SCOPE740	17801	4410	SCOPE761
18163	4430	SCOPE763	10533	7061	SECORD378A	10605	7250	SERVICETOD	16748	7174	SETADR720A
16953	7173	SETADR721A	16964	6470	SETADR721B	16361	5373	SETBRK621A	16738	5424	SETBRK624A
14780	5312	SETBUSA610A1	14830	5320	SETBUSA610A2	14893	5326	SETBUSA610B1	14948	5335	SETBUSA610B2
15011	5343	SETBUSA610C1	15062	5351	SETBUSA610C2	15119	5356	SETBUSA610D1	15158	5362	SETBUSA610D2
10673	6560	SETBYTER410	10680	7060	SETBYTER410	10908	7066	SETBYTER410	10691	6000	SETBYTER610
10697	6105	SETBYTEG410	12320	6260	SETCIN507F	7210	4132	SETD122A4	8111	4221	SETD320A
8169	4226	SETD320C	8230	4233	SETD320E	15760	6306	SETD624A	9822	6075	SETDC372A
9671	6000	SETDC372B	8932	6010	SETDDC382A	9040	6020	SETDDC383A	9099	6025	SETDDC384A
9126	6027	SETDDC384B	9285	6044	SETDDC386C	9436	6062	SETDDC370C	14248	6681	SETENJT637A
10975	6676	SETFETCHR500	10982	6126	SETFETCHR500	10987	6137	SETFETCHE500	10992	6130	SETFETCHF500
10998	6131	SETFETCHG500	11867	6216	SETFLAG510A	12497	6271	SETFLAG510D	15382	5377	SETFLG621A
15769	6453	SETFLG624A	9751	7034	SETIR373A	10602	5302	SETIR376A	13031	6964	SETIRS12A1
13088	6536	SETIRS12B1	13137	6317	SETIRS12C1	13192	6636	SETIRS12D1	13241	6327	SETIRS12E1
3039	5760	SETIRA	16175	6312	SETJAM720A	16753	7175	SETJAM720A	16858	7200	SETJAM721A
16969	4323	SETJAM722A	16862	5442	SETKT703A	16101	6481	SETKT703B	16859	8441	SETLSD703A
8105	4700	SETONE320A	8164	4225	SETONE330C	8224	4232	SETONE330E	12370	6406	SETONE8510A
17907	4463	SETPP6-762A	17912	4413	SETPRI762A	17986	7286	SETPRI762C	12301	6256	SETPS507F
13039	6310	SETPSFLAG512A1	13094	6314	SETPSFLAG512B1	13143	6320	SETPSFLAG512C1	13198	6324	SETPSFLAG512D1
13247	6330	SETPSFLAG512E1	8814	5570	SETRES364A	8984	6107	SETRES364B	9163	6031	SETRES365A
9216	6036	SETRES366A	9324	6050	SETRES367A	9387	6064	SETRES370A	9497	6067	SETRES371A
14285	7135	SETRESA537A	14377	7370	SETRESA537A	15316	5375	SETRET621A	15784	5426	SETRET624A

LINE	LOCN	SYMBOL	LINE	LOCN	SYMBOL	LINE	LOCN	SYMBOL	LINE	LOCN	SYMBOL
10519	5506	SETSP175A	9757	7037	SETSRJ73A	12310	6287	SETSR507F	14481	4306	SETSR551A
14506	4565	SETSR551B	14546	4561	SETSR551C	15395	5401	SETSR621A	15771	6454	SETSR624A
11965	6225	SETUCONS06A	13001	6555	SETUP512A	13007	6307	SETUP512B	13338	6334	SETUP520A
13367	6336	SETUP520B	13396	6340	SETUP520C	13424	6342	SETUP520D	13492	6344	SETUP520E
5862	7015	SETUPC8P00A	5852	7013	SETUPC8P05A	5887	7016	SETUPC8P07A	5847	7012	SETUPC8P12A
5842	7011	SETUPC8P14A	5837	7010	SETUPC8P15A	5892	7003	SETUPC8P16A	8026	5843	SETUPC8P17A
15197	7156	SETUPP8CC#DC	15202	7161	SETUPP8CC#DC02	17249	4344	SETSER7300	10615	5746	SFDFT08R
10428	7047	SFDFT08RA	10423	7040	SFDFT08RA	10434	7050	SFDFT08RB	10440	7051	SFDFT08RC
10446	7057	SFDFT08RD	10452	7053	SFDFT08RE	10458	7054	SFDFT08RF	10464	7059	SFDFT08RG
10470	7056	SFDFT08RH	10476	7057	SFDFT08RI	9068	6021	SHIFT363A	9230	6040	SHIFT366A
9290	6045	SHIFT368C	9381	6056	SHIFT370A	9442	6063	SHIFT370C	9511	6071	SHIFT371A
18747	7266	SPRINTOIR	18777	7272	SRINTOIRS	8343	4240	STARTI50	2816	7000	SUBR010
9439	6077	SUBRA372A	9489	5223	SUBRA372B	13482	7132	SUCBRTEST01	13487	7132	SUCBRTEST02
13492	7133	SUCBRTEST03	2633	4000	TEST001	2684	8002	TEST002	2673	6631	TEST003
2692	5525	TEST004	2711	5146	TEST005	2730	4474	TEST006	2749	4377	TEST007
2802	4775	TEST010	2847	5777	TEST011	3031	5775	TEST012A	3069	5773	TEST012B
3084	5771	TEST012C	3109	8767	TEST012D	3127	5768	TEST012E	3147	5764	TEST012F
3169	5763	TEST012G	3205	5761	TEST013A	3233	5757	TEST013B	3253	5755	TEST013C
3272	5753	TEST013D	3291	5751	TEST013E	3312	5750	TEST013F	3331	5747	TEST013G
3368	5745	TEST014A	3394	5752	TEST014B	3414	8762	TEST014C	3436	5710	TEST014D
3473	5743	TEST015A	3500	5772	TEST015B	3519	5722	TEST015C	3585	5741	TEST016A
3598	5737	TEST017A	3626	5735	TEST017B	3662	5733	TEST020A	3689	5731	TEST020B
3711	5727	TEST020C	3746	5725	TEST021A	3775	5723	TEST021B	3797	5774	TEST021C
3833	5721	TEST022A	3877	5717	TEST023A	3920	5715	TEST024A	3949	5711	TEST024B
3985	5707	TEST025A	4029	5705	TEST026A	4074	5701	TEST027A	4117	5701	TEST030A
4147	5712	TEST030B	4165	5713	TEST030C	4189	5734	TEST030D	4224	5677	TEST031A
4253	5675	TEST031B	4289	5673	TEST032A	4317	5671	TEST032B	4354	5667	TEST033A
4398	5665	TEST034A	4442	5663	TEST036A	4486	5661	TEST036A	4531	5657	TEST037A
4574	5655	TEST040A	4602	5653	TEST040B	4638	5651	TEST041A	4667	5647	TEST041B
4703	5645	TEST042A	4731	5643	TEST042B	4767	5641	TEST043A	4796	5637	TEST043B
4833	5635	TEST044A	4878	5633	TEST045A	4920	5631	TEST046A	4950	5770	TEST046B
4988	5561	TEST047A	5019	5670	TEST047B	5039	5730	TEST047C	5077	5551	TEST050A
5108	5726	TEST050B	5161	5853	TEST101A	5225	5821	TEST102A	5277	5874	TEST102B
5301	5864	TEST102C	5330	5854	TEST102D	5370	5931	TEST103A	5423	5572	TEST103B
5449	5576	TEST103C	5476	5674	TEST103D	5516	5547	TEST104A	5543	5646	TEST104B
5592	5541	TEST105A	5619	5810	TEST106A1	5642	5652	TEST105B	5686	5600	TEST105B1
5690	5642	TEST105C	5716	5636	TEST105D	5743	5620	TEST105E	5874	4807	TEST114A
5906	4603	TEST114A2	5939	4601	TEST115A1	5993	4616	TEST115A2	6000	4615	TEST115A3
5967	4604	TEST115A4	6021	4614	TEST115B1	6060	4613	TEST115B2	6077	4612	TEST115B3
6044	4602	TEST115B4	6104	4611	TEST115C1	6126	4617	TEST115C2	6143	4627	TEST115C3
6164	4626	TEST115D1	6186	4625	TEST115D2	6203	4624	TEST115D3	6235	4623	TEST116A1
6257	4621	TEST116A2	6271	4620	TEST116A3	6288	4637	TEST116A4	6305	4636	TEST116A5
6326	4606	TEST116B	6359	4635	TEST116C1	6381	4633	TEST116C2	6396	4632	TEST116C3
6413	4631	TEST116C4	6429	4630	TEST116C5	6482	4645	TEST116D	6498	4647	TEST117A1
6517	4644	TEST117A2	6534	4643	TEST117A3	6549	4642	TEST117A4	6570	4641	TEST117B1
6592	4640	TEST117B2	6609	4657	TEST117B3	6633	4656	TEST117C1	6655	4677	TEST117C2
6685	4867	TEST120A1	6707	4655	TEST120A2	6724	4654	TEST120A3	6745	4653	TEST120B1
6767	4652	TEST120B2	6784	4651	TEST120B3	6799	4650	TEST120B4	6832	4665	TEST121A1
6854	4678	TEST121A2	6871	4675	TEST121A3	6885	4674	TEST121A4	6906	4663	TEST121B1
6928	4662	TEST121B2	6945	4661	TEST121B3	6960	4660	TEST121B4	6987	4707	TEST121C1
7009	4605	TEST121C2	7025	4717	TEST121C3	7040	4716	TEST121C4	7061	4727	TEST121D1
7081	4726	TEST121D2	7100	4737	TEST121D3	7115	4736	TEST121D4	7148	4671	TEST122A1

LINE	LOCN	SYMBOL
7172	4570	TEST122A2
7314	4550	TEST130A2
7423	4546	TEST131A2
7527	4516	TEST132A2
7630	4500	TEST133A2
7742	4503	TEST134A2
7854	4506	TEST135A2
7961	4476	TEST136A2
8138	4552	TEST320R
8256	4556	TEST320F
8471	4742	TEST350C
8612	4723	TEST351C
8713	4724	TEST352C
8894	6767	TEST361E
9010	6755	TEST362E
9119	6745	TEST364B
9256	6739	TEST366B
9405	6723	TEST370B
9538	6713	TEST371B
9814	6572	TEST373B
10105	5524	TEST374C2
10512	5817	TEST375A
10666	5507	TEST410
10879	6727	TEST410D
11100	6726	TEST500C
11318	6630	TEST503B
11407	6700	TEST503E
11495	6742	TEST503I
11647	6730	TEST504B
11742	6672	TEST504F
11847	6623	TEST505A
12034	6754	TEST506B
12144	6613	TEST507A
12265	6674	TEST507E
1241*	6616	TEST510C
12524	6610	TEST510F
12389	7112	TEST511A4
12645	7120	TEST511B4
13108	6546	TEST512B2
13212	6576	TEST512D2
13360	6764	TEST520B
13528	6625	TEST533A
13691	6665	TEST534C
13844	6757	TEST534G
14019	6640	TEST535B
14196	6642	TEST536F
14539	4566	TEST551C
14910	5612	TEST610B2
15145	5574	TEST610D2
15344	5617	TEST621A
15495	5605	TEST621E
15605	5601	TEST622A

LINE	LOCN	SYMBOL
7188	4734	TEST132A3
7334	4530	TEST130B1
7443	4546	TEST131B1
7547	4517	TEST132B1
7652	4501	TEST133B1
7764	4504	TEST134B1
7877	4507	TEST135B1
7983	4475	TEST136B1
8157	4562	TEST320C
8337	4701	TEST350
8499	4743	TEST350D
8634	4712	TEST351D
8735	4725	TEST352D
8917	6765	TEST362A
9033	6753	TEST363A
9146	6743	TEST365A
9278	6733	TEST366C
9429	6721	TEST370C
9606	6571	TEST372A
9898	6563	TEST374
10178	5500	TEST374D2
10583	5625	TEST378B
10790	6711	TEST410A
10911	6705	TEST410E
11136	6746	TEST500D
11330	6632	TEST503C
11433	6710	TEST503F
11513	6756	TEST503J
11667	6740	TEST504C
11762	6702	TEST504G
11891	6712	TEST505B
12054	6744	TEST506C
12201	6724	TEST507B
12286	6647	TEST507F
12438	6600	TEST510D
12755	6607	TEST511A
12812	6551	TEST511B1
13028	6542	TEST512A1
13130	6556	TEST512[illegible]1
13234	6577	TEST512E1
13388	6772	TEST520C
13569	6633	TEST533B
13725	6635	TEST534D
13879	6661	TEST535A
14067	6655	TEST536C
14413	4572	TEST617A
14744	5627	TEST610A1
14987	5535	TEST610C1
15262	5511	TEST620A
15440	5613	TEST621B
15521	5603	TEST621F
15634	5577	TEST622B

LINE	LOCN	SYMBOL
7202	4735	TEST132A4
7368	4532	TEST130B2
7474	4780	TEST131B2
7572	4472	TEST132B2
7678	4502	TEST133B2
7791	4506	TEST134B2
7904	4477	TEST135B2
8010	4473	TEST136B2
8197	4537	TEST320D
8417	4702	TEST350A
8569	4738	TEST351A
8670	4706	TEST352A
8827	6513	TEST361A
8952	6763	TEST362B
9069	6731	TEST363B
9187	6741	TEST365B
9317	6731	TEST367A
9467	6717	TEST370D
9658	6701	TEST372B
9958	6604	TEST374A2
10251	5502	TEST374E2
10598	5513	TEST376A
10814	6704	TEST410B
10968	6561	TEST500
11164	6736	TEST500E
11379	6670	TEST503D
11493	6720	TEST503G
11931	6646	TEST503K
11988	6750	TEST504D
11703	6706	TEST504H
11912	6722	TEST505C
12076	6734	TEST506D
12321	6611	TEST507C
12743	6603	TEST510A
12478	6617	TEST510DA
12787	7076	TEST511A2
12821	7106	TEST511B2
13088	6644	TEST512A2
13187	6646	TEST512C2
13361	6535	TEST512E2
13417	6752	TEST520D
13615	6667	TEST534A
13789	6663	TEST534E
13926	6637	TEST535B
14093	6641	TEST536D
14466	6675	TEST551A
14906	5628	TEST610A2
15088	5602	TEST610C2
15210	5623	TEST620B
15483	5611	TEST621C
15680	5746	TEST621G
15687	6176	TEST622C

LINE	LOCN	SYMBOL
7263	4673	TEST130A1
7392	4541	TEST131A1
7502	4521	TEST132A1
7600	4523	TEST133A1
7708	4511	TEST134A1
7818	4513	TEST135A1
7931	4515	TEST136A1
8098	4543	TEST320A
8217	4554	TEST320E
8444	4703	TEST350B
8591	4722	TEST351B
8692	4713	TEST352B
8873	6771	TEST361D
8989	6757	TEST362D
9091	6747	TEST364A
9209	6737	TEST366A
9360	6725	TEST370A
9490	6715	TEST371A
9738	6541	TEST373A
10032	5606	TEST374B2
10324	5504	TEST376F2
10619	5614	TEST376A1
10844	6707	TEST410C
11070	6673	TEST500A
11238	6677	TEST503A
11359	6601	TEST503DA
11474	6732	TEST503H
11578	6613	TEST504A
11716	6760	TEST504E
11804	6716	TEST504I
11952	6621	TEST506A
12099	6605	TEST506E
12243	6557	TEST507D
12396	6675	TEST510B
12497	6604	TEST510E
12778	7110	TEST511A3
12834	7116	TEST511B3
13081	6565	TEST512B1
13185	6537	TEST512D1
13322	6627	TEST520A
13445	6778	TEST520E
13650	6634	TEST534B
13805	6636	TEST534F
13973	6657	TEST536A
14154	6653	TEST536E
14498	4576	TEST551B
14869	5533	TEST610B1
15101	5545	TEST610D1
15309	5621	TEST620C
15476	5607	TEST621D
15572	5754	TEST621H
15680	5573	TEST623

LINE	LOCN	SYMBOL
15724	5571	TEST624A
15925	5561	TEST701A
16040	5477	TEST702A
16236	6645	TEST710C
16374	6534	TEST711B
16512	6505	TEST712B
16640	5616	TEST713B
16788	6520	TEST720B
16918	4577	TEST721C
17084	4471	TEST730A
17233	4461	TEST730D
17401	4456	TEST731C
17591	4453	TEST740R
17765	4771	TEST761B
17924	4751	TEST762B
18031	4711	TEST762F
18144	4710	TEST763D
11088	6671	TEST0500
15400	5522	UBRK621A
18301	6774	VFY001
18391	7375	VFY005
8371	7016	WRITE8F350
9941	5227	ZERO37442
10087	5246	ZERO374C2
10233	5264	ZERO374E2
17734	4772	ZERODBRT761A
10388	7044	ZFRO8F
10375	7042	ZFRO8FDF
19256	7402	ZTARGET402
19264	7404	ZTARGET404
19272	7412	ZTARGET412
19280	7416	ZTARGET416
19288	7422	ZTARGET422
19296	7426	ZTARGET426
19304	7432	ZTARGET432
19312	7436	ZTARGET436
19320	7442	ZTARGET442
19328	7446	ZTARGET446
19336	7452	ZTARGET452
19344	7456	ZTARGET456
19352	7462	ZTARGET462
19360	7466	ZTARGET466
19368	7472	ZTARGET472
19376	7476	ZTARGET476
19384	7502	ZTARGET502
19392	7506	ZTARGET506
19400	7512	ZTARGET512
19408	7516	ZTARGET516
1941A	7522	ZTARGET522
19424	7526	ZTARGET526
19432	7532	ZTARGET532
19440	7536	ZTARGET536

LINE	LOCN	SYMBOL
15817	5565	TEST624B
15962	6523	TEST701B
16089	6761	TEST702B
16265	6574	TEST710D
16404	6530	TEST711C
16530	6524	TEST712C
16664	6522	TEST713C
16813	4760	TEST720C
16951	5463	TEST722A
17149	4466	TEST730B
17286	4462	TEST730E
17428	4455	TEST731D
17619	4452	TEST740C
17786	4774	TEST761C
17946	4740	TEST762C
18062	4761	TEST763A
11292	6714	TESTA50JA
11031	6777	TESTINHB8P500
15776	6255	UBRK624A
18308	4412	YFY002
18340	7374	YFY006
17961	4315	ZAPD762A
9992	5556	ZERO374B1
10139	5514	ZERO37401
10284	5516	ZERO374F1
12831	6276	ZERODFLAG510E
10365	7036	ZERO8F02DF04
10381	7043	ZERO8FDFA
19258	7403	ZTARGET403
19266	7407	ZTARGET407
19274	7413	ZTARGET413
19282	7417	ZTARGET417
19290	7423	ZTARGET423
19298	7427	ZTARGET427
19306	7433	ZTARGET433
19314	7437	ZTARGET437
19322	7443	ZTARGET443
19330	7447	ZTARGET447
19338	7453	ZTARGET453
19346	7457	ZTARGET457
19354	7463	ZTARGET463
19362	7467	ZTARGET467
19370	7473	ZTARGET473
19378	7477	ZTARGET477
19386	7503	ZTARGET503
19394	7507	ZTARGET507
19402	7513	ZTARGET513
19410	7517	ZTARGET517
19418	7523	ZTARGET523
19426	7527	ZTARGET527
19434	7533	ZTARGET533
19442	7537	ZTARGET537

LINE	LOCN	SYMBOL
15839	5778	TEST624C
15981	6770	TEST701C
16157	6478	TEST710A
16294	6875	TEST710E
16427	6536	TEST711D
16589	6583	TEST712D
16689	6601	TEST713D
16840	4671	TEST721A
17004	6064	TEST722B
17188	4460	TEST730C
17331	4786	TEST731A
17467	4464	TEST731E
17683	4461	TEST740D
17689	4773	TEST762A
17980	4741	TEST762D
18108	4720	TEST763B
11621	6766	TESTA504A
11033	6133	TESTINHB8P5n0
13389	6634	UCON520A
18313	4258	YFY003
10755	7386	WORD410
17122	4134	ZAPDBUF730A
10014	5536	ZERO37482
10160	5259	ZERO374D2
10306	5273	ZERO374F2
15824	5427	ZERODIR624B
10370	7041	ZERO8F04DF02
19282	7400	ZTARGET400
19260	7404	ZTARGET404
19268	7410	ZTARGET410
19276	7414	ZTARGET414
19284	7420	ZTARGET420
19292	7424	ZTARGET424
19300	7430	ZTARGET430
19308	7434	ZTARGET434
19316	7440	ZTARGET440
19324	7444	ZTARGET444
19332	7450	ZTARGET450
19340	7454	ZTARGET454
19348	7460	ZTARGET460
19356	7464	ZTARGET464
19364	7470	ZTARGET470
19372	7474	ZTARGET474
19380	7500	ZTARGET500
19388	7504	ZTARGET504
19396	7510	ZTARGET510
19404	7514	ZTARGET514
19412	7520	ZTARGET520
19420	7524	ZTARGET524
19428	7530	ZTARGET530
19436	7534	ZTARGET534
19444	7540	ZTARGET540

LINE	LOCN	SYMBOL
15864	5756	TEST624D
16011	5512	TEST701D
16212	6644	TEST710B
16328	5473	TEST711A
16460	5471	TEST712A
16594	5467	TEST713A
16730	5465	TEST720A
16894	4573	TEST721B
17029	4465	TEST722C
17211	4450	TEST730C1
17363	4457	TEST731B
17556	4744	TEST740A
17728	4745	TEST761A
17899	4853	TEST762A1
19006	4730	TEST762E
18128	4721	TEST763C
10707	6773	TESTD410
18089	4424	TESTVECT763A
18103	4733	VECTLOAD763A
18321	4433	VFY004
8364	4002	WRITEDF350
9919	5536	ZERO374A1
10065	5846	ZERO374C1
10211	5526	ZERO374E1
12504	6274	ZERODS10E
17739	4403	ZERO1T761A
10392	7045	ZERO8FA
19284	7401	ZTARGET401
19262	7405	ZTARGET405
19270	7411	ZTARGET411
19278	7415	ZTARGET415
19286	7421	ZTARGET421
19294	7425	ZTARGET425
19302	7431	ZTARGET431
19310	7435	ZTARGET435
19318	7441	ZTARGET441
19326	7445	ZTARGET445
19334	7451	ZTARGET451
19342	7455	ZTARGET455
19350	7461	ZTARGET461
19358	7465	ZTARGET465
19366	7471	ZTARGET471
19374	7475	ZTARGET475
19382	7501	ZTARGET501
19390	7505	ZTARGET505
19398	7511	ZTARGET511
19406	7515	ZTARGET515
19414	7521	ZTARGET521
19422	7525	ZTARGET525
19430	7531	ZTARGET531
19438	7535	ZTARGET535
19446	7541	ZTARGET541

LINE	LOCN	SYMBOL	LINE	LOCN	SYMBOL	LINE	LOCN	SYMBOL	LINE	LOCN	SYMBOL
19448	7542	ZTARGET542	19450	7543	ZTARGET543	19452	7544	ZTARGET544	19454	7545	ZTARGET545
19456	7546	ZTARGET546	19458	7547	ZTARGET547	19460	7550	ZTARGET550	19462	7551	ZTARGET551
19464	7552	ZTARGET552	19466	7553	ZTARGET553	19468	7554	ZTARGET554	19470	7555	ZTARGET555
19472	7556	ZTARGET556	19474	7557	ZTARGET557	19476	7560	ZTARGET560	19478	7561	ZTARGET561
19480	7562	ZTARGET562	19482	7563	ZTARGET563	19484	7564	ZTARGET564	19486	7565	ZTARGET565
19488	7566	ZTARGET566	19490	7567	ZTARGET567	19492	7570	ZTARGET570	19494	7571	ZTARGET571
19496	7572	ZTARGET572	19498	7573	ZTARGET573	19500	7574	ZTARGET574	19502	7575	ZTARGET575
19504	7576	ZTARGET576	19506	7577	ZTARGET577	19508	7600	ZTARGET600	19510	7601	ZTARGET601
19512	7602	ZTARGET602	19514	7603	ZTARGET603	19516	7604	ZTARGET604	19518	7605	ZTARGET605
19520	7606	ZTARGET606	19522	7607	ZTARGET607	19524	7610	ZTARGET610	19526	7611	ZTARGET611
19528	7612	ZTARGET612	19530	7613	ZTARGET613	19532	7614	ZTARGET614	19534	7615	ZTARGET615
19534	7616	ZTARGET616	19538	7617	ZTARGET617	19540	7620	ZTARGET620	19542	7621	ZTARGET621
19544	7622	ZTARGET622	19546	7623	ZTARGET623	19548	7624	ZTARGET624	19550	7625	ZTARGET625
19552	7626	ZTARGET626	19554	7627	ZTARGET627	19556	7630	ZTARGET630	19558	7631	ZTARGET631
19560	7632	ZTARGET632	19562	7633	ZTARGET633	19564	7634	ZTARGET634	19566	7635	ZTARGET635
19568	7636	ZTARGET636	19570	7637	ZTARGET637	19572	7640	ZTARGET640	19574	7641	ZTARGET641
19576	7642	ZTARGET642	19578	7643	ZTARGET643	19580	7644	ZTARGET644	19582	7645	ZTARGET645
19584	7646	ZTARGET646	19586	7647	ZTARGET647	19588	7650	ZTARGET650	19590	7651	ZTARGET651
19592	7652	ZTARGET652	19594	7653	ZTARGET653	19596	7654	ZTARGET654	19598	7655	ZTARGET655
19600	7656	ZTARGET656	19602	7657	ZTARGET657	19604	7660	ZTARGET660	19606	7661	ZTARGET661
19608	7662	ZTARGET662	19610	7663	ZTARGET663	19612	7664	ZTARGET664	19614	7665	ZTARGET665
19616	7666	ZTARGET666	19618	7667	ZTARGET667	19620	7670	ZTARGET670	19622	7671	ZTARGET671
19624	7672	ZTARGET672	19626	7673	ZTARGET673	19628	7674	ZTARGET674	19630	7675	ZTARGET675
19632	7676	ZTARGET676	19634	7677	ZTARGET677	19636	7700	ZTARGET700	19638	7701	ZTARGET701
19640	7702	ZTARGET702	19642	7703	ZTARGET703	19644	7704	ZTARGET704	19646	7705	ZTARGET705
19648	7706	ZTARGET70A	19650	7707	ZTARGET707	19652	7710	ZTARGET710	19654	7711	ZTARGET711
19656	7712	ZTARGET712	19658	7713	ZTARGET713	19660	7714	ZTARGET714	19662	7715	ZTARGET715
19664	7716	ZTARGET716	19666	7717	ZTARGET717	19668	7720	ZTARGET720	19670	7721	ZTARGET721
19672	7722	ZTARGET722	19674	7723	ZTARGET723	19676	7724	ZTARGET724	19678	7725	ZTARGET725
19680	7726	ZTARGET726	19682	7727	ZTARGET727	19684	7730	ZTARGET730	19686	7731	ZTARGET731
19688	7732	ZTARGET732	19690	7733	ZTARGET733	19692	7734	ZTARGET734	19694	7735	ZTARGET735
19696	7736	ZTARGET736	19698	7737	ZTARGET737	19700	7740	ZTARGET740	19702	7741	ZTARGET741
19704	7742	ZTARGET742	19706	7743	ZTARGET743	19708	7744	ZTARGET744	19710	7745	ZTARGET745
19712	7746	ZTARGET746	19714	7747	ZTARGET747	19716	7750	ZTARGET750	19718	7751	ZTARGET751
19720	7752	ZTARGET752	19722	7753	ZTARGET753	19724	7754	ZTARGET754	19726	7755	ZTARGET755
19728	7756	ZTARGET756	19730	7757	ZTARGET757	19732	7760	ZTARGET760	19734	7761	ZTARGET761
19736	7762	ZTARGET762	19738	7763	ZTARGET763	19740	7764	ZTARGET764	19742	7765	ZTARGET765
19744	7766	ZTARGET766	19746	7767	ZTARGET767	19748	7770	ZTARGET770	19750	7771	ZTARGET771
19752	7772	ZTARGET772	19754	7773	ZTARGET773	19756	7774	ZTARGET774	19758	7775	ZTARGET775
19760	7776	ZTARGET776	19762	7777	ZTARGET777						

LOCN	----0-----	----1-----	----2-----	----3-----	----4-----	----5-----	----6-----	----7-----
4000	TEXT001	INITIALIZE12	WRITEDF350	ALU114A	GETSEMOE#114A	GOBUT114A	GOBUT114A2	SCOPE114A
4010	GETALTN115A1	GOBUT115A1	GOBUT115A4	GOBUT115A2	GOBUT115A3	ALU115D1	GOBUT115B1	GOBUT115B4
4020	GOBUT115B2	GOBUT115B3	SCOPE115B	GOBUT115C1	GOBUT115C2	GOBUT115C3	ALU115D1	GOBUT115D1
4030	GOBUT115D2	GOBUT115D3	SCOPE115D	GOBUT116A1	GOBUT116A2	GOBUT116A3	GOBUT116A4	GOBUT116A5
4040	ALU116B	GOBUT116B	SCOPE116C	GOBUT116C1	GOBUT116C2	GOBUT116C3	GOBUT116C4	GOBUT116C5
4050	ALU116D	GOBUT116D	SCOPE116D	GOBUT117A1	GOBUT117A2	GOBUT117A3	GOBUT117A4	ALU117B1
4060	GOBUT117B1	GOBUT117B2	GOBUT117B3	ALU117C1	GOBUT117C2	GOBUT117C2	SCOPE117C	GOBUT120A1
4070	GOBUT120A2	GOBUT120A3	ALU120B1	GOBUT120B1	GOBUT120B2	GOBUT120B3	GOBUT120B4	SCOPE120B
4100	LOAD010	INITIALIZE10	INITIALIZE11	GOBUT121A1	GOBUT121A2	GOBUT121A3	GOBUT121A4	ALU121B1
4110	GOBUT121B1	GOBUT121B2	GOBUT121B3	GOBUT121B4	SCOPE121C	GOBUT121C1	GOBUT121C2	GOBUT121C3
4120	GOBUT121C4	ALU121D1	GOBUT121D1	GOBUT121D2	GOBUT121D3	GOBUT121D4	SCOPE121D	GOBUT122A1
4130	GOBUT122A2	GOBUT122A3	SETD122A4	GOBUT122A4	SCOPE122A	CONP130A1	GOBUT130A1	GOBUT130A2
4140	ARITH130B1	COMP130B1	GOBUT130B1	GOBUT130B2	SCOPE130B	CONP131A1	GOBUT131A1	GOBUT131A2
4150	ARITH131B1	COMP131B1	GOBUT131B1	GOBUT131B2	SCOPE131B	GOBUT132A1	GOBUT132A2	ARITH132B1
4160	GOBUT132B1	GOBUT132B2	SCOPE132B	DOPA133A1	GOTEST133A1	GOBUT133A2	OPA133B1	GOTEST133B1
4170	GOBUT133B2	SCOPE133B	OPB134A1	DOPA134A1	GOTEST134A1	GOBUT134A2	OPA134B1	GOTEST134B1
4200	GOBUT134B2	SCOPE134B	OPB135A1	DOPA135A1	GOTEST135A1	GOBUT135A2	OPA135B1	GOTEST135B1
4210	GOBUT135B2	SCOPE135B	DOPA136A1	GOTEST136A1	GOBUT136A2	OPA136B1	GOTEST136B1	GOBUT136B2
4220	SCOPE136B	SETD320A	GETDC320A	GOBUT320A	GOBUT320B	SETONE320C	SETD320C	GETDC320C
4230	GOBUT320C	GOBUT320D	SETONE320E	SETD320E	SETDC320E	GOBUT320E	GOBUT320F	SCOPE320
4240	START350	LOADBRD350	BL1=517A	GOBUT350A	CONP350B	GOBUT350B	CONP350C	GOBUT350C
4250	CONP350D	GOBUT350D	SCOPE350	NEXTPAYA350	LOAD06=351	VFY003	GOBUT351A	CONP351B
4260	GOBUT351A	CONP351C	GOBUT351C	CONP351D	GOBUT351D	SCOPE351	GOBUT352A	CONP352B
4270	GOBUT352B	CONP352C	GOBUT352C	CONP352D	GOBUT352D	SCOPE352	LOAD145537A	LOAD035137A
4300	LOAD035137A	COUNTER01	CONP537A	GOBUT537A	SCOPE537A	LOADIN551A	SETSR551A	LOADADR710A
4310	COUNTER02	BUSFCN711A	SETJAM710A	BUSFCN710A	GENADR712A	EAPD762A	BUSFCN712A	GOTEST720C
4320	LOADRET721A	GOTEST721B	GOTEST721C	SETJAM722A	BUSFCN722A	GOTEST722B	GOTEST722C	SCOPE722
4330	LOADDATA730A	BUSFCN730A	GETDBUF730A	CONP730A	SAVDBUF730A	GOBUT730A	EXPEC730B	BUSFCN730B
4340	GETIT730B	GOBUT730B	GOBUT730C	LOADDATA730D	SETSCR730D	MANGLED730D	BUSFCN730D	GETIT730D
4350	GOBUT730D	GOBUT730E	SCOPE730	GETDBUF731A	CONP731A	GOBUT731A	BUSFCN731B	GETIT731B
4360	GOBUT731B	CONP731C	GOBUT731C	LOADIN731D	BUSFCN731D	MANGLED731D	GOBUT731D	GOBUT731E
4370	SCOPE731	BUSFCN740A	LOADIR740B	BUSFCN740B	BUSFCN740C	GOBUT740C	BUSERROR4	TEST007
4400	LOADIR740D	BUSFCN740D	SCOPE740	ZEROIT761A	CLEAR761A	GOBUT761A	GOBUT761B	GOBUT761C
4410	SCOPE761	DN11L762A	VFY002	SETPRI762A	FILL762C	DELAY762C	GOBUT762D	EXPEC763E
4420	GOGET762E	GOBUT762F	EXPEC763A	READVECT763A	TESTVECT763A	GOBUT763B	GOBUT763C	GOBUT763D
4430	SCOPE763	SOP001	. . . .	VFY004	JAMUPP002C	JAMUPP006	JAMUPP007	JAMUPP010
4440	JAMUPP011	. . . .	. . . .	GOBUT730C1	. . . .	NEXT710A	NEXT711A	NEXT712A
4450	TEST730C1	TEST740D	TEST740C	TEST740B	TEST731E	TEST731D	TEST731C	TEST731B
4460	TEST730C	TEST730D	TEST730E	SETPR6=762A	TEST722B	TEST722C	TEST730B	NEXTD762A
4470	SETADR722A	TEST730A	TEST13282	TEST134B2	TEST006	TEST136B1	TEST136A2	TEST135B2
4500	TEST133A2	TEST133B1	TEST133B2	TEST134A2	TEST134B1	TEST134B2	TEST138A2	TEST135B1
4510	OPR133A1	TEST134A1	OPA134A1	TEST135A1	OPA135A1	TEST136A1	TEST132A2	TEST132B1
4520	ARITH131A1	TEST132A1	ARITH132A1	TEST133A1	NEXT722A	COUNTER09	COUNTER03	COUNTER10
4530	TEST130B1	COUNTER10B	TEST130B2	COUNTER10A	COUNTER06	COUNTER07	COUNTER08	TEST320D
4540	ARITH130A1	TEST131A1	OPB136A1	TEST320A	TEST131A2	COUNTER04	TEST13181	COUNTER11
4550	TEST130A2	BUMPD762A	TEST320B	TEST762A1	TEST320E	COUNTER12	TEST320F	COUNTER05
4560	GOTEST751B	SETSR551C	TEST320C	DIRTOIRA350	GOTEST551A	SETSR551B	TEST551C	LOAD05=351
4570	TEST127A2	TEST721A	TEST537A	TEST721B	LOAD16537A	TEST551A	TEST551B	TEST721C
4600	GETONES#114A	TEST115A1	TEST115B4	TEST114A2	TEST116A4	TEST121C2	TEST116B	TEST114A
4610	ALU115A1	TEST115C1	TEST115B3	TEST115B2	TEST115B1	TEST115A3	TEST115A2	TEST115C2
4620	TEST116A3	TEST116A2	ALU115C1	TEST116A1	TEST115D3	TEST115D2	TEST115D1	TEST115C3
4630	TEST116C5	TEST116C4	TEST116C3	TEST116C2	ALU116A1	TEST116C1	TEST116A5	TEST116A4

LOCN	----0-----	----1-----	----2-----	----3-----	----4-----	----5-----	----6-----	----7-----
4640	TEST11782	TEST11781	TEST117A4	TEST117A3	TEST117A2	TEST116D	ALU116C1	TEST117A1
4650	TEST120B4	TEST120B3	TEST120B2	TEST120B1	TEST120A3	TEST120A2	TEST117C1	TEST117B3
4660	TEST121B4	TEST121B3	TEST12182	TEST12181	ALU120A1	TEST121A1	ALU117A1	TEST120A1
4670	ALU121C1	TEST122A1	ALU122A1	TEST130A1	TEST121A4	TEST121A3	TEST121A2	TEST117C2
4700	SETONE320A	TEST350	COMP380A	REGETIR380	COMP381A	TEST392A	ALU121A1	TEST121C1
4710	TEST763D	TEST762F	TEST381D	TEST382B	COMP382A	RESTORE01	TEST121C4	TEST121C3
4720	TEST763B	TEST761C	TEST381B	TEST381C	TEST382C	TEST392D	TEST121D2	TEST121D1
4730	TEST762E	ALLOW763A	TEST381A	VECTLOAD743A	TEST121A3	TEST122A4	TEST121D4	TEST121D3
4740	TEST762C	TEST762D	TEST350C	TEST760D	TEST740A	TEST761A	JAMUPP004	JAMUPP005
4750	TEST11782	TEST762B	TEST350A	TEST760B	LOADIR730A	TEST731A	JAMUPP003	JAMUPP003B
4760	TEST720C	TEST763A	GOBUT762B	KILL764A	LOADPC731A	LOADIR740A	GOBUT740B	GOBUT740D
4770	. . . .	TEST761B	ZERODSR761A	TEST763A	TEST731C	TEST010	GOBUT740A	JAMUPP001

LOCN	----0-----	----1-----	----2-----	----3-----	----4-----	----5-----	----6-----	----7-----
5000	SETDC372B	LOAD011	LOAD012A	GOBUT012A	GOBUT012B	GOBUT012C	GOBUT012D	GOBUT012E
5010	GOBUT012F	GOBUT012G	SCOPE012	GOBUT013A	GOBUT013B	GOBUT013C	GOBUT013D	GOBUT013E
5020	GOBUT013F	GOBUT013G	SCOPE013	GOBUT014A	GOBUT014B	GOBUT014C	GOBUT014D	SCOPE014
5030	GOBUT015A	GOBUT015B	GOBUT015C	SCOPE015	GOBUT016A	SCOPE016	GOBUT017A	GOBUT017B
5040	SCOPE017	GOBUT020A	GOBUT020B	GOBUT020C	SCOPE020	GOBUT021A	GOBUT021B	GOBUT021C
5050	SCOPE021	GOBUT022A	SCOPE022	GOBUT023A	SCOPE023	GOBUT024A	GOBUT024B	SCOPE024
5060	GOBUT025A	SCOPE025	GOBUT026A	SCOPE026	GOBUT027A	SCOPE027	GOBUT030A	GOBUT030B
5070	GOBUT030C	GOBUT030D	SCOPE030	GOBUT031A	GOBUT031B	SCOPE031	GOBUT032A	GOBUT032B
5100	SCOPE032	GOBUT033A	SCOPE033	GOBUT034A	SCOPE034	GOBUT035A	SCOPE035	GOBUT036A
5110	SCOPE036	GOBUT037A	SCOPE037	GOBUT040A	GOBUT040B	SCOPE040	GOBUT041A	GOBUT041B
5120	SCOPE041	GOBUT042A	GOBUT042B	SCOPE042	GOBUT043A	GOBUT043B	SCOPE043	GOBUT044A
5130	SCOPE044	GOBUT045A	SCOPE045	GOBUT046A	GOBUT046B	SCOPE046	GOBUT047A	GOBUT047B
5140	GOBUT047C	SCOPE047	GOBUT050A	LOAD050B	GOBUT050B	SCOPE050	TEST008	GOTEST101A
5150	SCOPE101	LOAD02-102A	LOAD04-102A	LOAD10-102A	LOAD00-102A	LOADD102A	GOTEST102A	LOADD102B
5160	GOTEST102B	LOADD102C	GOTEST102C	LOADD102D	GOTEST102D	SCOPE102	LOAD15-103A	LOAD13-103A
5170	LOAD07-103A	LOAD17-103A	LOADD103A	GOTEST103A	LOADD103B	GOTEST103B	LOADD103C	GOTEST103C
5200	LOADD103D	GOTEST103D	SCOPE103	GOTEST104A	LOADD104B	GOTEST104B	SCOPE104	GOTEST105A
5210	GOBUT105A1	LOADSR105B	GOTEST105B	GOBUT105B1	LOADSR105C	GOTEST105C	LOADSR105D	GOTEST105D
5220	GOTEST105E	SCOPE105	EXPEC374A1	SUBRA372B	SCOPE626	DOWRITE374A1	GETTEM374A1	ZERO374A2
5230	DOWRITE374A2	GETTEM374A2	GOTEST374A2	SCOPE374A	DOWRITE374B1	GETTEM374B1	ZERO374B2	DOWRITE374B2
5240	GETTEM374B2	GOTEST374B2	SCOPE374B	DOWRITE374C1	GETTEM374C1	NEXT010	ZERO374C2	DOWRITE374C2
5250	GETTEM374C2	GOTEST374C2	SCOPE374C	DOWRITE374D1	GETTEM374D1	ZERO374D2	DOWRITE374D2	GETTEM374D2
5260	GOTEST374D2	SCOPE374D	DOWRITE374E1	GETTEM374E1	ZERO374E2	DOWRITE374E2	GETTEM374E2	GOTEST374E2
5270	SCOPE374E	DOWRITE374F1	GETTEM374F1	ZERO374F2	DOWRITE374F2	GETTEM374F2	GOTEST374F2	SCOPE374F
5300	CHECK375B	GOBUT375B	SETIR376A	GOBUT376A	SCOPE376	SCOPE651	LOAD01-610A1	LOAD05-610A1
5310	LOAD06-610A1	PSCC-DC610A1	SETBUSA610A1	DOIT610A1	GETIT610A1	LOAD06-610A2	LOAD01-610A2	PSCC-DC610A2
5320	SETBUSA610A2	DOIT610A2	GETIT610A2	SCOPE610A	LOAD01-610B1	PSCC-DC610B1	SETBUSA610B1	DOIT610B1
5330	GETIT610B1	LOADIP610B2	LOAD07-610B2	LOAD01-610B2	PSCC-DC610B2	SETBUSA610B2	DOIT610B2	GETIT610B2
5340	SCOPE610B	LOAD01-610C1	PSCC-DC610C1	SETBUSA610C1	DOIT610C1	GETIT610C1	LOAD06-610C2	LOAD01-610C2
5350	PSCC-DC610C2	SETBUSA610C2	DOIT610C2	GETIT610C2	SCOPE610C	PSCC-DC610D1	SETBUSA610D1	DOIT610D1
5360	GETIT610D1	PSCC-DC610D2	SETBUSA610D2	DOIT610D2	GETIT610D2	SCOPE610D	GOBUT620A	GOBUT620B
5370	MASK620C	GETJAM620C	CSP1L621A	SETBRK621A	LOOBRK621A	SETRET621A	BUTERRORS	SETTLG621A
5400	LODFLG621A	SETSP621A	GOBUT621B	GOBUT621C	GOBUT621D	EXPEC621E	GETJAM621E	EXPEC621F
5410	GETCUA621F	GOBUT621G	COMP621H	BCIFCH622A	CHGTO622A	GOBUT622A	BCERC622B	GOBUT622B
5420	GETJAM622C	EXPEC623	GETCUA623	SCOPE623	SETBRK624A	LODBRK624A	SETRET624A	ZERDIR624B
5430	EXPEC624C	GETCUA624C	GOBUT624D	SCOPE702	GOBUT701B	GOBUT701D	SCOPE701	EXPEC702A
5440	DOIT376A	SETLED702A	SETKT702A	LOADBA702A	GOTEST702A	SCOPE710	SCOPE711	SCOPE712
5450	GOTEST712B	SCOPE713	EXPEC713B	GOTEST713B	SCOPE721	. . . .	LOADRET722A	. . . .
5460	. . . .	. . . .	LOADRET720A	TEST722A	LOADRET713A	TEST720A	LOADRET712A	TEST713A
5470	LOADRET711A	TEST712A	LOADRET710A	TEST711A	MASK702A	TEST710A	MASK701A	TEST702A
5500	TEST374D2	SCOPE372B	TEST374E2	LOAD372B	TEST374F2	TEST7128	SETSP375A	TEST610
5510	LOAD01-610D1	TEST620A	TEST701D	TEST376A	ZERO374D1	EXPEC374E1	ZERO374F1	TEST375A
5520	LOADD101B	TEST102A	UBRK621A	TEST701B	TEST374C2	TEST004	ZERO374E1	EXPEC374F1
5530	LOAD01-102A	TEST103A	LOADIR610A1	TEST610B1	LOADIR610B1	TEST610C1	ZERO374A1	EXPEC374B1
5540	LOADD104A	TEST105A	LOADSR105A	SETUPCSP17A	LOADIR610C1	TEST610D1	LOAD16-103A	TEST104A
5550	LOAD047A	TEST050A	LOAD050A	TEST101A	TEST102D	CHA372B	ZERO374B1	EXPEC374C1
5560	LOAD046A	TEST047A	LODIP624A	TEST701A	TEST102C	TEST624B	ZERO374C1	EXPEC374D1
5570	BCERC620A	TEST624A	TEST101B	TEST623	TEST102B	TEST622C	TEST103C	TEST622B
5600	TEST105B1	TEST622A	TEST610C2	TEST621F	TEST374A2	TEST621E	TEST374B2	TEST621D
5610	TEST105A1	TEST621C	TEST610B2	TEST621B	TEST376A1	ERROR621A	TEST713B	TEST621A
5620	TEST105E	TEST620C	TEST610A2	TEST620B	TEST610D2	TEST375B	GOTEST951C	TEST610A1
5630	LOAD044A	TEST046A	LOAD044A	TEST045A	LOAD043A	TEST044A	TEST105D	TEST043B

LOCN	----0-----	----1-----	----2-----	----3-----	----4-----	----5-----	----6-----	----7-----
5640	LOAD042A	TEST043A	TEST105C	TEST042E	LOAD041A	TFST042A	TEST104B	TEST041B
5650	LOAD040A	TEST041A	TEST105B	TEST040B	LOAD037A	TEST040A	LOAD036A	TEST037A
5660	LOAD035A	TEST036A	LOAD034A	TEST035A	LOAD033A	TEST034A	LOAD032A	TEST033A
5670	TEST047B	TEST032B	LOAD031A	TEST032A	TEST103B	TEST031B	LOAD030A	TEST031A
5700	LOAD027A	TEST030A	LOAD026A	TEST027A	LOAD025A	TEST026A	LOAD024A	TEST025A
5710	TEST014D	TEST024B	TEST030B	TEST030C	LOAD013A	TEST024A	LOAD022A	TEST023A
5720	LOAD021A	TEST022A	TEST019C	TEST021B	LOAD020A	TEST021A	TEST050B	TEST020C
5730	TEST047C	TEST020B	LOAD017A	TEST020A	TEST030D	TEST017B	LOAD016A	TEST017A
5740	LOAD015A	TEST016A	LOAD014A	TEST015A	LOAD013A	TEST014A	SFDFTOSB	TEST013G
5750	TEST013F	TEST013E	TEST014B	TEST013D	TEST013F	TEST013C	TEST024D	TEST013B
5760	SETIRA	TEST013A	TEST014C	TEST013F	TEST012E	TEST021G	TEST012D	
5770	TEST046B	TEST012C	TEST015B	TEST012B	TEST021C	TEST012A	TEST024C	TEST011

LOCN	----0-----	----1-----	----2-----	----3-----	----4-----	----5-----	----6-----	----7-----
6000	SETBYTEE410	LOADRES361A	LOADBR361A	EXPEC361B	COMP361B	GOBUT361D	GOBUT361E	SETRES162A
6010	SETDDC362A	LOADRES362A	COMP362A	GOBUT362A	EXPEC362B	COMP362B	GOBUT362D	GOBUT362E
6020	SETDDC363A	SHIFT363A	COMP363A	GOBUT363A	GOBUT363B	SETDDC364A	GOBUT364A	SETDDC364B
6030	GOBUT364B	SETRES365A	LOADRES365A	EXPEC365A	COMP365A	GOBUT165B	SETRES366A	LOADRES366A
6040	SHIFT366A	EXPEC366A	COMP366A	GOBUT366B	SETDDC366C	SHIFT366C	EXPEC366C	COMP166C
6050	SETRES367A	LOADRES367A	LOADBR367A	GOBUT367B	SETRES370A	LOADRES370A	SHIFT370A	COMP370A
6060	GOBUT370A	GOBUT370B	SETDDC370C	SHIFT370C	COMP370C	GOBUT370C	GOBUT370D	SETRES371A
6070	LOADRES371A	SHIFT371A	COMP371A	GOBUT371A	GOBUT371B	SCOPE371	SETDC372A	SUBRA372A
6100	PAGE372B	GOBUT373B	SCOPE373	LOADRES374	LOADDC8506A	SETBYTEC410	COMP410A	INTOIR410A
6110	EXPEC410B	COMP410B	GOBUT410B	EXPEC410C	ASIDE410C	COMP410C	GOBUT410C	EXPEC410D
6120	COMP410D	GOBUT410D	EXPEC410E	COMP410E	GOBUT410E	SCOPE410	SETPETCND500	SETPETCHE500
6130	SETPETCHF500	SETPETCHG500	LOADIR500	TESTINMB8P800	COMP500A	GOBUT500A	EXPEC500C	COMP500C
6140	GOBUT500C	COMP500D	GOBUT500D	EXPEC500E	COMP500E	GOBUT500E	SCOPE500F	LOADFP5503A
6150	EXPEC503A	MASK503A	LOADFCC503A	DOFCC503A	GOPUT503A	GOBUT503A	GOBUT503B	GOBUT503C
6160	GOBUT503DA	LOADIR503D	GOBUT503D	EXPEC503E	GOPUT503E	GOBUT503F	GOBUT503G	GOBUT503H
6170	GOBUT503I	GOBUT503J	GOBUT503K	SCOPE503	LOADFP8504A	EXPEC504A	LOADFCC504A	DOFCC504A
6200	GOBUT504A	GOBUT504A	GOBUT504B	GOBUT504C	LOADIR504D	GOBUT504D	EXPEC504E	COPUT504E
6210	GOBUT504F	GOBUT504G	GOBUT504H	GOBUT504I	SCOPE504	EXPEC505A	SETFLAG505A	BUTCLR505A
6220	GOBUT505A	GOBUT505B	CUA372A	GOBUT505C	SCOPE505	SETUCOM504A	PSHIS506A	PSHID506A
6230	PSLO506A	FUNGFP8506A	MFS801	GOBUT506B	GOBUT506C	GOBUT506D	CLEAR506E	GOBUT506E
6240	SCOPE506	PSHI5507A	PSHID507A	PSLO507A	FUDGEP8507A	LOADDC8507A	GOBUT507A	GOBUT507B
6250	GOBUT507C	GOBUT507D	TEST002	GOBUT507E	CIN507F	DBRK624A	SETP8507F	SETSR507F
6260	SETCIN507F	GOBUT507F	SCOPE507	LOADPRIO510A	GOBUT510A	GOBUT510B	GOBUT510C	FLAG510D
6270	FNFLAG510D	SETFLAG510D	GOBUT510D	GOBUT510DA	ZEROD510E	GOBUT510E	ZEROFLAG510E	GOBUT510F
6300	SCOPE510	SCOPE511A	MFS802	MFS803	MFS804	SCOPE511B	SETD624A	SETUP512B
6310	SETP8FLAG512A1	GOTE8T512A1	GOTE8T512A2	SCOPE512A	SETP8FLAG812B1	GOTE8T512B1	GOTE8T512B2	SETIR512C1
6320	SETP8FLAG512C1	GOTE8T512C1	GOTE8T512C2	SCOPE512C	SETP8FLAG812D1	GOTE8T512D1	GOTE8T512D2	SETIR512E1
6330	SETP8FLAG512E1	GOTE8T512E1	GOTE8T512E2	SCOPE512E	SETUP8520A	GOTE8T520A	SETUP820B	GOTE8T520B
6340	SETUP520C	GOTE8T520C	SETUP520D	GOTE8T520D	SETUP520E	GOTE8T520E	SCOPE520	INITD533A
6350	COMP533A	GOBUT533A	INIT533B	INITD533B	COMP533B	GOBUT533B	SCOPE533B	COMP534A
6360	GOBUT534A	INITD534B	COMP534B	GOBUT534B	SCOPE534B	COMP534C	GOBUT534C	INITD534D
6370	COMP534D	GOBUT534D	SCOPE534D	INITD534E	COMP534E	GOBUT534E	BUTERR0R6	NEXTPAT500
6400	LOOCNTR534F	INITD534F	COMP534F	GOBUT534F	COUNTER534G	GOBUT534G	SCOPE534G	EXPEC535A
6410	INITD535A	COMP535A	GOBUT535A	EXPEC535B	INITD535B	COMP535B	GOBUT535B	SCOPE535B
6420	EXPEC536A	INITD536A	COMP536A	GOBUT536A	EXPEC536B	INITD536B	COMP536B	GOBUT536B
6430	SCOPE536F	COMP536C	GOBUT536C	INIT536D	EXPEC536D	INITD536D	COMP536D	GOBUT536D
6440	SCOPE536D	INITD536E	COMP536E	GOBUT536E	LOADBR536F	INITD536F	COMP536F	GOBUT536F
6450	SCOPE536F	ARB=537A	EOP002	SETFLG624A	SETSR624A	EXPEC701C	LOADBA701C	GOTE8T701C
6460	EXPEC702B	SETRT702B	LOADBA702B	GOTE8T702B	GOTE8T710B	EXPEC710C	GOTE8T710C	EXPEC710D
6470	CLEAR710D	GOTE8T710D	GOTE8T710E	EXPEC711B	GOTE8T711B	EXPEC711C	GOTE8T711C	GOTE8T711D
6500	GOBUT712C	EXPEC712D	GOTE8T712D	GOBUT713C	EXPEC713D	GOTE8T7136	GOTE8T720B	. . . .
6510	EOP003	EOP004	EOP007	. . . .	. . . .	. . . .	. . . .	. . . .
6520	TEST720B	TEST713D	TEST713C	TEST712D	TEST712C	ASSERTFOV500	TEST711D	FIXPAT500
6530	TEST711C	ONONZERO410	INITIALIZE01	DZERO410	TEST711B	TEST512E2	SETIR512B1	TEST512D1
6540	LOOP372A	TEST373A	TEST512A1	ERROR010	TEST512A2	MFS8EXPEC8	TEST512B2	MFS8EXPEC1
6550	GOTE8T511A1	TEST511B1	MFS805	MFS806	GOTE8T511B1	SETUP512A	TEST512C1	TEST507D
6560	SETBYTE8410	TEST500	RETURN373A	TEST374	SETIR512A1	TEST512B1	TEST512C2	ERROR624A
6570	SETRES361A	TEST372A	TEST373B	TEST361A	TEST710D	TEST710F	TEST512D2	TEST512E1
6600	TEST510D	TEST503DA	EXPEC507A	TEST510A	TEST510E	TEST506F	SETONES510A	TEST511A
6610	TEST510F	TEST507C	EXPEC506A	TEST507A	LOADFLAG503A	TEST504A	TEST510C	TEST5100A
6620	LOAD505A	TEST506A	LOADFLAG504A	TEST505A	UCON520A	TEST533A	SETIR512D1	TEST520A
6630	TEST5038	TEST003	TEST503C	TEST533B	TEST534B	TEST534D	TEST534F	TEST535B

LOCN	----0-----	----1-----	----2-----	----3-----	----4-----	----5-----	----6-----	----7-----
6640	TEST536R	TEST536D	TEST536F	NEWCTR537A	TEST710B	TEST710C	TEST503X	TEST507F
6650	LOADSP536E	SETEMI7537A	INITD536C	TEST536E	INITD534A	TEST536C	INITD33A	TEST536A
6660	LODCNTR534E	TEST535A	INITD534C	TEST534E	INITD534A	TEST534C	INITD33A	TEST534A
6670	TEST503D	TEST7500	TEST504F	TEST500A	TEST507E	TEST510B	SETFETCHB500	TEST503A
6700	TEST503E	TEST372B	TEST504G	LOAD372A	TEST410B	TEST410E	TEST504H	TEST410C
6710	TEST503F	TEST410A	TEST508B	TEST3710	TEST371A	TEST504I	TEST800C	TEST410D
6720	TEST503G	TEST370C	TEST508C	TEST370B	TEST370A	TEST800C	TEST500E	TEST366A
6730	TEST504B	TEST367A	TEST503H	TEST366C	TEST366D	TEST366B	TEST800D	TEST364A
6740	TEST504C	TEST365B	TEST503I	TEST365A	TEST365C	TEST503J	TEST3620	
6750	TEST504D	TEST363B	TEST534G	TEST363A	TEST363B	TEST362A	TEST1504A	TEST361E
6760	TEST504E	TEST7029	TEST520D	TEST3620	TEST362B	TEST520E	EOP006	TESTINNASP500
6770	TEST701C	TEST361D	TEST520C	TEST0410	VFY001	TEST520E		

LOCN	----0-----	----1-----	----2-----	----3-----	----4-----	----5-----	----6-----	----7-----
7000	SUBR010	INITIALIZE01	SETUPCSP16A	INITIALIZE04	INITIALIZE05	INITIALIZE06	INITIALIZE07	ALUCARRY1
7010	SETUPCSP15A	SETUPCSP14A	SETUPCSP12A	SETUPCSP05A	SETUPCSP00A	SETUPCSP00A	WRITEBF350	ALUCARRY1A
7020	ALUCARRY2	ALUCARRY2A	RESETIR350A	NEXTPAT350	LOADMIA350	AGAINSRD350	RESETIR350B	RESETIR350C
7030	RESTORE02	CSP16XORSPTOIR	RESTORE03	RESTORE04	SETIR373A	CSP16XORFLTTOIR	ZERO8F02DF04	SETSR373A
7040	SFDFTOSRAA	ZERO8F04DF02	ZERO8FDF	ZERO8FDFA	ZERO8F	ZERO8FA	FIRST375A	SFDFTOSRA
7050	SFDFTOSRB	SFDFTO8RC	SFDFTO8RD	SFDFTO8RE	SFDFTO8RF	SFDFTO8RG	SFDFTO8RH	SFDFTOSRI
7060	SETBYTEC410	SECOND375A	BYTE375A	CHECK375A	GOBUT375A	NEXTPAT410	SETBYTED410	LOADUCON506A
7070	BYTEFIRST410	GOBUT506A	DOIT506A	FLAGFP8SEQLOD	TEST511A2	FLAGFP802	FLAGFP803	FLAGFP804
7100	PS8EQLOD	PS8EQLOD02	PS8EQLOD03	PS8EQLOD04	PS8EQLOD05	PS8EQLOD06	TEST511B2	GOTEST511A2
7110	TEST511A3	GOTEST511A3	TEST511A4	GOTEST511A4	XTSRCD8T01	GOTEST511B2	TEST511B3	GOTEST511B3
7120	TEST511B4	GOTEST511B4	SUCBRTEST01	KTSRCD8T02	KTSRCD8T03	KTSRCD8T04	KTSRCD8T04B	KTSRCD8T07
7130	KTSRCD8TOR	INITSP537A	SUCBFTEST02	SUCBRTEST03	SR3=537A	SETRCSA537A	BUTCOUNT=I8=377	CR3=537A
7140	OR3=537A	AR7=537A	BR7=537A	CR7=537A	DR7=537A	AR11=537A	8R11=537A	CR11=537A
7150	DR11=537A	ER3=537A	FR3=537A	GR3=537A	HR3=537A	AL1=537A	SETUPPSCC8DC	GOTEST551B1
7160	GOBUT621H	SETUPPSCC8DC02	PSCCTO8R3=0	PSCCTO8R3=0AA	PSCCTO8R3=0BB	CLEAR624	EXPEC701A	BUSFCN713A
7170	LOADBA701A	GOTEST701A	LOADIR720A	SETADR721A	SETADR720A	SETJAM720A	BUSFCN720A	8DK12
7200	SETJAM721A	BUSFCN721A	MASK762A	C17K12	8DK08	C17K08	SETPAI762C	PRIO6T62A
7210	GETIT762A	EOP005	EOP010	JAMUFP002D	INSERTEOPAEVNO	INSERT02	INSERTREVND	INSERT03
7220	INSER704	DISPLAY	DISP002	DISP003	DISP004	DISP005	DISP006	CLEAR=I=0=A
7230	CLEAR=I=0=B	CLEARI002	CLEARI004	CLEARI005	D[15=12]	D1512A	D[11=06]	D1106A
7240	D[05=00]	DO500A	DZERO	DTOIRB	CLR8ERVICETOD	DATI8ERVICETOD	DATO8ERVICETOD	CJE8ERVIC8TOD
7250	SERVICETOD	PBATOD	FLAGFP8TOD	P8TOD	OBDJAMTOB	CLRJAMTOD	JAMTOD	CUATOD
7260	GETMSKPPOCDAT	MSKPROCDAT	CMPPROCDAT	RESETPROCDAT	LOADFP8CC	LOADFP8CC02	8RINTDIR	DINTOIR
7270	DBUFINTOIR	RESETUCONP	8RINTOIR5	DINTOIR5	DBUFINTOIR8	RESETUCONP5	BUTSR3=0	. . . . .
7300	BUTIR15-12	BUTINSTR5	BUTIR1[FLTPT3=0	NEXT007	BUTIR8=6	BUTMOVDRTIR5=3	BUTINSTR1	BUTBGFP8ERV
7310	BUTD[C]C	BUTCOUTTOOUT7	BUTFP808	BUTDN8NRYTE	BUTCD3=2	BUTSR1=0	BUTBG8ERYL	BUTMMA8KP8[T]
7320	BUTMNXT000	BUTM000	BUTMNXT001	BUTMP8[N]	BUTMNXT002	BUTMFLAG7	BUTMNXT003	BUTMEXFLAG1
7330	BUTMNXT004	BUTMFLTPT8	BUTMNXT005	BUTMEXFLAG2	BUTMNXT006	KTSRCD8T05	BUTMINITJAM	KTSRCD5T06
7340	BUTMNXT007	BUTD-15-ZERO	BUTIR11B	BUTP815	BUTD[C]A	BUT8ERVICE	BUTVECTLOAD	BUTDR6-7L
7350	BUTD[C]B	BUTRADD	BUTOTHERJAM	DINTDIR350	BUTFPPROC	BUTINTRHIGH	BUTIN8TRBRANCH	LOADNI350
7360	BUTPREFETCHJAM	GOFOR410	NEXT721A	EXPEC410A	NEXT720A	NEXT713A	WORD410	BYTESECOND410
7370	SETRE88537A	BUTFPFD	. . . .	GETPROCDAT	VFY006	VFY005	BUTERROR7	. . . .
7400	ZTARGET400	ZTARGET401	ZTARGET402	ZTARGET403	ZTARGET404	ZTARGET405	ZTARGET406	ZTARGET407
7410	ZTARGET410	ZTARGET411	ZTARGET412	ZTARGET413	ZTARGET414	ZTARGET415	ZTARGET416	ZTARGET417
7420	ZTARGET420	ZTARGET421	ZTARGET422	ZTARGET423	ZTARGET424	ZTARGET425	ZTARGET426	ZTARGET427
7430	ZTARGET430	ZTARGET431	ZTARGET432	ZTARGET433	ZTARGET434	ZTARGET435	ZTARGET436	ZTARGET437
7440	ZTARGET440	ZTARGET441	ZTARGET442	ZTARGET443	ZTARGET444	ZTARGET445	ZTARGET446	ZTARGET447
7450	ZTARGET450	ZTARGET451	ZTARGET452	ZTARGET453	ZTARGET454	ZTARGET455	ZTARGET456	ZTARGET457
7460	ZTARGET460	ZTARGET461	ZTARGET462	ZTARGET463	ZTARGET464	ZTARGET465	ZTARGET466	ZTARGET467
7470	ZTARGET470	ZTARGET471	ZTARGET472	ZTARGET473	ZTARGET474	ZTARGET475	ZTARGET476	ZTARGET477
7500	ZTARGET500	ZTARGET501	ZTARGET502	ZTARGET503	ZTARGET504	ZTARGET505	ZTARGET506	ZTARGET507
7510	ZTARGET510	ZTARGET511	ZTARGET512	ZTARGET513	ZTARGET514	ZTARGET515	ZTARGET516	ZTARGET517
7520	ZTARGET520	ZTARGET521	ZTARGET522	ZTARGET523	ZTARGET524	ZTARGET525	ZTARGET526	ZTARGET527
7530	ZTARGET530	ZTARGET531	ZTARGET532	ZTARGET533	ZTARGET534	ZTARGET535	ZTARGET536	ZTARGET537
7540	ZTARGET540	ZTARGET541	ZTARGET542	ZTARGET543	ZTARGET544	ZTARGET545	ZTARGET546	ZTARGET547
7550	ZTARGET550	ZTARGET551	ZTARGET552	ZTARGET553	ZTARGET554	ZTARGET555	ZTARGET556	ZTARGET557
7560	ZTARGET560	ZTARGET561	ZTARGET562	ZTARGET563	ZTARGET564	ZTARGET565	ZTARGET566	ZTARGET567
7570	ZTARGET570	ZTARGET571	ZTARGET572	ZTARGET573	ZTARGET574	ZTARGET575	ZTARGET576	ZTARGET577
7600	ZTARGET600	ZTARGET601	ZTARGET602	ZTARGET603	ZTARGET604	ZTARGET605	ZTARGET606	ZTARGET607
7610	ZTARGET610	ZTARGET611	ZTARGET612	ZTARGET613	ZTARGET614	ZTARGET615	ZTARGET616	ZTARGET617
7620	ZTARGET620	ZTARGET621	ZTARGET622	ZTARGET623	ZTARGET624	ZTARGET625	ZTARGET626	ZTARGET627
7630	ZTARGET630	ZTARGET631	ZTARGET632	ZTARGET633	ZTARGET634	ZTARGET635	ZTARGET636	ZTARGET637

LOCN	----0-----	----1-----	----2-----	----3-----	----4-----	----5-----	----6-----	----7-----
7640	ZTARGET640	ZTARGET641	ZTARGET642	ZTARGET643	ZTARGET644	ZTARGET645	ZTARGET646	ZTARGET647
7650	ZTARGET650	ZTARGET651	ZTARGET652	ZTARGET653	ZTARGET654	ZTARGET655	ZTARGET656	ZTARGET657
7660	ZTARGET660	ZTARGET661	ZTARGET662	ZTARGET663	ZTARGET664	ZTARGET665	ZTARGET666	ZTARGET667
7670	ZTARGET670	ZTARGET671	ZTARGET672	ZTARGET673	ZTARGET674	ZTARGET675	ZTARGET676	ZTARGET677
7700	ZTARGET700	ZTARGET701	ZTARGET702	ZTARGET703	ZTARGET704	ZTARGET705	ZTARGET706	ZTARGET707
7710	ZTARGET710	ZTARGET711	ZTARGET712	ZTARGET713	ZTARGET714	ZTARGET715	ZTARGET716	ZTARGET717
7720	ZTARGET720	ZTARGET721	ZTARGET722	ZTARGET723	ZTARGET724	ZTARGET725	ZTARGET726	ZTARGET727
7730	ZTARGET730	ZTARGET731	ZTARGET732	ZTARGET733	ZTARGET734	ZTARGET735	ZTARGET736	ZTARGET737
7740	ZTARGET740	ZTARGET741	ZTARGET742	ZTARGET743	ZTARGET744	ZTARGET745	ZTARGET746	ZTARGET747
7750	ZTARGET750	ZTARGET751	ZTARGET752	ZTARGET753	ZTARGET754	ZTARGET755	ZTARGET756	ZTARGET757
7760	ZTARGET760	ZTARGET761	ZTARGET762	ZTARGET763	ZTARGET764	ZTARGET765	ZTARGET766	ZTARGET767
7770	ZTARGET770	ZTARGET771	ZTARGET772	ZTARGET773	ZTARGET774	ZTARGET775	ZTARGET776	ZTARGET777

PAGE	---USED--- OCTAL/DEC.	---OPEN--- OCTAL/DEC.
4	773/507	5/ 5
5	774/508	4/ 4
6	772/506	6/ 6
7	775/509	3/ 3
TOTAL	3756/2030	22/ 18