Cromemco

BIART Communication Processor

Instruction Manual

November 1984

023-2024
Rev. A

CROMEMCO, Inc.
P.O. Box 7400
280 Bernardo Avenue
Mountain View, CA 94039

Copyright © 1984
CROMEMCO, Inc.
All Rights Reserved

This manual was produced using a Cromemco System Three computer running under the Cromemco Cromix Operating System. The text was edited with the Cromemco Cromix Screen Editor. The edited text was proofread by the Cromemco SpellMaster Program and formatted by the Cromemco Word Processing System Formatter II. Camera-ready copy was printed on a Cromemco 3355B printer.

The following are registered trademarks of Cromemco, Inc.

C-Net®
Cromemco®
Cromix®
FontMaster®
SlideMaster®
SpellMaster®
System Zero®
System Two®
System Three®
WriteMaster®

The following are trademarks of Cromemco, Inc.

C-10™
CalcMaster™
DiskMaster™
Maximizer™
System One™
TeleMaster™

TABLE OF CONTENTS

TECHNICAL SPECIFICATIONS

INTRODUCTION 1

Chapter 1: SET UP AND INSTALLATION 3

BIART ROM 5
BIART Base I/O Address 5
Connectors and Cables 6
Reset Connector 7
Connector J2 8
Connectors J1 and J3 9
Connector 4 9

Chapter 2: PROGRAMMING INFORMATION 11

Introduction 11
BIART Reset 13
BIART Interrupts 14
Internal BIART Interrupts 14
Host Interrupts 15
BIART, Host Communications 15
Host Commands to the BIART 15
BIART Status to the Host 16
Host Data to the BIART 16
BIART Data to the Host 17

LIST OF APPENDICES

Appendix A: BIART Register Descriptions 19

Appendix B: RS-232C, RS-422, and RS-423 Interfaces 37

Appendix C: Six-Bit Transcode 41

Appendix D: ASCII Character Code 43

Appendix E: EBCDIC Character Code 45

Appendix F: Parts List 47

SCHEMATIC 51

INDEX 55

LIMITED WARRANTY

LIST OF ILLUSTRATIONS

Figure 1-1:	The BIART Board	4
Figure 1-2:	BIART Switch Settings	8
Figure 1-3:	Installing the Priority Cable	8
Figure 1-4:	BIART RS-232C Interface	9
Figure B-1:	RS-232C Interface Wiring	38
Figure B-2:	RS-422 Interface Wiring	38
Figure B-3:	RS-423 Interface Wiring	39

LIST OF TABLES

Table 1-1:	Host, BIART Communication Ports	6
Table 1-2:	BIART Connectors and Pin-Outs	7
Table 2-1:	BIART Register Summary	12
Table 2-2:	Registers after BIART Reset	14
Table B-1:	Serial Interface Specifications	37

BIART TECHNICAL SPECIFICATIONS

Processor:	Z80B, 5.5 MHz clock
Memory:	Software selectable configurations: - 64 Kbytes DRAM, or - 16 Kbytes ROM, 32 Kbytes DRAM
Parallel Channels:	One 8-bit input channel; one Centronics printer compatible 8-bit output channel
Parallel Interface:	LSTTL levels
Serial Channels:	2 independent channels
Serial Protocols:	- Asynchronous Byte - Synchronous Byte (IBM BiSync) - Synchronous Bit (SDLC/HDLC)
Serial Bit Coding:	- NRZ - NRZI - FM0 (bi-phase space) - FM1 (bi-phase mark)
Serial Interface:	RS-232C, RS-422, and RS-423 (with circuits TxD, RxD, TxC, RxC, CTS, RTS, DCD, and DTR)
Channel Data Rates:	Asynchronous, 30.5 to 19,200 baud with x16 clock multiplier, baud rate varied by dividing a local, crystal controlled 5.5 MHz clock by a programmable 16-bit scale factor; synchronous, 0 to 1 Mbits/sec with x1 clock multiplier
Character Length:	Transmitter, 1 to 8 bits; receiver, 5 to 8 bits
Error Detection:	Parity, CRC-16 or CRC-CCITT generation and checking, receiver framing and overrun, transmitter underrun, break generation and detection
Interrupts:	Software controlled, vectored maskable interrupts from the BIART directed to the host processor; vectored maskable interrupts from Z-SCC channel, the parallel port, and from host I/O with the BIART registers directed to the BIART Z80B
LSI Device Types:	1 - Z80B CPU (Central Processing Unit) 1 - Z8530A SCC (Serial Communications Controller) 8 - 4164 64Kx1 bit DRAM (dynamic RAM) 1 - 27128 16Kx8 bit ROM (user supplied) 1 - Custom CMOS Circuit

Host Interface:	Two bi-directional S-100/IEEE-696 bus I/O ports; S-100/IEEE-696 maskable vectored interrupts
S-100/IEEE-696 Power:	+ 8 VDC @ 1.5 A +18 VDC @ 250 mA -18 VDC @ 250 mA
Operating Environment:	0 - 55 degrees Celsius

INTRODUCTION

This manual provides installation, operating, and programming instructions for Cromemco's **BIART** communications processor board. The BIART board is a second generation, co-processing subsystem which interfaces two serial channels and a bi-directional parallel port to a host S-100/IEEE-696 bus. A typical BIART application might consist of interfacing two computer terminals or modems and a Centronics-compatible parallel printer to the host system. Unlike earlier serial interface boards, which merely formatted and exchanged individual data characters, the BIART features a sophisticated dual channel Z-SCC serial communications circuit, **plus** an independent Z80B processor with 64 Kbytes of memory. Because the BIART performs all protocol and error detection/recovery functions, buffers large amounts of serial data, and passes only pre-processed data over the host bus, the host CPU is relieved of many I/O functions, and system throughput is dramatically increased.

The BIART is a versatile serial subsystem, compatible with both Z80 Cromix version 11.27 and higher, and 68000 Cromix version 20.63 and higher. Under program control, it can switch its internal memory configuration from 16 Kbytes of ROM and 32 Kbytes of RAM to a full 64 Kbytes of RAM. Thus the board can include a ROM bootstrap program which loads an application program, and then switches to 64 Kbytes of RAM for maximum buffer space. The two serial channels can operate independently of one another in asynchronous, byte-synchronous (IBM BiSync), or bit-synchronous (SDLC/HDLC) mode, and each channel can encode and decode NRZ, NRZI, FM0, or FM1 data.

Chapter 1 describes how to install the BIART board in an S-100 bus system. Chapter topics include cables, switch settings, and connector pin assignments.

Chapter 2 presents BIART programming information. This chapter assumes the reader is familiar with programming in general, and with Z80 Assembly Language in particular (see Reference 1). Most of the information in this chapter relates to several BIART registers through which the Z80B processor manages all board functions. Several of these registers are mapped to access internal Z-SCC registers; detailed programming information for this device is contained in Reference 2 below.

Positive logic is assumed throughout the manual. That is, logic 0 is associated with a more negative voltage (near 0 VDC), and logic 1 with a more positive voltage (near +4 VDC). **Reset** means logic 0, and **set** means logic 1, as these terms apply to bit states. The * notation appearing after a signal name means that the signal is active in the logic 0 state (e.g., signal RESET* is active when at logic 0 and inactive when at logic 1).

Reference 1 Zilog, Inc., Z80 Assembly Language Programming Manual, 1977

Reference 2 Zilog, Inc., Z8030/Z8530 Serial Communication Controller Technical Manual, April 1982

Chapter 1

SET UP AND INSTALLATION

The BIART board is set up by inserting a ROM device in socket IC11, cutting a jumper strap (B) if the ROM has an access time of 150 nSec or less, and setting switch SW-1 to select an S-100 base port address. After the board is set up, insert it in an empty S-100 bus slot with power off. Connect cables between the serial DCE equipment, the parallel printer, and the BIART, and place the BIART in the S-100 bus interrupt priority chain. All of these steps are explained, in order, in the following sections.

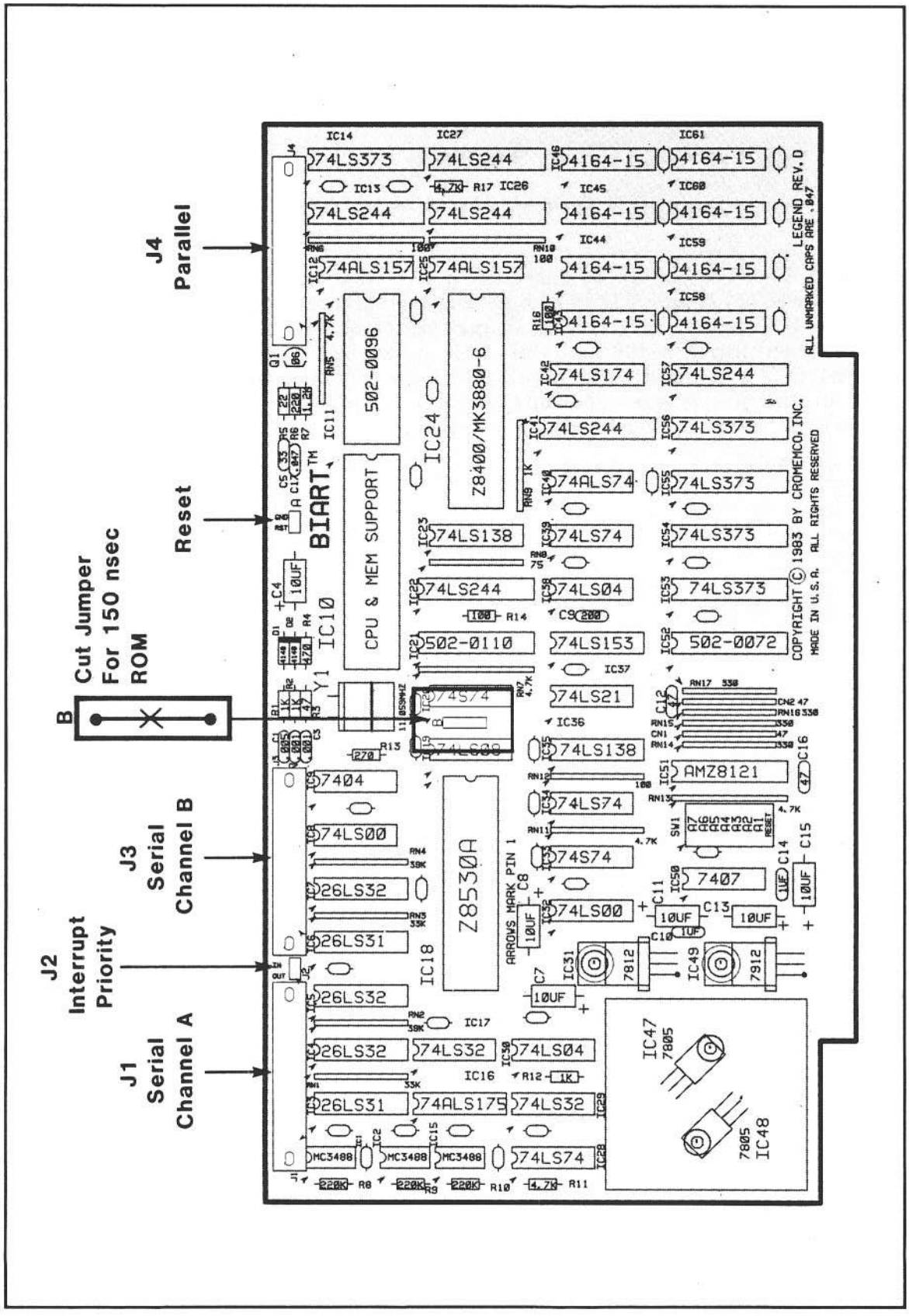


Figure 1-1: THE BIART BOARD

BIART ROM

Whenever the BIART board is reset (many events can reset the board -- see the section, "BIART RESET" in chapter 2).

1. The BIART memory configuration is switched to 16 Kbytes of ROM from 0000h - 3FFFh, and 32 Kbytes of RAM from 4000h - BFFFh, and

2. The Z80B processor is reset. This means, among other things, that it starts executing whatever program code starts at memory address 0000h.

Thus, the ROM firmware must be in place to start up the BIART board after a reset.

The BIART board supports the 16K x 8-bit 27128 ROM, or any pin compatible equivalent. The access speed of the installed ROM is 450 nanoseconds. If you want to use a faster ROM (150 nSec or less), cut jumper "B" between IC19 and IC20 (see figure 1-1).

BIART BASE I/O ADDRESS

The host processor and the BIART communicate through two bi-directional S-100 bus I/O ports. The communication can be either polled or interrupt-driven. BIART switch SW-1 defines the base I/O port number, **Bbase**, which the host

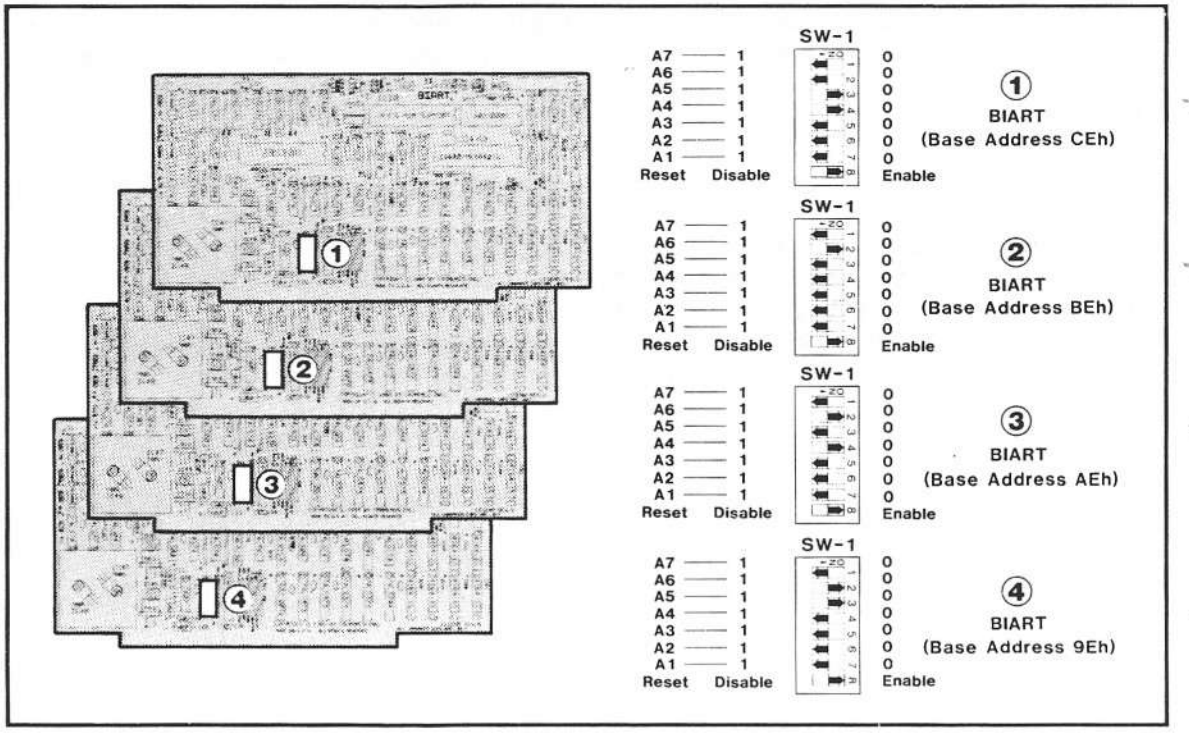


Figure 1-2: BIART SWITCH SETTINGS

uses to access these two ports. Table 1-1 shows the port number assignments. Notice that port addresses may be changed relative to the host processor with switch SW-1, but are fixed relative to the BIART's Z80B processor.

Table 1-1: HOST, BIART COMMUNICATION PORTS

Register Name	Host	BIART
Commands from Host	OUT Bbase+00h	-> IN 00h
Status to Host	IN Bbase+00h	<- OUT 00h
Data from Host	OUT Bbase+01h	-> IN 01h
Data to Host	IN Bbase+01h	<- OUT 01h

Figure 1-2 shows the recommended switch settings for each of the four BIART boards supported by the Cromix Operating System. With the settings shown for BIART #1, **Bbase** = CEh. Thus the host processor addresses output port CFh to send a data byte to register **Data From Host**, and the BIART's Z80B processor reads the same byte from input port 01h.

CONNECTORS AND CABLES

There are five BIART connectors, Reset and J1 through J4, located along the top of the board (see figure 1-1). The Reset connector resets the BIART board, J2 connects the BIART in the S-100 interrupt priority chain, J1 connects to serial channel A, J3 connects to serial channel B, and J4 connects to a Centronics-style parallel printer or other parallel I/O device.

With system power off, route three 25-conductor ribbon cable assemblies (Cromemco part number 519-0017, 62 cm long, or 519-0008, 110 cm long) through the computer system housing for BIART connectors J1, J3, and J4 (use fewer cables if not all connectors are used). Secure the DB-25S socket end of each cable to the connector cutouts on the system rear panel. Clearly mark each ribbon cable with the connector number from the rear panel. Route the cables so that the 26-pin female connectors comfortably reach connectors J1, J3, and J4 when the BIART board is installed. Install the BIART board in an empty S-100 bus slot. Do **not** turn on system power until all cabling is installed.

Table 1-2 defines the pin-outs of all BIART connectors. A dashed table entry (---) denotes no connection (a floating pin). GROUND entries denote a direct connection to S-100 bus GROUND at 0 VDC.

Table 1-2: BIART CONNECTORS AND PIN-OUTS

PIN	CONNECTOR	
	RESET	J2
<1>	RESET*	PRIORITY IN*
<2>	GROUND	PRIORITY OUT*

PIN	CONNECTOR		
	J1	J3	J4
<1>	---	---	---
<2>	RxD-A	RxD-B	INPUT STROBE*
<3>	TxD-A	TxD-B	+5 VDC
<4>	RTS-A	RTS-B	BIT 6 IN
<5>	CTS-A	CTS-B	BIT 4 IN
<6>	DCD-A	DCD-B	BIT 2 IN
<7>	GROUND	GROUND	BIT 0 IN
<8>	DCD-A	DCD-B	DISABLE*
<9>	TxC-A*	TxC-B*	OUTPUT STROBE*
<10>	---	---	BIT 6 OUT
<11>	TxD-A2*	TxD-B2*	BIT 4 OUT
<12>	DCD-A*	DCD-B*	BIT 2 OUT
<13>	CTS-A*	CTS-B*	BIT 0 OUT
<14>	TxD-A2	TxD-B2	GROUND
<15>	TxC-A	TxC-B	SENSE*
<16>	RxD-A*	RxD-B*	BIT 7 IN
<17>	RxC-A	RxC-B	BIT 5 IN
<18>	RxC-A*	RxC-B*	BIT 3 IN
<19>	RTS-A2	RTS-B2	BIT 1 IN
<20>	DTR-A	DTR-B	NMI*
<21>	---	---	WAIT*
<22>	---	---	BIT 7 OUT
<23>	DTR-A2	DTR-B2	BIT 5 OUT
<24>	DTR-A2*	DTR-B2*	BIT 3 OUT
<25>	RTS-A2*	RTS-B2*	BIT 1 OUT

RESET CONNECTOR

The pins of this connector can be wired to a normally open pushbutton switch, or pin RESET* can be wired to any output capable of sinking 2 mA to ground (one bit of an output port, for example). Forcing pin RESET* to logic 0 resets the entire BIART board.

CONNECTOR J2

Connecting the priority interrupt cable (part number 519-0029) to J2 on the BIART board determines the order in which the host processor services BIART interrupt requests when they conflict with those from other boards in the system.

Attach the first connector on the priority interrupt cable to J1 on the 64FDC/16FDC board by aligning the blue dot on the connector with the blue dot on the plug (see figure 1-2). Aligning the yellow dots on the remaining plugs and connectors, attach the second connector to the next highest priority board, the third connector to the next highest, and so on. The suggested order of board priorities is: 64FDC/16FDC, STDC, OCTART, TU-ART, IOP, BIART, MAXIMIZER, GPIB, CTI, PRI, and SCC. The order of the boards between the 64FDC/16FDC and the SCC is not critical, as long as no boards or connectors are skipped. Any unused connectors must be at the end of the cable farthest from the 64FDC/16FDC.

If your priority connectors are not color coded, install the cable so that the OUT pin of the higher priority board is linked to the IN pin of the next highest board (the IN pin is on the right on all boards except 64FDC/16FDC, STDC, and CTI). The IN pin of the highest priority board (the 64FDC/16FDC) is not connected, nor is the OUT pin of the lowest priority board (the last board in the chain).

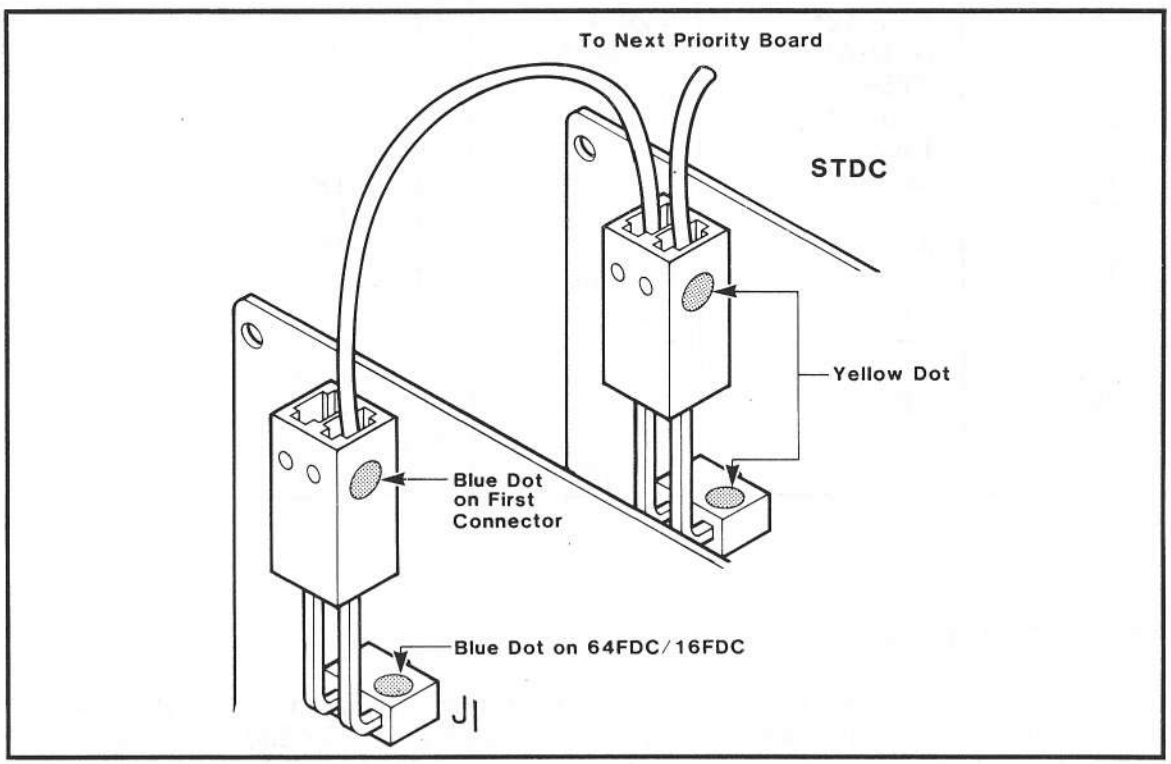


Figure 1-3: INSTALLING THE PRIORITY CABLE

CONNECTORS J1 AND J3

Connectors J1 and J3 interface serial (RS-232C, RS-422 or RS-423) DCE equipment to the BIART board. Connector J1 attaches to serial channel A, and J3 attaches to serial channel B. Figure 1-3 shows the relationships among incoming and outgoing voltage levels and logic states.

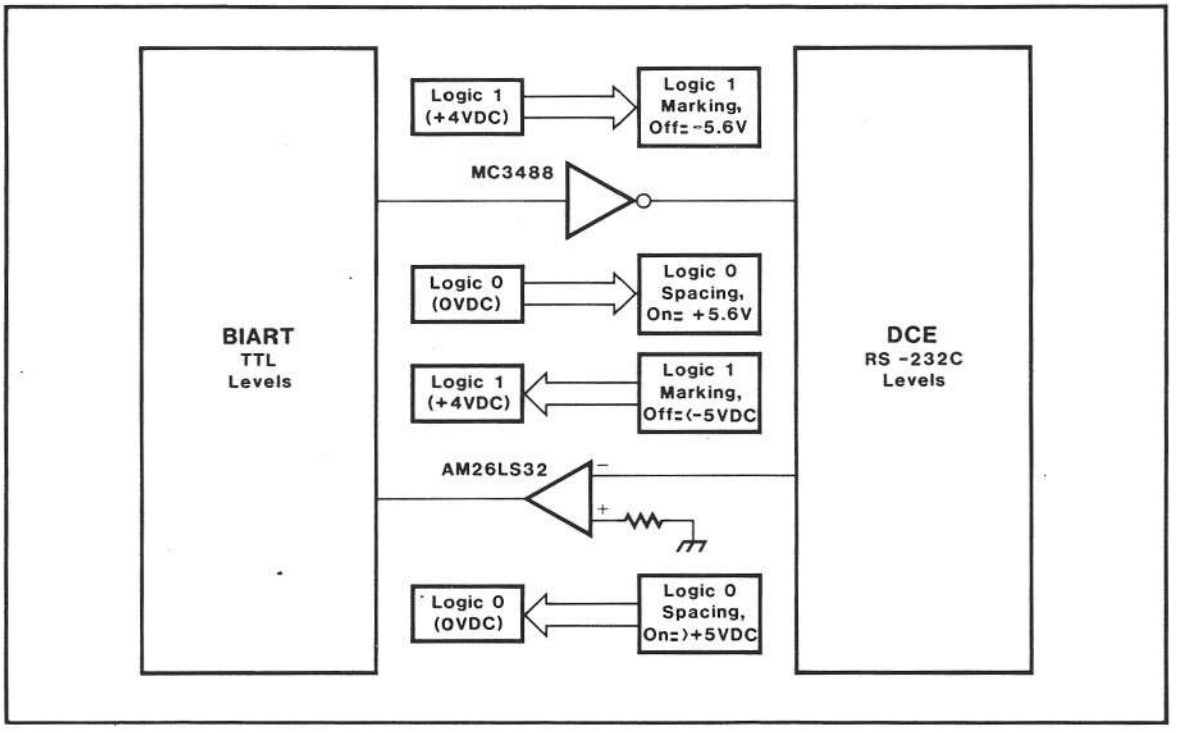


Figure 1-4: BIART RS-232C INTERFACE

Choose two ribbon cables attached to the system housing rear panel (installed above), align the red cable stripe of each cable with the BIART board legend arrowheads, and attach the 26-pin female connectors to BIART connectors J2 and J3. Attach the terminals or other DCE equipment at the system back panel DB-25S sockets. The BIART, supports the RS-232C interface without modification. For the RS-422 or RS-423 interface, see appendix B.

CONNECTOR J4

Connector J4 interfaces either a Centronics-style parallel printer, or a general purpose TTL level parallel I/O device to the BIART board. Attach the 26-pin female connector to BIART connector J4, aligning the red cable stripe with the board legend arrowhead. Attach the parallel printer or other parallel I/O device at the system back panel DB-25S socket.

Chapter 2

PROGRAMMING INFORMATION

INTRODUCTION

The BIART is a co-processing I/O management subsystem for the S-100 bus host processor. Application software for this architecture is normally structured as follows:

1. The host to BIART link is made as short, fast, and simple as possible. This means the routine running in host memory should view the BIART as a simple data source or sink, with a minimum amount of status and command information required to carry out a data exchange. Normally, only pre-processed data should travel on the S-100 bus on a vectored interrupt-driven basis.

2. The routine running in BIART exchanges commands, status, and data with the host processor. To off-load the host processor, the BIART routine should also assume responsibility for connecting and disconnecting serial links, managing serial protocols, formatting or processing the serial data (such as data encryption and decryption), managing the parallel printer, monitoring data integrity, and attempting all possible error recovery procedures for the host processor.

The BIART program store is 16 Kbytes of ROM from 0000h - 3FFFh, and 32 Kbytes of RAM from 4000h - BFFFh, immediately after a BIART reset (see the next section). Under program control, the memory configuration can later be switched to 64 Kbytes (0000h - FFFFh).

The BIART Z80B processor manages all board functions through several I/O mapped registers, which are listed in table 2-1. The BIART Z80B accesses all but four of these registers by executing either a single input instruction (e.g., IN A, (51h) to read register **Channel B Receive Data**), or a single output instruction (e.g., OUT A, (51h) to write to register **Channel B Transmit Data**) to the target register's port address. The port number and direction (from the BIART Z80B's point of view) for each register is shown in the table. Four of the registers (**Channel A Status, Channel B Status, Channel A Commands, Channel B Commands**), require either one or two I/O instructions, depending upon which internal Z-SCC register is accessed. **Detailed descriptions for these registers appear in appendix A.**

Table 2-1: BIART REGISTER SUMMARY

BIART PORT	HOST PORT	REGISTER FUNCTION
IN 00h	OUT Bbase+00h	Commands from Host
OUT 00h	IN Bbase+00h	Status to Host
IN 01h	OUT Bbase+01h	Data from Host
OUT 01h	IN Bbase+01h	Data to Host
IN 02h	---	BIART Flags
OUT 02h	---	BIART Control
OUT 03h	---	Interrupt Vector To Host
IN 1Xh	---	Parallel Input
OUT 1Xh	---	Parallel Output
OUT 4Xh	---	Configure Memory
IN 50h	---	Channel B Status
OUT 50h	---	Channel B Command
IN 51h	---	Channel B Receive Data
OUT 51h	---	Channel B Transmit Data
IN 52h	---	Channel A Status
OUT 52h	---	Channel A Command
IN 53h	---	Channel A Receive Data
OUT 53h	---	Channel A Transmit Data

The last eight BIART registers are mapped to access the internal Z-SCC registers which manage serial channels A and B. In this group, the command and status registers each access multiple Z-SCC registers. Through register **Channel B Command**, for example, the BIART Z80B can write to 16 channel B internal registers, WR0 through WR15. See Reference 2 for detailed Z-SCC programming information.

The next two registers in the table pass 8-bit parallel data to and from BIART connector J4. Bit D0 of the next register, **Configure Memory**, controls the BIART on-board memory configuration.

The next four registers are accessible to both the host processor and the BIART Z80B processor. They are used to exchange commands, status, and data between the host and BIART processors. Notice that a host input port is a BIART output port, and vice versa. The host (S-100 bus) port address of these four registers can be changed with BIART switch SW-1 (see Section 1.2), but to the BIART's Z80B, the port numbers are fixed.

The first three registers in the table are used for monitoring BIART flags (for example, to determine if a command byte from the host processor is available), to control the BIART interrupt circuitry, and to issue interrupt vectors to the host processor, respectively.

BIART RESET

Several events can reset the BIART board:

1. Applying power to the BIART board. BIART Power On Clear (POC) circuitry generates a momentary active low pulse on line RESET* (IC14 pin 8) whenever S-100 bus lines 1 and 51 go from 0 VDC to +8 VDC.

2. An S-100 bus reset. This occurs whenever S-100 bus line 75, pRESET*, pulses active low. The reset condition persists as long as pRESET* is held low.

3. Forcing the RESET* pin of BIART connector J1 active low. This can be done by either shorting pins RESET* and GROUND of J1 with a normally open pushbutton switch, or by driving pin RESET* of J1 active low with any output capable of sinking 2 mA @ +0.4 VDC or less. Pin RESET* must go inactive high again to remove the reset condition.

4. A software reset. If BIART switch SW-1, section 8 is ON (see figure 1-1), then the host can reset the BIART board by sending the following six bytes to register **Commands From Host:** 7Eh, 55h, 0Fh, 70h, 2Ah, and 7Eh. These bytes cause a momentary reset condition. If this command sequence is altered in any way, no BIART reset occurs. To disable the software feature, set switch 8 of SW-1 OFF. Only D0 through D6 are used. D7 is not used for software reset.

All four of these events force line RESET* on the internal BIART control bus to go active low. This event is called a **BIART reset** throughout this manual. A BIART reset initializes the BIART board as follows:

1. The BIART memory configuration is unconditionally switched to 16 Kbytes of RAM from 0000h - 3FFFh, and 32 Kbytes of RAM from 4000h - BFFFh.

2. The BIART Z80B is reset. This means Z80B maskable interrupts are disabled, the I-register is initialized to 00h, the R-register is initialized to 00h, interrupt mode IM0 is selected, and the Z80B automatically starts executing program code at 0000h as soon as the reset condition is removed.

3. Selected BIART register bits are forced either set (logic 1) or reset (logic 0), as shown in table 2-2. All other BIART register bits are unaffected.

Table 2-2: REGISTERS AFTER BIART RESET

Bit #	OUT 4Xh	OUT 50h	IN 52h	OUT 52h
D7	---	1	1	---
D6	---	0	0	---
D5	---	0	---	---
D4	---	0	---	---
D3	---	0	---	---
D2	---	0	---	0
D1	---	1	1	---
D0	1	RESET*	0	0

BIART INTERRUPTS

For maximum system throughput, all I/O between the host and BIART, and between the BIART and its peripherals, should be interrupt-driven. There are two categories of BIART interrupts: internal BIART interrupts and host interrupts. The following paragraphs discuss each category.

INTERNAL BIART INTERRUPTS

The BIART's Z80B processor can be interrupted by a variety of sources on the board itself. The maskable interrupts issued by these sources, and directed to the BIART's Z80B (in contrast to those directed to the host processor) are collectively termed **internal BIART interrupts.** The internal BIART interrupt sources, from highest to lowest interrupt priority, are:

1. Z-SCC interrupt requests. The Z-SCC can be programmed to issue interrupt requests to the BIART Z80B in a variety of channel conditions, and to supply a variable interrupt vector which pinpoints the channel and interrupt condition within the channel when the BIART Z80B acknowledges the request. These conditions include Tx Buffer Empty, Rx Character Available, External or Status Conditions, and Special Receive Conditions. Z-SCC interrupts have the highest internal BIART interrupt priority. See Reference 2 for more information on Z-SCC interrupts.

2. Parallel device interrupt requests. A parallel device attached to BIART connector J4 can issue a maskable interrupt request to the BIART Z80B by forcing pin 15, SENSE*, active low. When the Z80B acknowledges the request, on-board circuitry automatically supplies interrupt vector **FAh** (**F8h** when host I/O and parallel device requests coincide) to the processor, and removes the interrupt request. If the parallel device is a printer, the interrupt request would normally signify that the printer is ready to accept a new character, and the interrupt service routine would send it one, if available. Parallel device interrupt requests have lower priority than Z-SCC requests, but the same priority as host I/O requests. If bit **Enable Parallel Port Interrupt** is reset, this feature is disabled. (See **BIART Control** register.)

3. Host I/O with the BIART registers. If bit **Enable BIART Interrupts** of register **BIART Control** is set, then a maskable interrupt request is automatically issued to the BIART Z80B whenever the host processor either reads from, or writes to, the four BIART registers to which it has access (**Data To Host, Data From Host, Status To Host,** and **Commands From Host).** When the Z80B acknowledges the interrupt request, on-board circuitry automatically supplies interrupt vector **FCh** (**F8h** when host I/O and parallel device requests coincide) to the processor. This feature provides a convenient means to alert the BIART Z80B that the host has either read data or status, or has written data or a command to the BIART. The interrupt service routine for this interrupt source should read register **BIART Flags** to determine which of these events has occurred, and take appropriate action. If bit **Enable BIART Interrupts** is reset, then this feature is disabled (see **BIART Control** register.) Host I/O interrupt requests and parallel device requests share the lowest internal BIART interrupt priority.

HOST INTERRUPTS

When the BIART has status or receive data available it alerts the host processor by sending a maskable interrupt request. The BIART does this by setting bit **Enable Host Interrupts** in register **BIART Control,** and then writing an interrupt vector to register **Interrupt Vector To Host.** When the vector is written to this register, on-board circuitry automatically drives S-100 bus line pINT* active low, which relays the interrupt request to the host processor. When the host acknowledges the request, the contents of register **Interrupt Vector To Host** are automatically placed on the S-100 bus Data In lines to vector the host processor to an appropriate service routine, and the interrupt request is removed. If bit **Enable Host Interrupts** is reset, then writing to register **Interrupt Vector To Host** is a null operation.

BIART, HOST COMMUNICATIONS

The BIART Z80B and the host processor communicate through two bi-directional S-100 bus ports, normally using interrupt-driven I/O. This communication falls into four categories: host commands to the BIART, BIART status to the host, host data to the BIART, and BIART data to the host.

HOST COMMANDS TO THE BIART

The host sends an 8-bit command to the BIART by first polling bit **Command From Host Empty** of register **Status To Host.** If this bit is set, then the host can output a command byte to register **Commands From Host.** If status bit **Commands From Host Empty** is reset, the BIART has not read the previous command, and the host processor should hold off writing a new one; otherwise, the previous command byte will be overwritten.

The BIART determines that a new command byte from the host is available in register **Commands From Host** by polling bit **Command From Host Available** of register **BIART Flags.** This bit is set whenever the host writes a command to register **Commands From Host.** After reading a command from this register, the BIART must write a logic 1 to bit **Command From Host Empty** of register **Status To Host; it is not** automatically reset when the BIART reads a command. This action both sets status bit **Command From Host Empty** and resets flag bit **Command From Host Available.**

The meaning of individual command bits is completely defined by the software, with the sole exception of the command sequence used to reset the BIART software.

BIART STATUS TO THE HOST

The BIART sends four status bits to the host by first polling bit **Status To Host Empty** of register **BIART Flags.** If this bit is set, the host can output a status byte to register **Status To Host.** Only bits D5 through D2 (**Status 5** through **Status 2,** respectively) are passed to the host; except for bit **Command From Host Empty** (see above), all other bit states written are irrelevant (they are hardware controlled). If status bit **Status To Host Empty** is reset, the host has not read the previous status word, and the BIART should hold off writing a new one; otherwise, the previous status word will be overwritten.

There is no fixed status bit available to alert the host that a new status word from the BIART is available. Normally, one of the four status bits is set aside for this handshaking function. (Although Bit D1 can be set or reset by the same instruction that sets/resets bits D5 to D2, it cannot be used as a handshaking signal because of timing differences.) Again, the meaning of individual status bits is completely defined by the software. When the host reads register **Status To Host,** flag bit **Status To Host Empty** is automatically set for the next status exchange.

HOST DATA TO THE BIART

The host sends 8-bits of data to the BIART for serial or parallel transmission by first polling bit **Data From Host Empty.** If this bit is set, then the host can output a data byte to register **Data From Host.** If status bit **Data From Host Empty** is reset, the BIART has not read the previous data byte, and the host processor should hold off writing a new one; otherwise, the previous data byte will be overwritten.

The BIART determines that a new data byte from the host is available in register **Data From Host** by polling bit **Data From Host Available** of register **BIART Flags.** A set bit implies a data byte is available; a reset bit implies the opposite. Reading the data byte from register **Data From Host** sets status bit **Data From Host Empty** for the next data transfer.

BIART DATA TO THE HOST

The BIART sends 8 bits of received data to the host by first polling bit **Data To Host Empty** of register **BIART Flags.** If this bit is set, then the BIART can output a data byte to register **Data To Host.** If status bit **Data To Host Empty** is reset, the host has not read the previous data byte, and the BIART should hold off writing a new one; otherwise, the previous data byte will be overwritten.

The host determines that a new data byte from the BIART is available in register **Data to Host** by polling bit **Data to Host Available** of register **Status to Host.** A set bit implies a data byte is available; a reset bit implies the opposite. Reading the data byte from register **Data to Host** sets flag bit **Data to Host Empty** for the next data transfer.

Appendix A

BIART REGISTER DESCRIPTIONS

Register
COMMANDS FROM HOST

BIART: **IN 00h**
Host: **OUT Bbase+00h**

D7	Command Bit 7 (MSB)
D6	Command Bit 6
D5	Command Bit 5
D4	Command Bit 4
D3	Command Bit 3
D2	Command Bit 2
D1	Command Bit 1
D0	Command Bit 0 (LSB)

The host passes commands to the BIART through this port. The bits are user defined, so their interpretations depend on the BIART/Host software. (The command sequence 7Eh, 55h, 0Fh, 70h, 2Ah, 7Eh is reserved for software reset -- see the section "BIART RESET" in chapter 2 for details.) These bits are not affected by a BIART reset, and should be assumed to be random until a command is written to this port for the first time. Bit **Command From Host Available** of register **BIART Flags** is set to alert the BIART that a command from the host is available in this register. A host write to this port may be programmed to generate an internal BIART interrupt by setting bit **Enable BIART Interrupts** of register **BIART Control.** Only D0 through D6 are used - D7 is not used for software reset.

Register
STATUS TO HOST

BIART: **OUT 00h**
Host: **IN Bbase+00h**

D7	Data From Host Empty
D6	Data To Host Available
D5	Status 5
D4	Status 4
D3	Status 3
D2	Status 2
D1	Command From Host Empty
D0	BIART Reset*

This register supplies the host processor with handshaking lines for exchanging data with the BIART, and also allows the BIART to supply four software-driven status bits to the host. A host read from this port may be programmed to generate an internal BIART interrupt by setting bit **Enable BIART Interrupts** of register **BIART Control.**

D7 Data From Host Empty

This bit is hardware controlled; data output by the BIART to this bit position is ignored. This bit is reset immediately after the host processor writes a data byte to register **Data From Host.** This bit is set (signifying that the host may write another data byte to the BIART) by a BIART reset or after the BIART reads register **Data From Host.**

D6 Data To Host Available

This bit is hardware controlled; data output by the BIART to this bit position is ignored. This bit is set (signifying that a BIART written data byte is available for host reading) when the BIART writes a data byte to register **Data To Host.** This bit is reset after the host processor reads register **Data To Host,** or by a BIART reset.

D5 Status 5

This bit is defined by the BIART software, and is reset by a BIART reset.

D4 Status 4

This bit is defined by the BIART software, and is reset by a BIART reset.

D3 Status 3

This bit is defined by the BIART software, and is reset by a BIART reset.

D2 Status 2

This bit is defined by the BIART software, and is reset by a BIART reset.

D1 Command From Host Empty

This bit is set (signifying that the host may write a new command byte to the BIART) when the BIART outputs a logic 1 to this bit or by a BIART reset. This bit is reset as the host processor writes a byte to register **Command From Host.** The BIART must set this bit for the host under program control since the bit is **not** automatically set when the BIART reads register **Command From Host.** Outputting a logic 0 will reset this bit.

D0 BIART Reset*

This bit monitors the state of the line RESET* on the internal BIART control bus. If this bit is 0 when read by the host, the RESET* line is active low (the BIART is in the middle of a reset operation). If this bit is 1 when read by the host, the RESET* line is inactive high (there is no reset operation in progress). Outputs to this bit position are null operations.

Register
DATA FROM HOST

BIART: **IN 01h**
Host: **OUT Bbase+01h**

D7	Data Bit 7 (MSB)
D6	Data Bit 6
D5	Data Bit 5
D4	Data Bit 4
D3	Data Bit 3
D2	Data Bit 2
D1	Data Bit 1
D0	Data Bit 0 (LSB)

The host passes eight bits of parallel data to the BIART through this register. These bits are not affected by a BIART reset, and should be assumed to be random until data is written to this register for the first time. Bit **Data From Host Available** of register **BIART Flags** is set to alert the BIART that a data byte from the host is available in this register. When the BIART reads register **Data From Host,** bit **Data From Host Empty** of register **Status To Host** is set to alert the host that it may output a new data byte. If the host writes data to this port before the BIART has read the previous byte, the new data overwrites the old. If the BIART reads register **Input Data** before new data is available, the previous byte is re-read. A host write to this port may be programmed to generate an internal BIART interrupt by setting bit **Enable BIART Interrupts** of register **BIART Control.**

Register
DATA TO HOST

BIART: **OUT 01h**
Host: **IN Bbase+01h**

D7	Data Bit 7 (MSB)
D6	Data Bit 6
D5	Data Bit 5
D4	Data Bit 4
D3	Data Bit 3
D2	Data Bit 2
D1	Data Bit 1
D0	Data Bit 0 (LSB)

The BIART passes eight bits of parallel data to the host processor through this register. These bits are unaffected by a BIART reset, and should be assumed to be random until data is written to this port for the first time. **Status To Host** bit **Data To Host Available** is set to alert the host that a data byte from the BIART is available in this register. When the host reads register **Data To Host,** bit **Data To Host Empty** of register **BIART Flags** is set to alert the BIART that it may output a new data byte. If the BIART writes data to this port before the host processor has read the previous byte, the new data overwrites the old. If the host reads register **Data To Host** before new data is available, the previous byte is re-read. A host read from this port may be programmed to generate an internal BIART interrupt by setting bit **Enable BIART Interrupts** of register **BIART Control.** The BIART may issue a host interrupt request (after writing data to the host) by writing a vector byte to register **Interrupt Vector To Host** while bit **Enable Host Interrupt Request** of register **BIART Control** is set.

Register
BIART FLAGS

BIART: **IN 02h**
Host: **No Access**

D7	Data To Host Empty
D6	Data From Host Available
D5	Logic 1
D4	Logic 1
D3	Host Interrupt Pending
D2	Logic 1
D1	Status To Host Empty
D0	Command From Host Available

The BIART reads this register to determine the hardware status of its I/O registers and S-100 bus line pINT*. All **Flag Register** bits are controlled by BIART hardware.

D7 Data To Host Empty

This bit is reset when the BIART writes a data byte to register **Data To Host.** This bit is set (signifying that the BIART may write another data byte to the host processor) immediately after the host reads the **Data To Host** register. This event may, optionally, be programmed to generate an internal BIART interrupt by setting bit **Enable BIART Interrupts** of register **BIART Control.** A BIART reset forces this bit to 1.

D6 Data From Host Available

This bit is set (signifying that a host written data byte is available for BIART reading) immediately after the host has loaded a data byte into register **Data From Host.** This event may, optionally, be programmed to generate an internal BIART interrupt by setting bit **Enable BIART Interrupts** of register **BIART Control.** This bit is reset when the BIART reads the **Data From Host** register. A BIART reset forces this bit to 0.

D3 Host Interrupt Pending

This bit is set when the BIART drives S-100 bus line pINT* active low, and is awaiting interrupt servicing from the host processor. This bit is reset after the host acknowledges the BIART interrupt request.

D1 Status From Host Empty

This bit is set immediately after the host reads register **Status To Host.** This event may optionally be programmed to generate an internal BIART interrupt by setting bit **Enable BIART Interrupts** of register **BIART Control.** This bit is reset when the BIART outputs a byte to register **Status To Host.** A BIART reset forces this bit to 1.

D0 Command From Host Available

This bit is set as the host processor writes a byte to register **Command From Host.** This event may be programmed to generate an internal BIART interrupt by setting bit **Enable BIART Interrupts** of register **BIART Control.** This bit is reset when the BIART sets bit **Command From Host Empty** in register **Status To Host.** Note that the bit is **not** automatically reset when the BIART reads the **Command From Host** register. A BIART reset forces this bit to 0.

Register
BIART CONTROL

BIART: **OUT 02h**
Host: **No Access**

D7	Not Used
D6	Not Used
D5	Not Used
D4	Not Used
D3	Enable BIART RS-422 Transmitter
D2	Enable BIART Interrupts
D1	Enable Parallel Port Interrupts
D0	Enable Host Interrupts

D3 Enable BIART RS-422 Transmitter

When this bit is set, the RS-422 transmitters on the BIART are enabled; otherwise they are disabled (high impedance). When power is first turned on, this bit is reset, and the transmitters are disabled. This function is very useful for multi-drop communication.

D2 Enable BIART Interrupts

If this bit is set, a maskable interrupt request is issued to the BIART Z80B when the host exchanges I/O data with the BIART registers (port Bbase+00h or port Bbase+01h). Interrupt vector FCh, which anticipates either a Z80B interrupt mode IM1 or IM2 response, is automatically placed on the internal BIART data bus during Interrupt Acknowledge. Resetting this bit inhibits this feature. This bit is automatically reset by a BIART reset.

D1 Enable Parallel Port to BIART Interrupts

When this bit is set, parallel port to BIART interrupts are enabled. When it is reset, they are disabled. (This affects SENSE* only - pin 15.)

D0 Enable Host Interrupts

When this bit is set, S-100 Bus line *pINT is forced active low, thereby interrupting the host processor each time a vector is loaded into register **Interrupt Vector To Host.** When the host processor acknowledges the interrupt, the contents of register **Interrupt Vector To Host** are placed on the S-100 Data In bus, and the BIART releases line pINT*. Resetting this bit inhibits subsequent interrupts to the host, but is does **not** remove any interrupt request which is pending at the time the bit is reset. A BIART reset both resets this bit and removes any pending interrupt request to the host from the BIART.

Register
INTERRUPT VECTOR TO HOST

BIART: **OUT 03h**
Host: **No Access**

D7	Vector Bit V7 (MSB)
D6	Vector Bit V6
D5	Vector Bit V5
D4	Vector Bit V4
D3	Vector Bit V3
D2	Vector Bit V2
D1	Vector Bit V1
D0	Vector Bit V0 (LSB)

Loading a vector into this register issues a maskable interrupt request to the host processor if bit **Enable Host Interrupts** of register **BIART Control** is set. When the host processor acknowledges the request, the BIART places the contents of this register on the S-100 Data In bus to vector the host processor to an appropriate BIART service routine. If bit **Enable Host Interrupts** is reset, then BIART outputs to this register are null operations. If there is more than one S-100 interrupt source, BIART interrupt requests are prioritized by the S-100 interrupt cable (BIART connector J2). If the host processor is operating in interrupt mode IM0, the BIART typically outputs the opcode corresponding to one of eight RST (restart) instructions to this register. If operating in IM1, the value output is irrelevant since the Z80B defaults to host memory address 0038h for interrupt servicing. If operating in IM2, the BIART would output a byte which, when concatenated with the contents of the host's Z80B I-register, yields the indirect jump address of the service routine in host memory. The contents of this register are unaffected by a BIART reset, and should be assumed to be random until written for the first time.

Register
PARALLEL INPUT

BIART: **IN 1Xh**
Host: **No Access**

D7	Bit 7 In, Connector J4 (MSB)
D6	Bit 6 In, Connector J4
D5	Bit 5 In, Connector J4
D4	Bit 4 In, Connector J4
D3	Bit 3 In, Connector J4
D2	Bit 2 In, Connector J4
D1	Bit 1 In, Connector J4
D0	Bit 0 In, Connector J4 (LSB)

This register reads 8 parallel input bits from BIART connector J4 in non-inverted form (see table 2-1). The port address of register **Parallel Input** is 1Xh; that is, the most significant hex digit must be a 1, while the least significant hex digit may be any value (0h through Fh).

Register
PARALLEL OUTPUT

BIART: **OUT 1Xh**
Host: **No Access**

D7	Bit 7 Out, Connector J4 (MSB)
D6	Bit 6 Out, Connector J4
D5	Bit 5 Out, Connector J4
D4	Bit 4 Out, Connector J4
D3	Bit 3 Out, Connector J4
D2	Bit 2 Out, Connector J4
D1	Bit 1 Out, Connector J4
D0	Bit 0 Out, Connector J4 (LSB)

This register writes 8 parallel output bits to BIART connector J4 in non-inverted form (see table 2-1). The digits written to register **Parallel Output** are latched, and are not affected by a BIART reset. The port address of register **Parallel Output** is 1Xh; that is, the most significant hex digit must be a 1, while the least significant hex digit may be any value (0h through Fh).

Register
CONFIGURE MEMORY

BIART: **OUT 4Xh**
Host: **No Access**

D7	Not Used
D6	Not Used
D5	Not Used
D4	Not Used
D3	Not Used
D2	Not Used
D1	Not Used
D0	RAM/ROM*

Bit D0 of this register is used to control the BIART memory configuration. The port address of register **Configure Memory** is 4Xh; that is, the most significant hex digit must be a 4, while the least significant hex digit may be any value (0h through Fh).

D0 RAM/ROM*

A BIART reset forces this bit set. If this bit is set, BIART on-board memory is configured as follows:

C000h - FFFFh (16 Kbytes): ROM (see note)
4000h - BFFFh (32 Kbytes): RAM
0000h - 3FFFh (16 Kbytes): ROM (see note)

Note: These two 16 Kbyte blocks of ROM are **not** independent. The same 16 Kbyte ROM chip is mapped into both low memory (0000h - 3FFFh) and high memory (C000h - FFFFh).

If bit RAM/ROM* is reset, BIART on-board memory is configured as follows:

0000h - FFFFh (64 Kbytes): RAM

Register
CHANNEL B STATUS

BIART: **IN 50h**
Host: **No Access**

D7	RR Bit 7 (MSB)
D6	RR Bit 6
D5	RR Bit 5
D4	RR Bit 4
D3	RR Bit 3
D2	RR Bit 2
D1	RR Bit 1
D0	RR Bit 0 (LSB)

This register is mapped to access the nine Z-SCC internal Channel B read registers; RR0, RR1, RR2, RR3, RR8, RR10, RR12, RR13, and R13. The function of each bit varies from one register to the next. For example, Bit 0 in RR0 reports on **Rx Character Available** status, while Bit 0 in RR12 is the LSB of the current 16-bit Channel B time constant. See Reference 2 for descriptions of all read register bits.

Only RR0 can be read directly from BIART register **Channel B Status** with a single input instruction, e.g., IN A,(00h). All other SCC read registers must be read in two steps. First, output a byte to Channel B write register WR0 through BIART register **Channel B Commands**, and make the six low-order bits of this byte point to the target read register. Second, input a byte from BIART register **Channel B Status** to read the target register itself. For example, to read RR12, output byte XX001100b to register **Channel B Commands**, then input a byte from register **Channel B Status** to read RR12.

Register
CHANNEL B COMMANDS

BIART: **OUT 50h**
Host: **No Access**

D7	WR Bit 7 (MSB)
D6	WR Bit 6
D5	WR Bit 5
D4	WR Bit 4
D3	WR Bit 3
D2	WR Bit 2
D1	WR Bit 1
D0	WR Bit 0 (LSB)

This register is mapped to access the 16 Z-SCC internal Channel B write registers, WR0 through WR15. The function of each bit varies from one register to the next. For example, Bit 0 in WR3 enables and disables the Channel B receiver, while Bit 0 in WR12 programs the LSB of the 16-bit Channel B time constant. The Z-SCC has no default state, so all Channel B write registers must be programmed to define the channel characteristics (mode, clock multiplier, character length, parity, and so on) **before** the channel is used. See Reference 2 for descriptions of all write register bits.

Only WR0 can be directly accessed through BIART register **Channel B Commands** with a single output instruction, e.g., OUT (50h), A. All other SCC write registers must be accessed in two steps. First, output a byte to Channel B write register WR0 through BIART register **Channel B Commands**, and make the six low-order bits of this byte point to the target write register. Second, output another byte through BIART register **Channel B Commands** to the target register itself. For example, to write to WR12, output byte XX001100b to register **Channel B Commands**, then output the byte destined for WR12 to register **Channel B Commands.**

Use a BR clock period of 250 nSec (derived from a 4 MHz crystal clock) when programming a channel's baud rate generator for asynchronous operation.

Register
CHANNEL B RECEIVE DATA

BIART: **IN 51h**
Host: **No Access**

D7	Receive Data Bit 7 (MSB)
D6	Receive Data Bit 6
D5	Receive Data Bit 5
D4	Receive Data Bit 4
D3	Receive Data Bit 3
D2	Receive Data Bit 2
D1	Receive Data Bit 1
D0	Receive Data Bit 0 (LSB)

This register is mapped to access data characters assembled by the Z-SCC Channel B receiver. The Channel B receiver assembles serial data into 5- to 8-bit characters, and feeds these characters to the bottom of a three-deep receive register in the Z-SCC, where the characters are queued for reading. The top of the FIFO feeds register **Channel B Receive Data.** When an assembled character rises to the top of the FIFO, bit **Rx Character Available** in register RR0 is set (the Z-SCC can be programmed to interrupt the BIART Z80B when this occurs). Reading a character from register **Channel B Receive Data** removes the FIFO's top character, resets bit **Rx Character Available,** and makes room for another character at the bottom of the FIFO.

Data characters are shifted into the receiver LSB first, MSB last. Asynchronous mode characters are always right justified with a parity bit, or extraneous bits, to the left. Synchronous mode characters are also right justified, provided that the transmit and receive character lengths match, and are not changed during the transmission.

Register
CHANNEL B TRANSMIT DATA

BIART: **OUT 51h**
Host: **No Access**

D7	Transmit Data Bit 7 (MSB)
D6	Transmit Data Bit 6
D5	Transmit Data Bit 5
D4	Transmit Data Bit 4
D3	Transmit Data Bit 3
D2	Transmit Data Bit 2
D1	Transmit Data Bit 1
D0	Transmit Data Bit 0 (LSB)

This register is mapped to access the internal SCC Channel B transmitter buffer. This buffer empties directly into the Channel B transmitter either immediately, or as soon as the preceding character is completely shifted out. Bit **Tx Buffer Empty** of register RR0 is set whenever the buffer becomes empty, and is reset when a character is loaded into the buffer through register **Channel B Transmit Data.** The Z-SCC can be programmed to interrupt the BIART Z80B when the Tx buffer becomes empty.

Assembled characters of 5 to 8 bits in length should be loaded into register **Channel B Transmit Data** (right justified). Characters less than 5 bits in length must be specially formatted (see Reference 2). Data bits are shifted out of the Channel B transmitter LSB first, MSB last.

Register
CHANNEL A STATUS

BIART: **IN 52h**
Host: **No Access**

See the description of register **Channel B Status.**

Register
CHANNEL A COMMANDS

BIART: **OUT 52h**
Host: **No Access**

See the description of register **Channel B Commands.**

Register
CHANNEL A RECEIVE DATA

BIART: **IN 53h**
Host: **No Access**

See the description of register **Channel B Receive Data.**

Register
CHANNEL A TRANSMIT DATA

BIART: **OUT 53h**
Host: **No Access**

See the description of register **Channel B Transmit Data.**

Appendix B

RS-232C, RS-422, AND RS-423 INTERFACES

The BIART board supports three standard serial interfaces: RS-232C, RS-422, and RS-423. The essential specifications of these standards are shown in table B-1.

Table B-1: SERIAL INTERFACE SPECIFICATIONS

Characteristics	EIA RS-232C	EIA RS-423	EIA RS-422
Form of Operation	Single Ended	Single Ended	Differential
Max. cable length	50 FT	2000 FT	4000 FT
Max. baud rate	20K	300K	10M
Driver output voltage, Loaded output	±5V to ±15V	±3.6V to ±5.4V	2V Difference
Driver output resistance power off	Ro = 300 ohms	100uA between -6 to +6V	100uA between +6 and -.25V
Driver output short circuit current I_{SC}	±500mA	±150mA	±150mA
Receiver input resistance (ohms)	3K to 7K	≥4K	4K
Receiver input thresholds	-3V to +3V	-0.2V to +0.2V	-0.2V to +0.2V
Receiver input voltage	-25V to +25V	-12V to +12V	-12V to +12V

Figures B-1 through B-3 show how each interface connects to the BIART board. Though only signals TXD-A and RXD-A are shown in the figures, all the other signals (CTS, RTS, DCD, DTR, TXC, and RXC) are connected in the same manner. The pin numbers apply to both serial channels (refer to table 1-2).

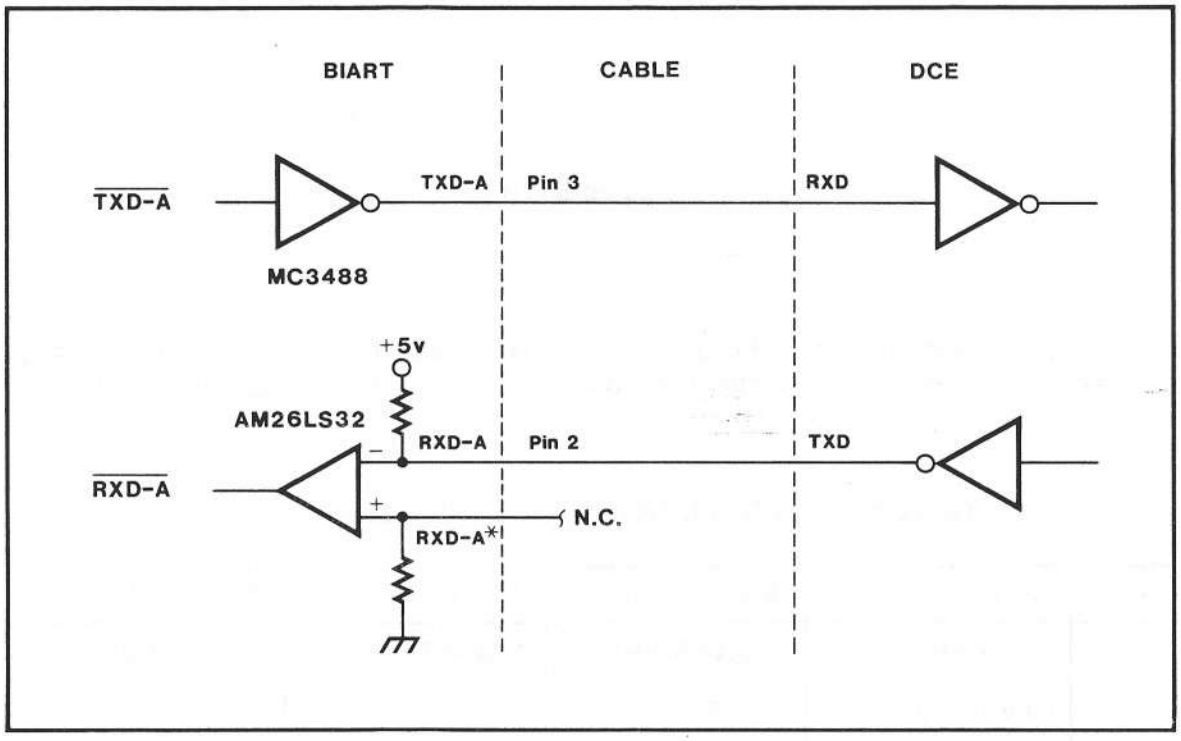


Figure B-1: RS-232C INTERFACE WIRING

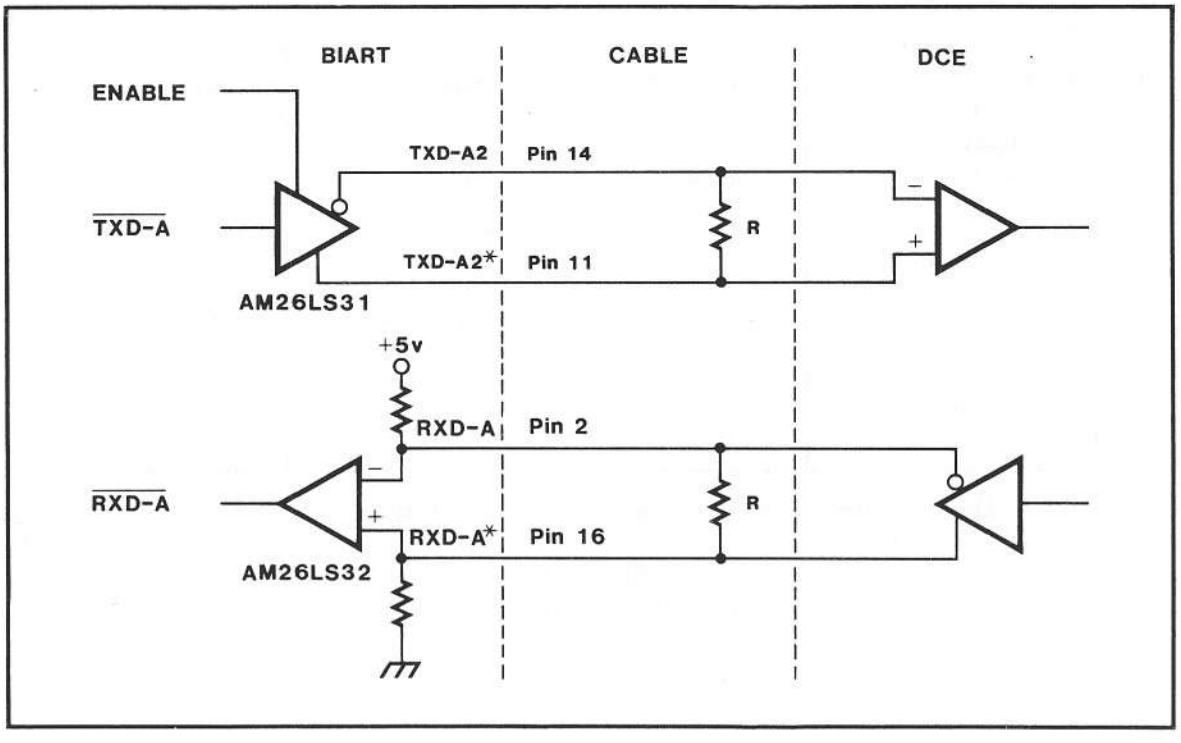


Figure B-2: RS-422 INTERFACE WIRING

In figure B-2, resistor R is a termination resistor, and the "Enable" bit is bit 3 of the **BIART control register** (refer to appendix A).

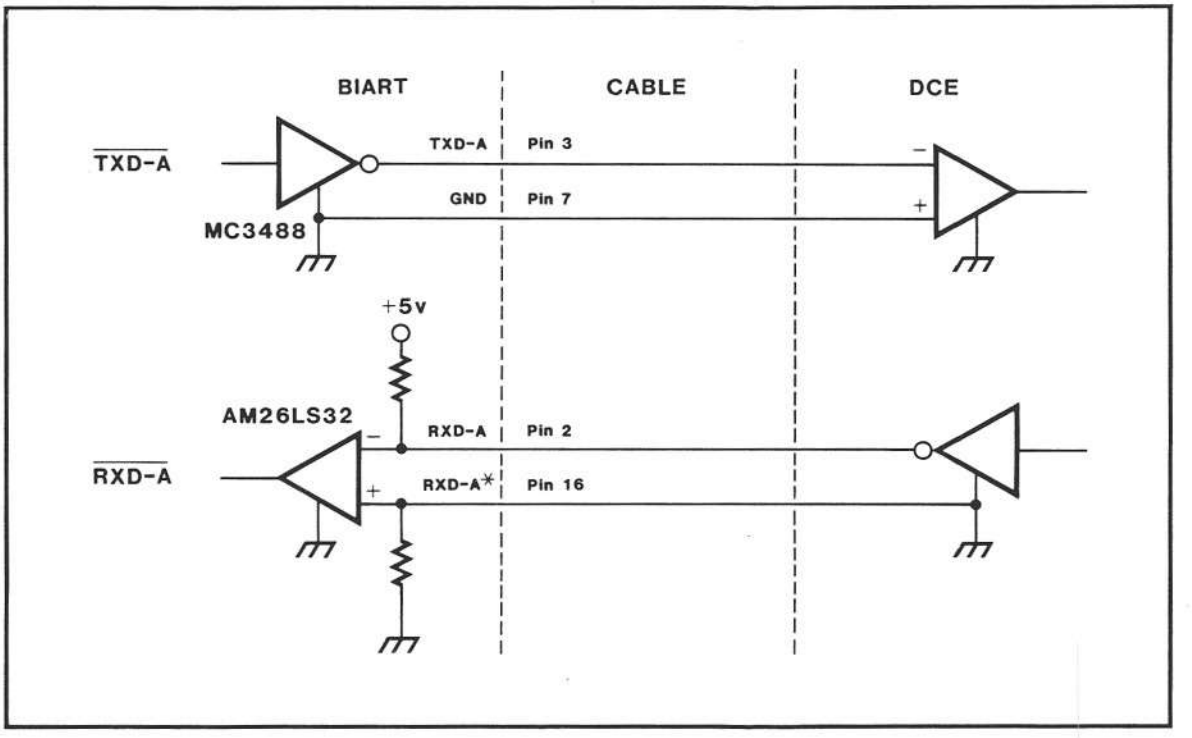


Figure B-3: RS-423 INTERFACE WIRING

Appendix C

SIX-BIT TRANSCODE

00h	SOH	20h	-
01h	A	21h	/
02h	B	22h	S
03h	C	23h	T
04h	D	24h	U
05h	E	25h	V
06h	F	26h	W
07h	G	27h	X
08h	H	28h	Y
09h	I	29h	Z
0Ah	STX	2Ah	ESC
0Bh	.	2Bh	,
0Ch	<	2Ch	%
0Dh	BEL	2Dh	ENQ
0Eh	SUB	2Eh	ETX
0Fh	ETB	2Fh	HT
10h	&	30h	0
11h	J	31h	1
12h	K	32h	2
13h	L	33h	3
14h	M	34h	4
15h	N	35h	5
16h	O	36h	6
17h	P	37h	7
18h	Q	38h	8
19h	R	39h	9
1Ah	SPACE	3Ah	SYN
1Bh	$	3Bh	#
1Ch	*	3Ch	@
1Dh	US	3Dh	NAK
1Eh	EOT	3Eh	EM
1Fh	DLE	3Fh	DEL

Appendix D

ASCII CHARACTER CODE

00h	NUL	(CONTROL-@)	20h	SPACE	40h	@	60h	`
01h	SOH	(CONTROL-A)	21h	!	41h	A	61h	a
02h	STX	(CONTROL-B)	22h	"	42h	B	62h	b
03h	ETX	(CONTROL-C)	23h	#	43h	C	63h	c
04h	EOT	(CONTROL-D)	24h	$	44h	D	64h	d
05h	ENQ	(CONTROL-E)	25h	%	45h	E	65h	e
06h	ACK	(CONTROL-F)	26h	&	46h	F	66h	f
07h	BEL	(CONTROL-G)	27h	'	47h	G	67h	g
08h	BS	(CONTROL-H)	28h	(	48h	H	68h	h
09h	HT	(CONTROL-I)	29h	)	49h	I	69h	i
0Ah	LF	(CONTROL-J)	2Ah	*	4Ah	J	6Ah	j
0Bh	VT	(CONTROL-K)	2Bh	+	4Bh	K	6Bh	k
0Ch	FF	(CONTROL-L)	2Ch	,	4Ch	L	6Ch	l
0Dh	CR	(CONTROL-M)	2Dh	-	4Dh	M	6Dh	m
0Eh	SO	(CONTROL-N)	2Eh	.	4Eh	N	6Eh	n
0Fh	SI	(CONTROL-O)	2Fh	/	4Fh	O	6Fh	o
10h	DLE	(CONTROL-P)	30h	0	50h	P	70h	p
11h	DC1	(CONTROL-Q)	31h	1	51h	Q	71h	q
12h	DC2	(CONTROL-R)	32h	2	52h	R	72h	r
13h	DC3	(CONTROL-S)	33h	3	53h	S	73h	s
14h	DC4	(CONTROL-T)	34h	4	54h	T	74h	t
15h	NAK	(CONTROL-U)	35h	5	55h	U	75h	u
16h	SYN	(CONTROL-V)	36h	6	56h	V	76h	v
17h	ETB	(CONTROL-W)	37h	7	57h	W	77h	w
18h	CAN	(CONTROL-X)	38h	8	58h	X	78h	x
19h	EM	(CONTROL-Y)	39h	9	59h	Y	79h	y
1Ah	SUB	(CONTROL-Z)	3Ah	:	5Ah	Z	7Ah	z
1Bh	ESC	(CONTROL-[)	3Bh	;	5Bh	[	7Bh	{
1Ch	FS	(CONTROL-\)	3Ch	<	5Ch	\	7Ch	|
1Dh	GS	(CONTROL-])	3Dh	=	5Dh	]	7Dh	}
IEh	RS	(CONTROL-^)	3Eh	>	5Eh	^	7Eh	~
1Fh	US	(CONTROL-_)	3Fh	?	5Fh	_	7Fh	DEL

Appendix E

EBCDIC CHARACTER CODE

00h	NUL	2Ah	SM	7Dh	'	C7h	G
01h	SOH	2Dh	ENQ	7Eh	=	C8h	H
02h	STX	2Eh	ACK	7Fh	"	C9h	I
03h	ETX	2Fh	BEL	81h	a	D0h	}
04h	PF	32h	SYN	82h	b	D1h	J
05h	HT	34h	PN	83h	c	D2h	K
06h	LC	35h	RS	84h	d	D3h	L
07h	DEL	36h	UC	85h	e	D4h	M
09h	RLF	37h	EOT	86h	f	D5h	N
0Ah	SMM	3Ch	DC4	87h	g	D6h	O
0Bh	VT	3Dh	NAK	88h	h	D7h	P
0Ch	FF	3Fh	SUB	89h	i	D8h	Q
0Dh	CR	40h	SPACE	91h	j	D9h	R
0Eh	SO	4Ah	¢	92h	k	E0h	\
0Fh	SI	4Bh	.	93h	l	E2h	S
10h	DLE	4Ch	<	94h	m	E3h	T
11h	DC1	4Dh	(	95h	n	E4h	U
12h	DC2	4Eh	+	96h	o	E5h	V
13h	DC3	4Fh	|	97h	p	E6h	W
14h	RES	50h	&	98h	q	E7h	X
15h	NL	5Ah	!	99h	r	E8h	Y
16h	BS	5Bh	$	A1h	~	E9h	Z
17h	IL	5Ch	*	A2h	s	F0h	0
18h	CAN	5Dh	)	A3h	t	F1h	1
19h	EM	5Eh	;	A4h	u	F2h	2
1Ah	CC	5Fh	^	A5h	v	F3h	3
1Ch	IFS	60h	-	A6h	w	F4h	4
1Dh	IGS	61h	/	A7h	x	F5h	5
1Eh	IRS	6Ah	|	A8h	y	F6h	6
1Fh	IUS	6Bh	,	A9h	z	F7h	7
20h	DS	6Ch	*	C0h	{	F8h	8
21h	SOS	6Eh	>	C1h	A	F9h	9
22h	FS	6Fh	?	C2h	B		
24h	BYP	79h	`	C3h	C		
25h	LF	7Ah	:	C4h	D		
26h	EOB/ETB	7Bh	#	C5h	E		
27h	PRE/ESC	7Ch	@	C6h	F		

Appendix F

PARTS LIST

Integrated Circuits

Designation	Cromemco Description	Part No.
IC1,2	MC3488	010-0416
IC3	26LS31	010-0361
IC4,5	26LS32	010-0360
IC6	26LS31	010-0361
IC7	26LS32	010-0360
IC8	74LS00	010-0069
IC9	7404	010-0030
IC10	CPU & Memory Support	011-0095
	Socket 40-pin	017-0006
IC11	27128 ROM	502-0096
	Socket 28-pin	017-0071
IC12	74ALS157	010-0412
IC13	74LS244	010-0100
IC14	74LS373	010-0102
IC15	MC3488	010-0416
IC16	74ALS175	010-0358
IC17	74LS32	010-0058
IC18	Z8530A	011-0109
	Socket 40-pin	017-0006
IC19	74LS08	010-0064
IC20	74S74	010-0142
IC21	PAL Terminal Interface	502-0110
	Socket 20-pin	017-0004
IC22	74LS244	010-0100
IC23	74LS138	010-0096
IC24	Z8400/MK3880-6	011-0113
	Socket 40-pin	017-0006
IC25	74ALS157	010-0412
IC26,27	74LS244	010-0100
IC28	74LS74	010-0055
IC29	74LS32	010-0058
IC30	74LS04	010-0066
IC31	7812/340T-12	012-0002
IC32	74LS00	010-0069
IC33	74S74	010-0142

Integrated Circuits (Continued)

Designation	Cromemco Description	Part No.
IC34	74LS74	010-0055
IC35	74LS138	010-0096
IC36	74LS21	010-0060
IC37	74LS153	010-0048
IC38	74LS04	010-0066
IC39	74LS74	010-0055
IC40	74ALS74	010-0357
IC41	74LS244	010-0100
IC42	74LS174	010-0097
IC43-46	4164-150nS RAM	011-0079
IC47-48	7805/340T-5	012-0001
IC49	7912	012-0014
IC50	7407	010-0104
IC51	AMZ8121	010-0328
IC52	82S159	502-0072
	Socket 20-pin	017-0004
IC53-56	74LS373	010-0102
IC57	74LS244	010-0100
IC58-61	4164-150nS RAM	011-0079

Diodes/ Transistors

Designation	Cromemco Description	Part No.
D1-2	1N4148	008-0002
Q1	2N3906	009-0002

Capacitors

Designation	Cromemco Description	Part No.
C1	.005 uf 100V	004-0142
C2-3	.001 uf crdc	004-0141
C4	10 uf 20V	004-0032
C5	33 pf crdc	004-0146
C7-8	10 uf 20V	004-0032
C9	180 pf npo	004-0168
C10	1 uf 50V	004-0147

Capacitors (Continued)

Designation	Cromemco Description	Part No.
C11	10 uf 20v	004-0032
C12	47 pf crdc	004-0152
C13	10 uf 20V	004-0032
C14	1 uf 50V	004-0147
C15	10 uf 20V	004-0032
C16	47 pf crdc	004-0152
C17 & 41 bypass	.047 uf 50V	004-0061

Capacitor Networks

Designation	Cromemco Description	Part No.
CN1,2	47 pf 8-pin	005-0000

Resistors

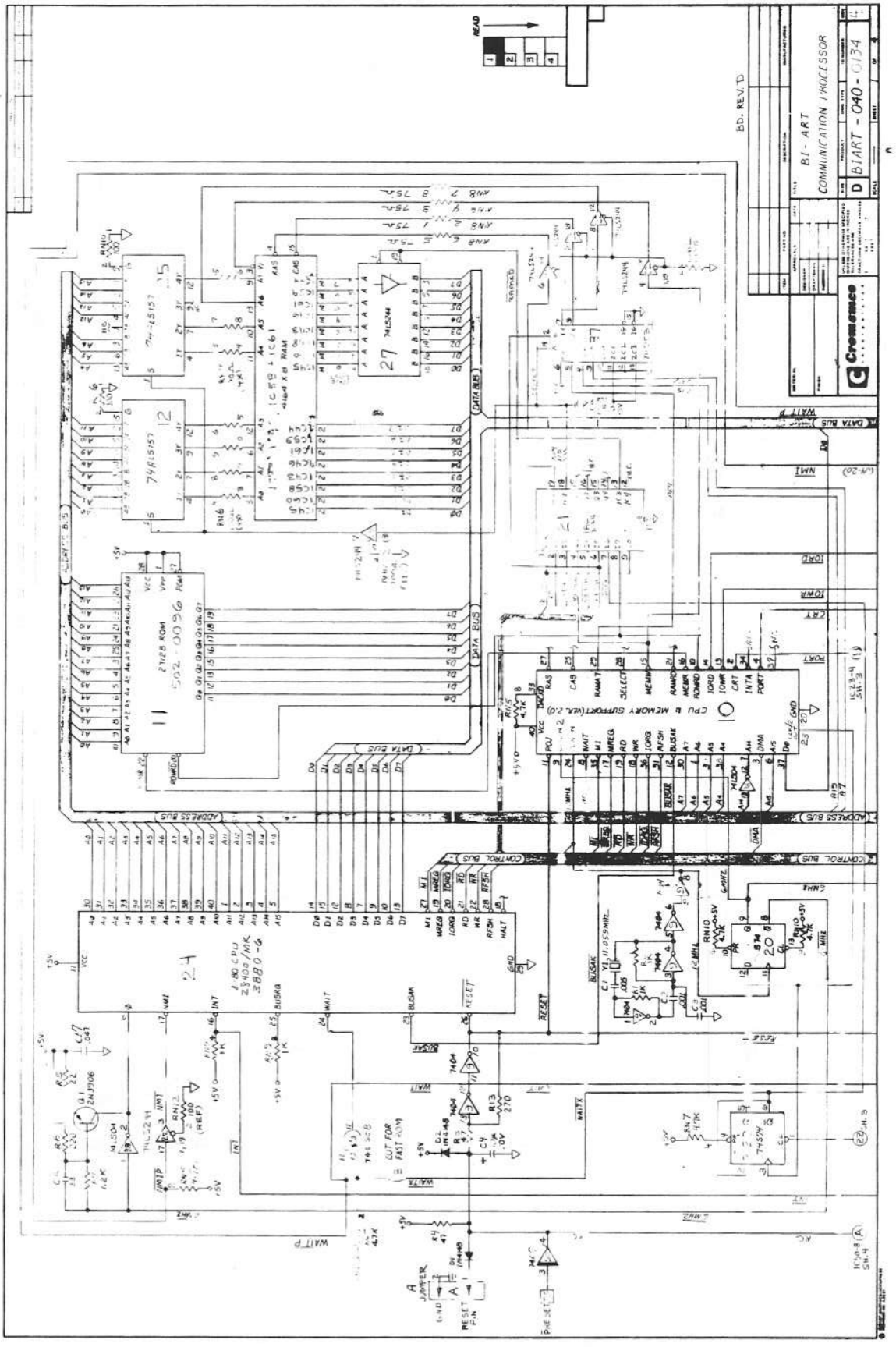
Designation	Cromemco Description	Part No.
R1,2	1 Kohm 1/4 watt	001-0018
R3	47 ohm 1/4 watt	001-0003
R4	470 ohm 1/4 watt	001-0014
R5	22 ohm 1/4 watt	001-0001
R6	220 ohm 1/4 watt	001-0010
R7	1.2 Kohm 1/4 watt	001-0019
R8-10	220 Kohm 1/4 watt	001-0078
R11	4.7 Kohm 1/4 watt	001-0024
R12	1 Kohm 1/4 watt	001-0018
R13	270 ohm 1/4 watt	001-0011
R14	110 ohm 1/4 watt	001-0069
R16	100 ohm 1/4 watt	001-0007
R17	4.7 kohm 1/4 watt	001-0024

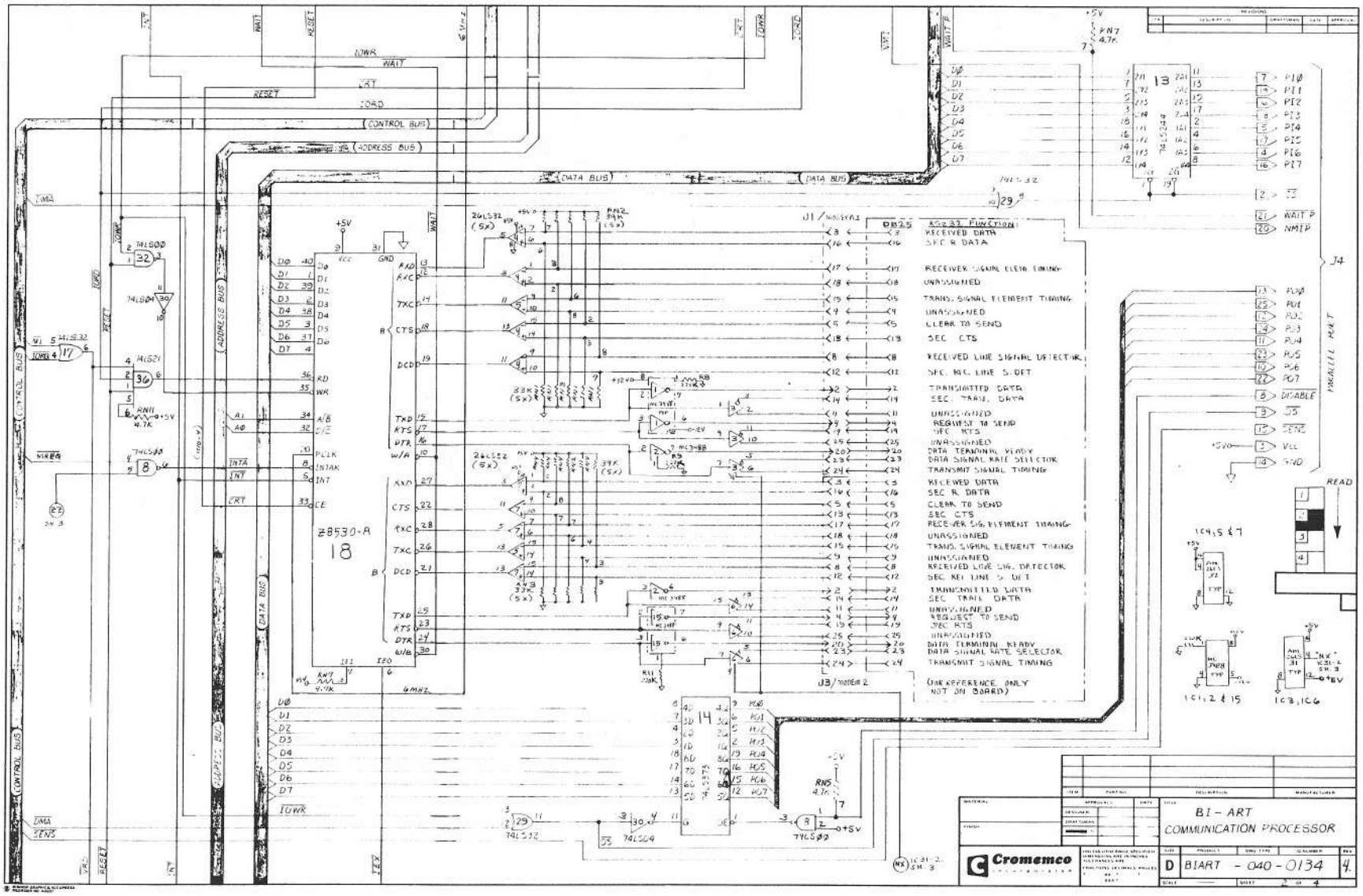
Resistor Networks

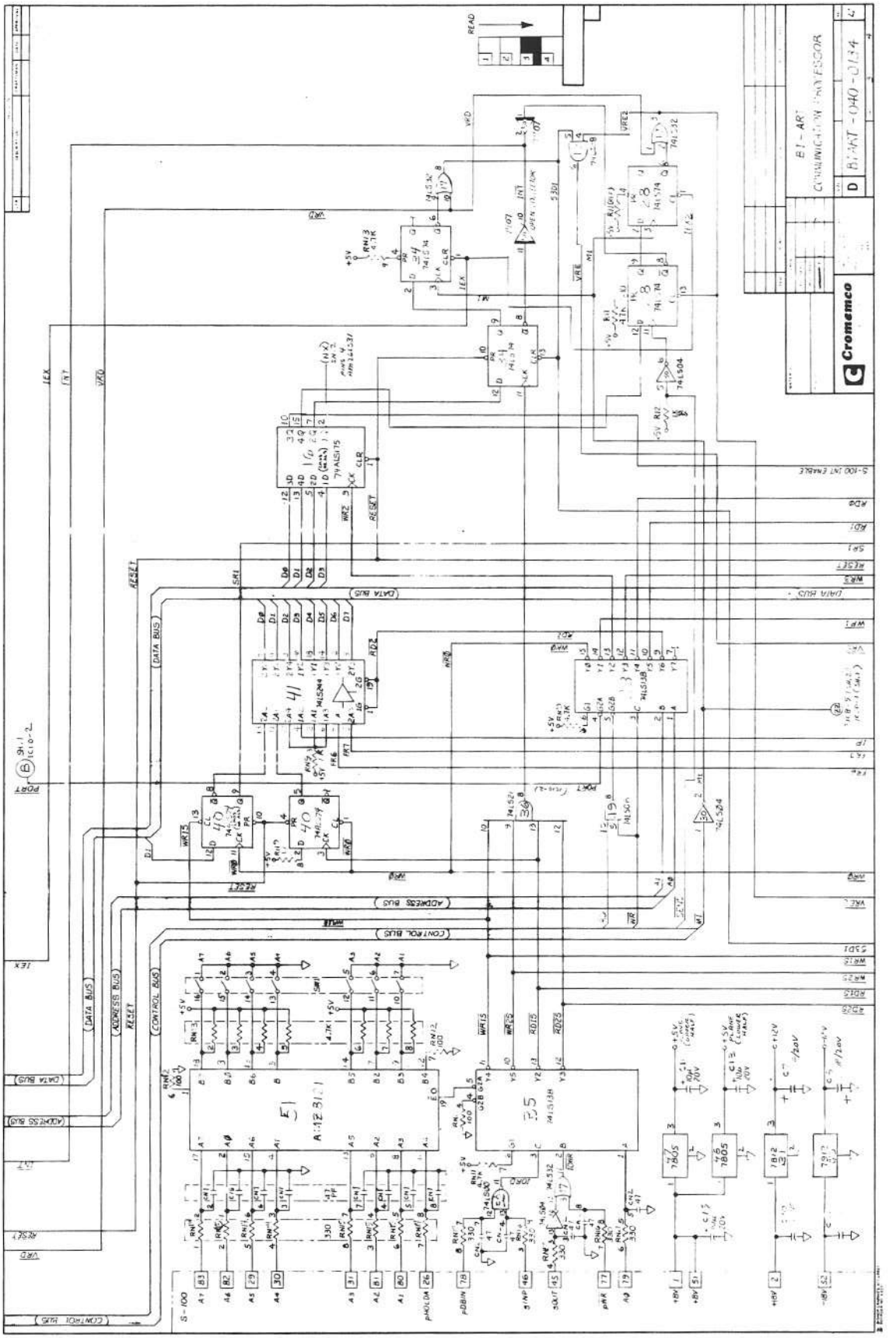
Designation	Cromemco Description	Part No.
RN1	33 Kohm 7R 8-pin	003-0096
RN2	39 Kohm 7R 8-pin	003-0097
RN3	33 Kohm 7R 8-pin	003-0096
RN4	39 Kohm 7R 8-pin	003-0097
RN5	4.7 Kohm 7R 8-pin	003-0009
RN6	100 ohm 5R 10-pin	003-0093
RN7	4.7 Kohm 9R 10-pin	003-0014
RN8	75 ohm 4R 8-pin	003-0080
RN9	1 Kohm 7R 8-pin	003-0007
RN10	100 ohm 5R 10-pin	003-0093
RN11	4.7 Kohm 7R 8-pin	003-0009
RN12	100 ohm 7R 8-pin	003-0036
RN13	4.7 Kohm 9R 10-pin	003-0014
RN14-17	330 ohm 4R 8-pin	003-0004

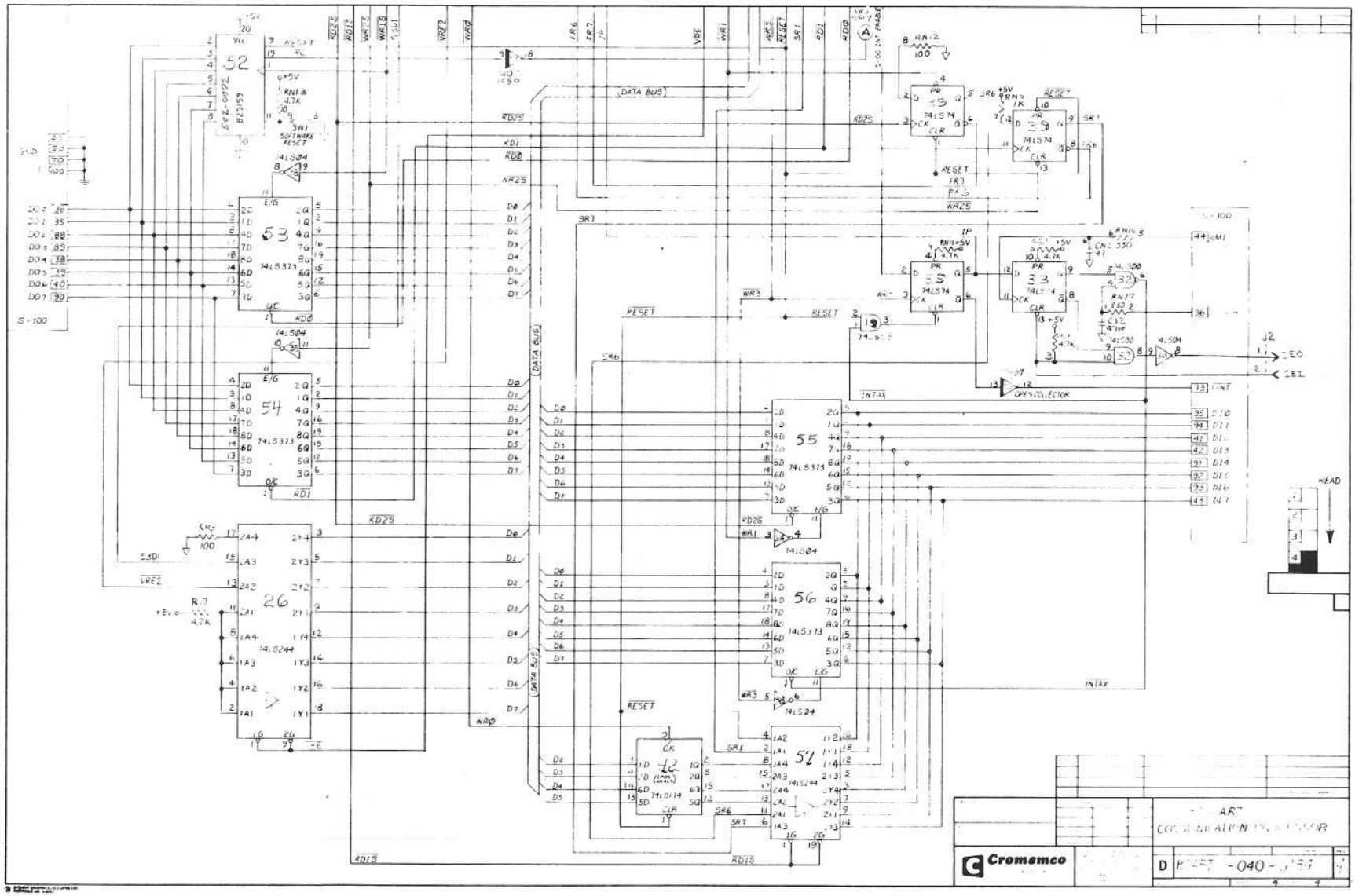
Miscellaneous

Designation	Cromemco Description	Part No.
HS1	Heatsink dual TO-220	021-0165
J1	26-pin connector	017-0259
J2	2-pin connector	017-0009
J3,4	26-pin connector	017-0259
	Lockwasher #6	015-0020
	Lockwasher #4	015-0139
	Spacer 1/4 hex 6-32 x 1/4	015-0169
	Standoff 4-40 x 1/4 hex	015-0173
	BIART PC board	020-0134
	Solder stud TO-220	021-0164
	Sil-pad	021-0109
SW1	Switch 8 position DIP	013-0002
Y1	Crystal 11.059 MHz	026-0037


BI - ART
COMMUNICATION PROCESSOR
BIART - 040 - 0134
Cromemco
BD. REV. D
CPU & MEMORY SUPPORT (REV. 2.0)
Z-80 CPU
Z8400/MK
Z8B0-6
21128 ROM
502-0096
74LS157
74LS244
74LS374
74LS04
74LS74
74LS32
74LS00
74LS20
74LS138
74LS139
74LS259
74LS367
74LS393
74LS123
74LS221
74LS08
74LS86
74LS175
74LS273
74LS283
74LS299
74LS373
74LS374
74LS377
74LS393
74LS670
74LS688
74LS689
74LS690
74LS691
74LS692
74LS693
74LS694
74LS695
74LS696
74LS697
74LS698
74LS699
74LS700
74LS701
74LS702
74LS703
74LS704
74LS705
74LS706
74LS707
74LS708
74LS709
74LS710
74LS711
74LS712
74LS713
74LS714
74LS715
74LS716
74LS717
74LS718
74LS719
74LS720
74LS721
74LS722
74LS723
74LS724
74LS725
74LS726
74LS727


BI - ART
COMMUNICATION PROCESSOR
Cromemco
D BIART - 040 - 0134 4.
Z8530-A
18
74LS244
74LS373
74LS00
74LS04
74LS32
74LS21
74LS00
74LS00
74LS32
74LS04
74LS00
26LS32 (5X)
26LS32 (5X)
RS232 FUNCTION
RECEIVED DATA
SEC R DATA
RECEIVER SIGNAL ELEM TIMING
UNASSIGNED
TRANS. SIGNAL ELEMENT TIMING
UNASSIGNED
CLEAR TO SEND
SEC CTS
RECEIVED LINE SIGNAL DETECTOR
SEC. REC. LINE S. DET
TRANSMITTED DATA
SEC. TRANS. DATA
UNASSIGNED
REQUEST TO SEND
SEC RTS
UNASSIGNED
DATA TERMINAL READY
DATA SIGNAL RATE SELECTOR
TRANSMIT SIGNAL TIMING
RECEIVED DATA
SEC R DATA
CLEAR TO SEND
SEC CTS
RECEIVER SIG. ELEMENT TIMING
UNASSIGNED
TRANS. SIGNAL ELEMENT TIMING
UNASSIGNED
RECEIVED LINE SIG. DETECTOR
SEC REC LINE S. DET
TRANSMITTED DATA
SEC TRANS DATA
UNASSIGNED
REQUEST TO SEND
SEC RTS
UNASSIGNED
DATA TERMINAL READY
DATA SIGNAL RATE SELECTOR
TRANSMIT SIGNAL TIMING
(FOR REFERENCE ONLY NOT ON BOARD)
J1 / MODEM 1
J3 / MODEM 2
J4
PARALLEL PORT
READ
IC4,5 & 7
IC1,2 & 15
IC3,IC6
IC31-2 SH. 3
CONTROL BUS
ADDRESS BUS
DATA BUS
IOWR
WAIT
RESET
IORD
CRT
INT
DMA
SENS
DISABLE
SEND
VCC
GND
PI0
PI1
PI2
PI3
PI4
PI5
PI6
PI7
PO0
PO1
PO2
PO3
PO4
PO5
PO6
PO7
WAIT P
NMIP
RN7 4.7K
RN5 4.7K
RN11 4.7K
RN7 4.7K
RN2 39K (5X)
RN3 39K (5X)
RN4 33K (5X)
RN5 33K (5X)
+5V
+12V
-12V
6MHZ


BI-ART
COMMUNICATIONS PROCESSOR
D
BIART-040-0134
Cromemco
READ
DATA BUS
ADDRESS BUS
CONTROL BUS
RESET
S-100


52
53
54
55
56
57
26
74LS373
74LS374
74LS244
74LS04
DATA BUS
RESET
INTAX
OPEN COLLECTOR
READ
Cromemco
D
-040-

*, Notation, 1

Access time, ROM, 5
ASCII character code, 43

Base address, BIART, 5
Baud rate, asynchronous mode, 32
Bbase, defined, 5, 12
BIART control, register, 26
BIART flags, register, 16, 17, 24
BIART reset, bit, 21

Cables, BIART, 6
Channel A commands, register, 35
Channel A receive data, register, 35
Channel A status, register, 35
Channel A transmit data, register, 35
Channel B commands, register, 32
Channel B receive data, register, 33
Channel B status, register, 31
Channel B transmit data, register, 34
Command from host available, bit, 15, 25
Command from host empty, bit, 15, 21
Commands from host, register, 15, 19
Configure memory, register, 30
Connector J1, BIART, 13
Connector J1, J3, BIART, 9
Connector J4, BIART, 9
Connectors, BIART, 6

Data from host available, bit, 16, 24
Data from host empty, bit, 16, 20
Data from host, register, 16, 22
Data to host available, bit, 17, 20
Data to host empty, bit, 17, 24
Data to host, register, 17, 23

Ebcdic character code, 45
Enable BIART interrupts, bit, 15, 26
Enable host interrupts, bit, 15, 26
Enable Parallel port to BIART interrupts, bit 26

Fifo, Z-SCC receive, 33

Host interrupt pending, bit, 24

Host interrupts, 15
Host/BIART, communications ports, 6, 15

In 00h, port, 31
In 01h, port, 33
In 02h, port, 35
In 03h, port, 35
In 1xh, port, 28
In 50h, port, 19
In 51h, port, 22
In 52h, port, 24
In bbase+00h, port, 20
In bbase+01h, port, 23
Installation, BIART board, 3
Internal BIART interrupts, 14
Interrupt modes, Z80B, 27
Interrupt vector to host, register, 15, 27
Interrupt vector, FAh, FCh, 14, 26
Interrupts, host, 15
Interrupts, internal BIART, 14
Interrupts, parallel port, 14
Interrupts, Z-SCC, 14

Jumper, ROM speed, 5

Memory configuration, BIART, 5, 13, 30

Out 00h, port, 32
Out 01h, port, 34
Out 02h, port, 35
Out 03h, port, 35
Out 1xh, port, 29
Out 4xh, port, 30
Out 50h, port, 20
Out 51h, port, 23
Out 52h, port, 26
Out 53h, port, 27
Out bbase*00h, port, 19
Out bbase+01h, port, 22

Parallel input, register, 28
Parallel output, register, 29
Parallel printer, 9
Pin-outs, BIART, 7
Positive logic, definition, 1
Program store, BIART, 11
Programming information, BIART, 11

RAM, BIART memory, 11
RAM/ROM*, bit, 30
Register descriptions, BIART, 31
Register summary, BIART, 12
Reset Connector, BIART, 7
Reset state, BIART registers, 14
Reset, BIART, 7, 13
Reset, definition, 1
Reset, S-100 bus, 13
Reset, software, BIART, 13
Reset, Z80B, 13
ROM, BIART memory, 5
RR0 - RR13, registers, Z-SCC, 31
RS-232C levels, BIART, 9

Set up, BIART, 3
Set, definition, 1
Six-bit transcode, 41
Status 5, 4, 3, 2, bit, 20
Status from host empty, bit, 25
Status to host empty, bit, 16
Status to host, register, 16, 20
Switch SW-1, BIART, 6, 12
Switch SW-1, section 8, 13

Transmit buffer, Z-SCC register, 34

WR0 - WR15, registers, Z-SCC, 32

Z-SCC read registers, 31
Z-SCC write registers, 32

LIMITED WARRANTY

Cromemco, Inc. ("Cromemco") warrants this equipment against defects in material and workmanship to the original purchaser for ninety (90) days from the date of purchase, subject to the following terms and conditions.

What Is Covered By This Warranty:

During the ninety (90) day warranty period Cromemco will, at its option, repair or replace this Cromemco product or repair or replace with new or used parts any parts or components, manufactured by Cromemco, which prove to be defective, provided the product is returned to your Dealer as set forth below.

How To Obtain Warranty Service:

You should immediately notify IN WRITING your Dealer or Cromemco of problems encountered during the warranty period in order to obtain warranty service; first obtain a return authorization number by contacting the Dealer from whom you purchased the product. Then attach to the product:

1. Your name, address, and telephone number,
2. the return authorization number
3. a description of the problem, and
4. proof of the date of retail purchase

Ship or otherwise return the product, transportation and insurance costs prepaid, to your Dealer. If you are unable to receive warranty repair from the Dealer from whom you purchased the product, you should contact Cromemco Customer Support at: Cromemco, Inc., 280 Bernardo Ave., Mountain View, Ca. 94043.

What Is Not Covered By This Warranty:

Cromemco does not warrant any products, components, or parts not manufactured by Cromemco.

This warranty does not apply if the product has been damaged by accident, abuse, misuse, modification, or misapplication; by damage during shipment; or by improper service. This product is not warranted to operate satisfactorily with peripherals or products not manufactured by Cromemco. Transportation and insurance charges incurred in transporting the product to and from your Dealer or Cromemco are not covered by this warranty.

Exclusion of Liability, Damages, and Other Warranties:

THIS WARRANTY IS IN LIEU OF ALL OTHER WARRANTIES, WHETHER ORAL OR WRITTEN, EXPRESS OR IMPLIED. ANY IMPLIED WARRANTIES, INCLUDING IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, ARE EXCLUDED WHERE SUCH EXCLUSION IS ALLOWED AND OTHERWISE LIMITED IN DURATION TO NINETY (90) DAYS FROM THE DATE OF PURCHASE OF THIS PRODUCT. IF THIS PRODUCT IS NOT IN GOOD WORKING ORDER AS WARRANTED ABOVE, YOUR SOLE REMEDY SHALL BE REPAIR OR REPLACEMENT AS PROVIDED ABOVE. CROMEMCO SHALL NOT BE LIABLE FOR INCIDENTAL AND/OR CONSEQUENTIAL DAMAGES FOR THE BREACH OF ANY EXPRESS OR IMPLIED WARRANTY, INCLUDING DAMAGE TO PROPERTY AND, TO THE EXTENT PERMITTED BY LAW, DAMAGES FOR PERSONAL INJURY, EVEN IF CROMEMCO HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. THE AGENTS, DEALERS, AND EMPLOYEES OF CROMEMCO ARE NOT AUTHORIZED TO MAKE MODIFICATIONS TO THIS WARRANTY, OR ADDITIONAL WARRANTIES BINDING ON CROMEMCO ABOUT OR FOR PRODUCTS SOLD OR LICENSED BY CROMEMCO. ACCORDINGLY, ADDITIONAL STATEMENTS WHETHER ORAL OR WRITTEN EXCEPT SIGNED WRITTEN STATEMENTS FROM AN OFFICER OF CROMEMCO DO NOT CONSTITUTE WARRANTIES AND SHOULD NOT BE RELIED UPON. SOFTWARE, TECHNICAL INFORMATION, AND FIRMWARE ARE LICENSED ONLY BY A SEPARATE AGREEMENT ON AN "AS IS" BASIS.

Limitation on Statute of Limitation and Transferability:

THIS WARRANTY AND THE STATUTE OF LIMITATIONS SHALL RUN CONCURRENTLY WITH ANY ACCEPTANCE PERIOD. THIS WARRANTY IS NOT TRANSFERABLE. NO SUIT, LITIGATION, OR ACTION SHALL BE BROUGHT BASED ON THE ALLEGED BREACH OF THIS WARRANTY OR IMPLIED WARRANTIES OR FOR OTHER CAUSE MORE THAN ONE YEAR AFTER THE DATE OF PURCHASE IN THOSE JURISDICTIONS ALLOWING SUCH A LIMITATION; OTHERWISE NO SUCH ACTION SHALL BE BROUGHT MORE THAN ONE YEAR AFTER THE EXPIRATION OF THIS WARRANTY.

Other Important Provisions:

Some states do not allow the exclusion or limitation of incidental or consequential damages or limitations on how long an implied warranty lasts, so the above limitation or exclusion may not apply to you. This warranty shall not be applicable to the extent that any provision of this warranty is prohibited by any federal, state, or municipal law which cannot be preempted. This warranty gives you specific legal rights, and you may also have other rights which vary from state to state.

Rev. D 6/14/84