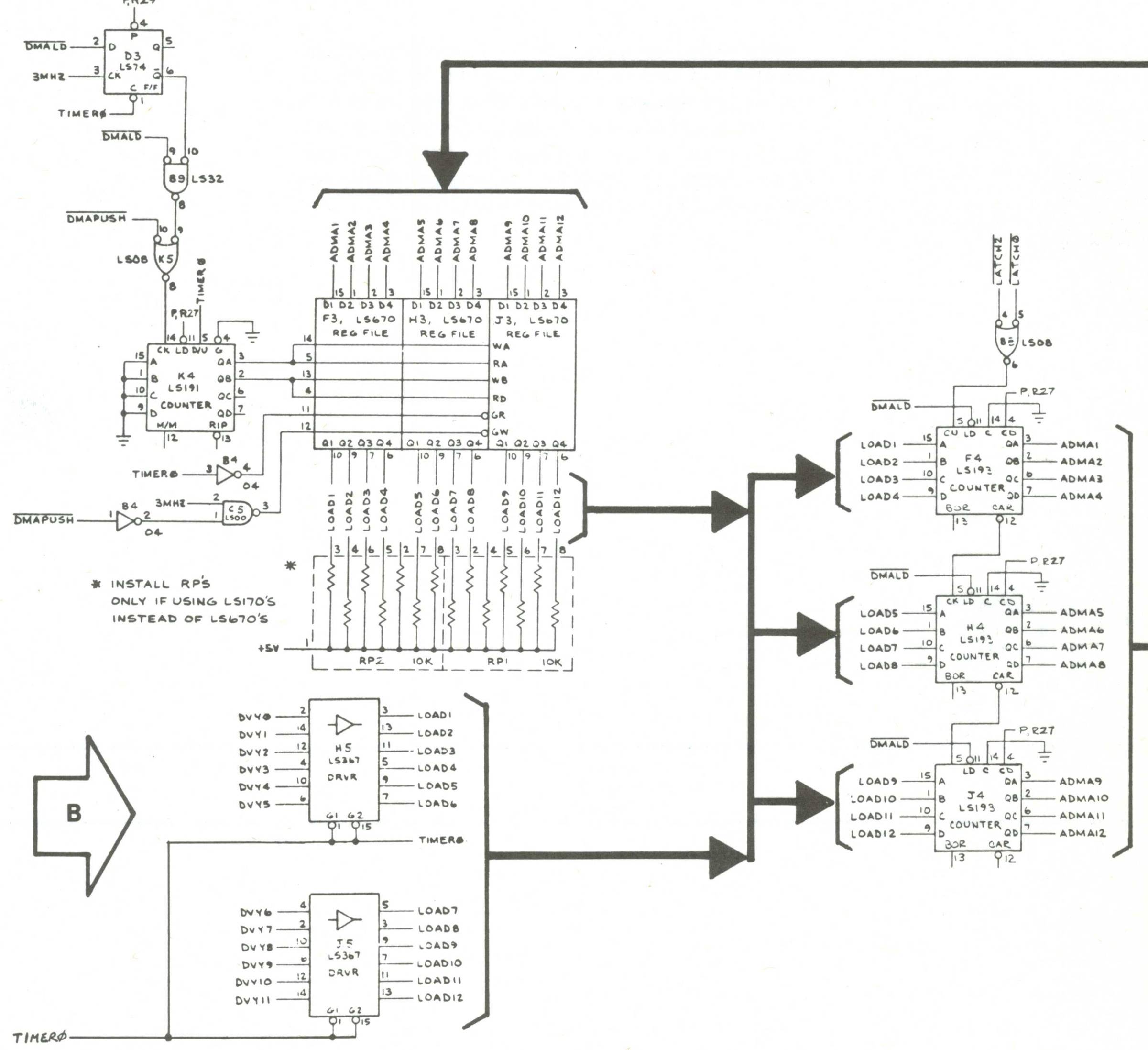
PROGRAM COUNTER

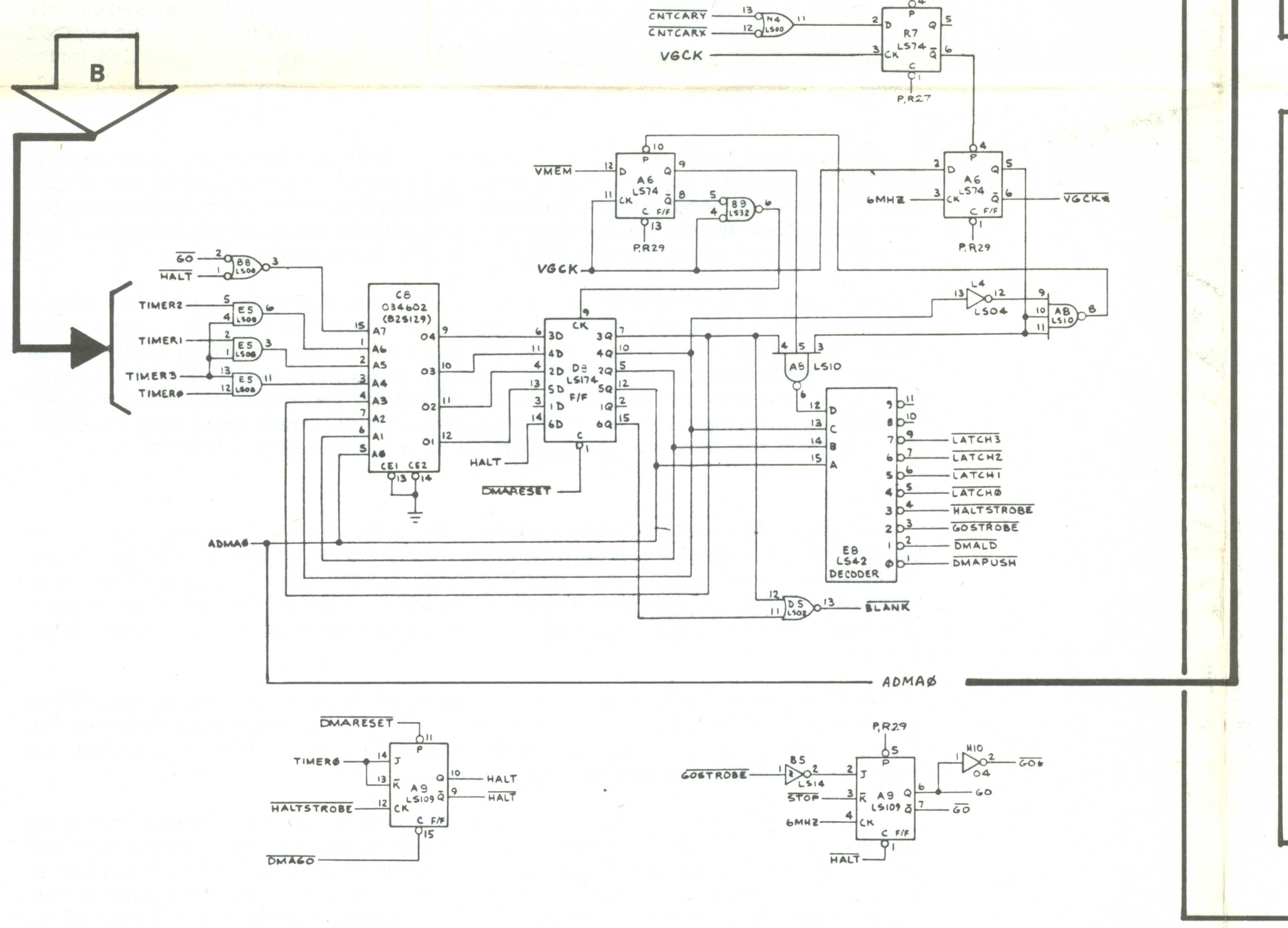


Counters F4, H4 and J4 contain the address of the next data byte (instruction) to be fetched from the Vector Generator memory. Because these counters point to the next instruction in memory to be retrieved and performed, they are called the program counter. This program counter is incremented one count (to the next sequential address) each time the information at its current address is loaded into data latch 0 or data latch 2.

The program counter may also be preset to "jump" to a new address. This new address can be loaded into the program counter from the vector generator memory via data latches F6 and H6 and buffers H5 and J5.

The program counter may also be preset to "return" to a previous address which it had stored in its "stack". The stack consists of register files F3, H3, & J3, and down/up counter K4. The stack is a 4-word 12-bit memory, used to save the contents of the program counter for future reference. It is loaded when DMAPUSH is low. Immediately after information is written into the stack, counter K4 increments one count. Immediately before loading the program counter from the stack, counter K5 decrements one count.

STATE MACHINE



The state machine is the "master controller" of the vector-generator circuitry. It receives instructions from the game MPU, via the vector generator RAM. It carries out these instructions by accessing the appropriate sections of the vector-generator ROM memory, using the vector-generator program counter to do so. The state machine reads the vector-generator ROM data (via Timer 0-3) and decodes this information to determine how it should use this data: 1) to draw a vector; 2) to move the monitor beam to a new position on the monitor display; 3) to "jump" to a new vector memory address; 4) to return to a previous vector memory address; or 5) to tell the game MPU that it has completed its current instructions, and is waiting for its next command.

The state machine consists of input gates B8 and E5, ROM C8, latch D8, clock circuitry A6, and decoder E8. Four-bit input TIMER0 thru TIMER3 is the operation-code input to the state machine. The A4 thru A6 address input to ROM C8 tells the ROM which instructions to perform. Address inputs A0 thru A3 from latch D8 tells the ROM which state was last performed. The address A7 input GO tells the ROM that the position counters are presently drawing a vector. The HALT input to A7 tells the ROM that the vector generator has completed its operations.

During initial power-up of the game, the HALT signal is preset low. The microcomputer reads the high HALT signal through its switch input port (sel/mux L10) on data line DB7. This tells the microcomputer that the vector generator is halted and waiting for an instruction. To ensure that the beam is off when the state machine is halted, the high HALT, clocked through latch D8, results in a low BLANK to the Z-axis output.

The microcomputer outputs an address that results in a DMAGO signal that causes HALT to go high, and clears the vector-generator data latches. This makes TIMER0 thru TIMER3 signals all low. The state machine now begins executing instructions, starting at vector memory location 0.

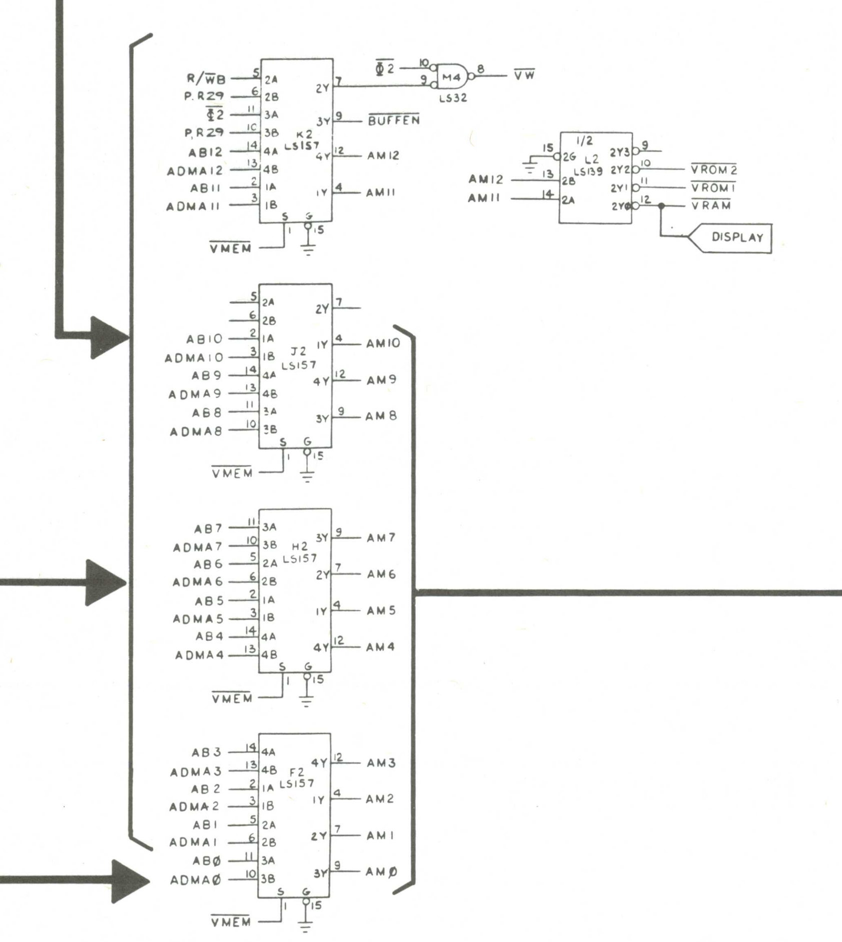
When the state machine receives the operation code for a HALT instruction, it outputs a low HALTSTROBE, setting the HALT flip-flop A9, and suspending state machine operation.

The GO signals load and enable the vector timer and the X and Y position counters and tell the ROM that the vector generator is now actively drawing a vector. The HALT input to GO flip-flop A9 sets the outputs to ensure that the vector timer and position counters are not active when the state machine is halted. When a low GOSTROBE is clocked through A9, the vector timer and X- and Y-position counters begin to operate from the GO, GO and GO* signals. When STOP is clocked through A9, the vector timer has reached its maximum count, and GO goes high. This means the vector has been drawn.

The VGCK input to the clock circuitry is a buffered 1.5-MHz clock signal from the microcomputer. This is the same frequency used to clock the MPU of the microcomputer. The signal clocks latch D8 unless the microcomputer is addressing the vector RAM or ROM memories (when VMEM goes low). Then the clock input to latch D8 goes high and stays high until VMEM goes high.

FROM MICROCOMPUTER SHEET 1, SIDE B

VECTOR GENERATOR MEMORY ADDRESS SELECTOR

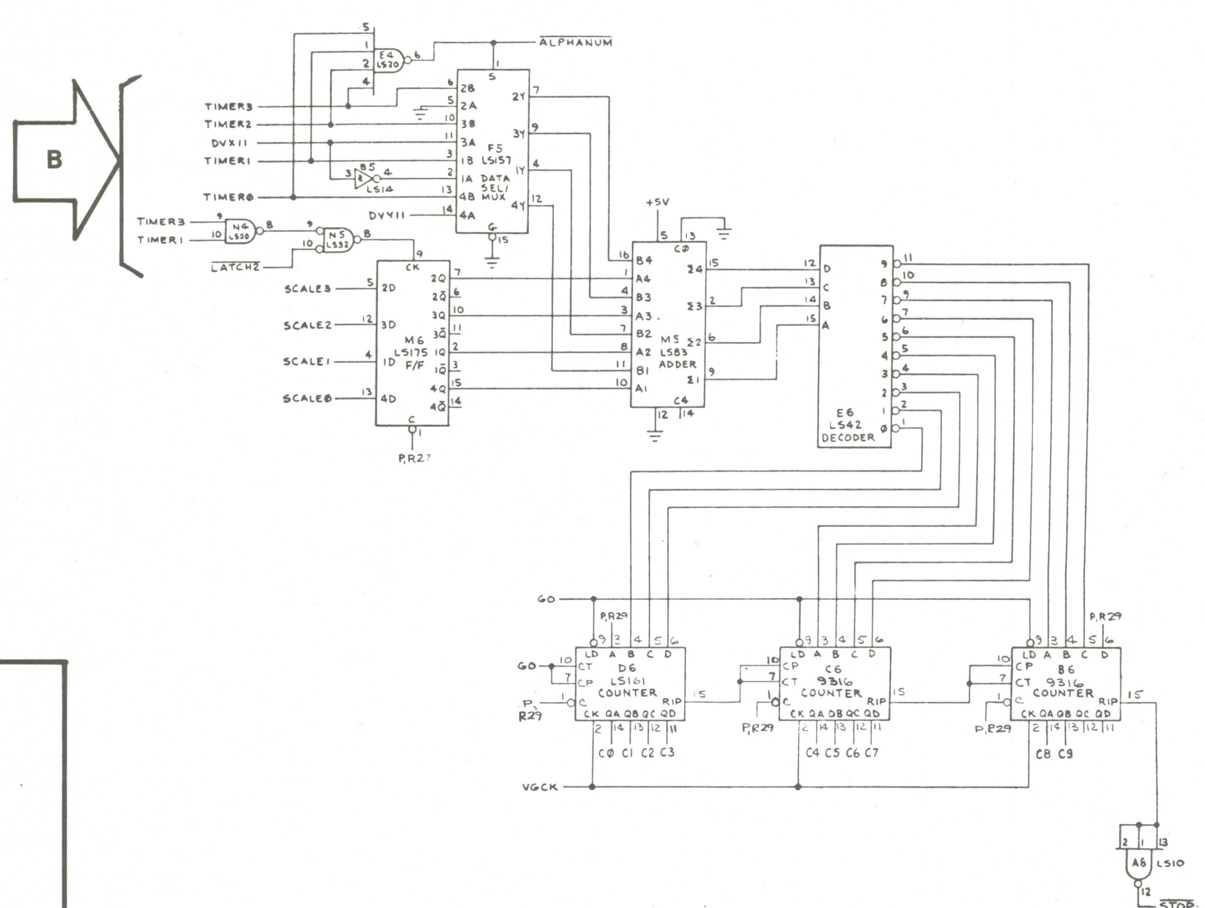


The address selector consists of multiplexers F2, H2, J2 and K2. When VMEM is low, the MPU of the microcomputer gains access to the address inputs of the vector generator memory. In this state, BUFFEN is from Φ2 and VW (vector generator write) is low when Φ2 and R/WB are both low. When VMEM is high, the address input to the vector generator memory is from the vector generator program counter and state machine. In this state, BUFFEN and VW are both held high by the pullup resistors connected to the 2B and 3B inputs of multiplexer K2.

Address decoder L2 decodes address bits A11 and A12, and selects the RAM or one of three ROMs of the vector-generator memory.

This address-selecting arrangement allows the game MPU to access the vector-generator memory, i.e., write data into the vector-generator RAM to instruct the vector generator what it should do next. The address selector can then allow the vector-generator program counter and state machine to access this same area of RAM also, and read what instructions were sent to it by the game MPU.

VECTOR TIMER

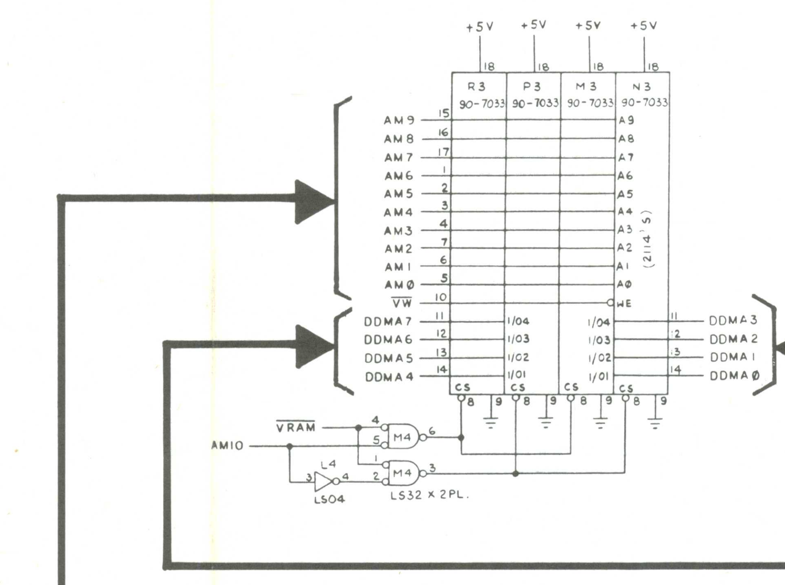


The purpose of the vector timer is to time out the length of time it takes to "draw" an actual vector on the monitor display. During the interval when the X- and Y-position counters are actually drawing the vector, STOP is high. This prevents the vector-generator state machine from advancing to its next state until the vector currently being drawn is completed. As soon as the vector has been drawn, STOP goes low, allowing the state machine to advance to the next state in its intended sequence.

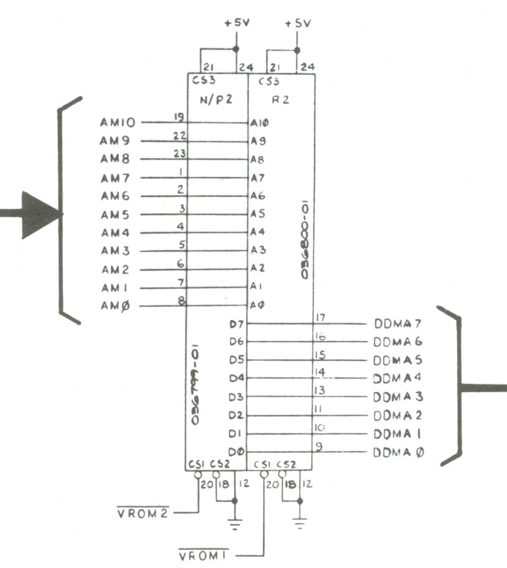
The vector timer consists of multiplexer F5, decoder E6, latch M6, adder M5, and counters B6, C6, and D6. M6 contains a scale factor which is added in M5 to the four timer signals. If TIMER0 thru TIMER3 inputs are any state but all high, decoder E6 directly decodes the sum and loads the decoded low into one of the counters. When GO goes low, the counters count from the loaded count until the counters all reach their maximum count. This count is a maximum length of 1024. At this time STOP goes low and clears the GO flip-flop of the state machine.

If the TIMER signals are all high, ALPHANUM goes low and data signals DVX11 and DVY11 are decoded by decoder E6. This is added to the scale factor and loaded into the counters.

VECTOR GENERATOR RAM

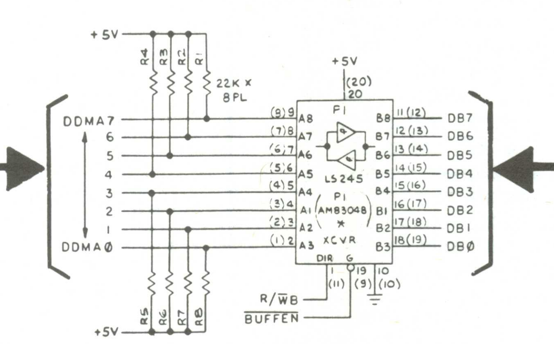


VECTOR GENERATOR ROM

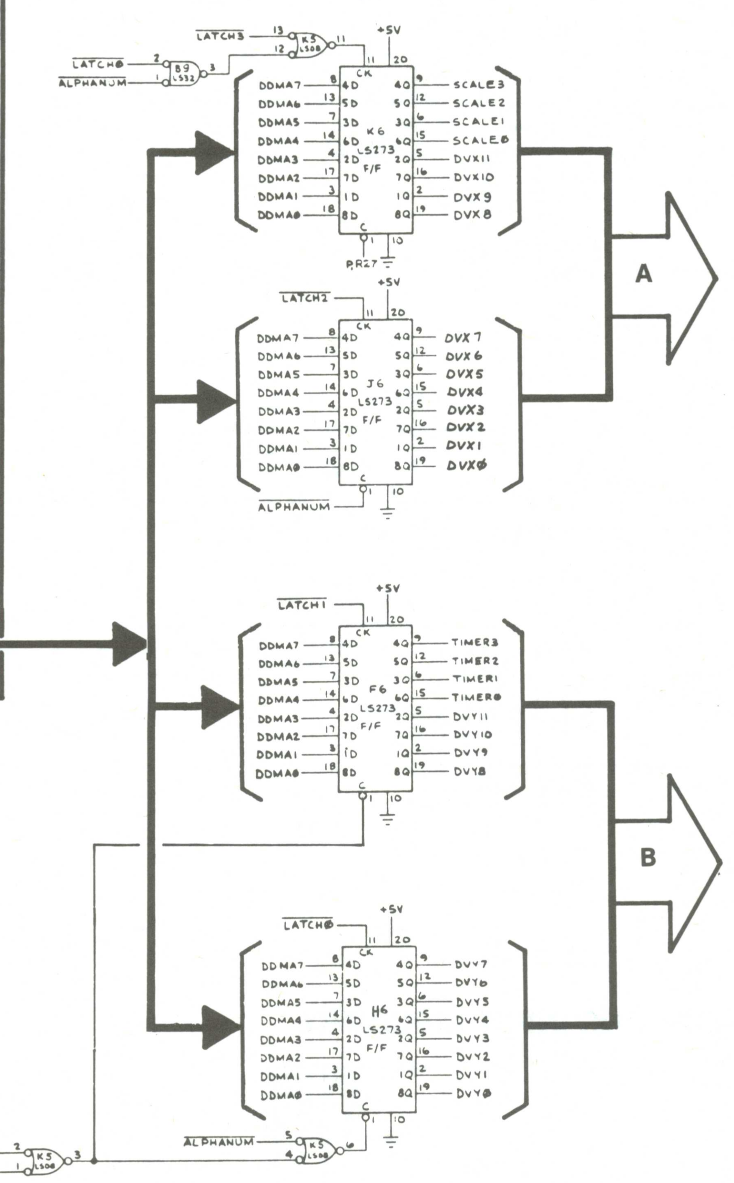


VECTOR GENERATOR DATA BUFFER

TO/FROM MPU DATA BUS SHEET 1, SIDE B



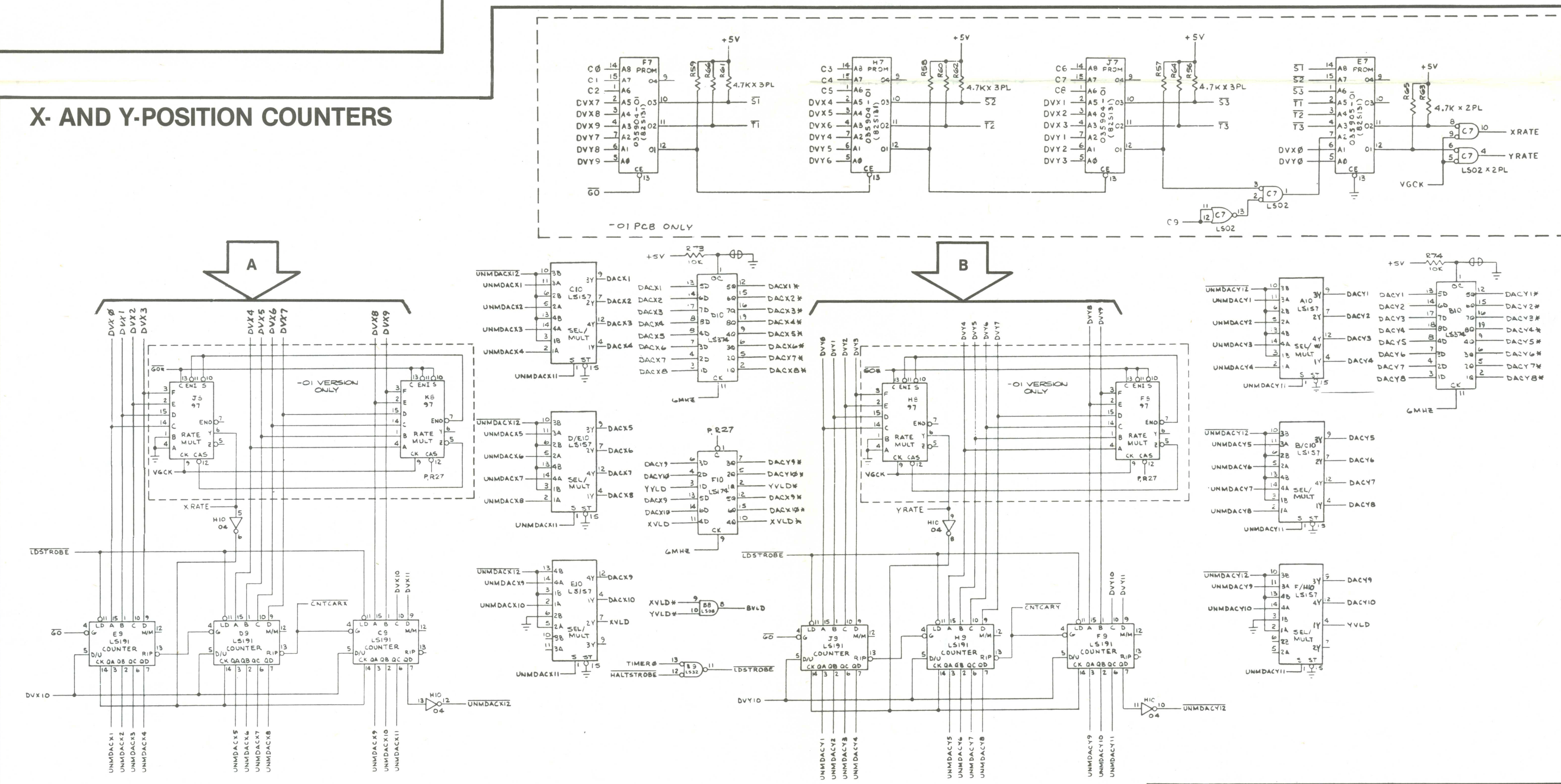
VECTOR GENERATOR MEMORY DATA LATCHES



The data latches consist of latch 0 (H6), latch 1 (F6), latch 2 (J6), and latch 3 (K6). Inputs DDMA0 thru DDMA7 are the data outputs from the vector-generator memory.

Latches 0 thru 2 are directly clocked by the rising edge of the LATCH0, LATCH1, and LATCH2 outputs from the vector generator's state machine. Latch 3 is clocked by LATCH3 or by LATCH0, if ALPHANUM is low. Latch 0 is cleared when RESET, DMAGO, or ALPHANUM goes low. Latch 1 is cleared by ALPHANUM.

X- AND Y-POSITION COUNTERS



The X- and Y-position counters are two identical circuits. Therefore, the following description discusses only the X-position counters.

The X-position counters contain rate multipliers (J8 and K8), down/up counters (C9, D9 and E9), multiplexers (C10, D/E10, E10), latch (F10), and associated gates (B8 and H10). The output of the down/up counters is a 12-bit binary number that represents the horizontal location of the beam on the monitor screen (or X axis), with 0 being the far left side of the screen and 1023 being the far right side of the screen. Increasing or decreasing this binary number output will cause the beam to move to the right or left, respectively. The vector generator state machine decodes instructions from its memory, and then is capable of using that data to alter the binary count of these counters in one of two ways.

The state machine can preset these counters to an entirely different number from their previous contents. This will cause the beam to "jump" to a new location on the monitor screen instantaneously, i.e., for drawing a new vector from a different starting position than where the previous vector ended. While the beam is "jumping" to this new position, the beam itself is turned off to prevent unwanted lines from appearing on the screen. To preset this new position into the counters, the state generator causes LDSTROBE to go low. At this time, a new 12-bit number (DVX0-11) is loaded into the counters from the vector generator memory data latches.

The state machine can also instruct these counters to count up or down any specific number of counts. This will cause the beam to move to the left or to the right a specific distance relative to where it was. During this beam movement, the beam is turned on with the desired intensity. This is the procedure used to draw a vector on the monitor screen. The direction (to the left or right) and length (0 to 1023) of the vector to be drawn relative to the beam's current position is determined by DVX0-11 (from the vector generator memory data latches). This data contains information that determines how many clock pulses the counters will receive and whether the counters will count up or down.

DVX0-9 memory data is loaded into rate multipliers J8 and K8. The function of these devices is to space the desired number of counter clock pulses at equal intervals over the time period that it will take to draw the desired vector. This insures that vectors of different lengths will still be displayed with the same relative beam intensity. DVX10 and 11 are loaded directly into the counters. DVX10 determines whether the counters count up or down. DVX11 determines the quadrant of the vector being drawn.

The UNMDACX1 thru UNMDACX10 (X-axis unmultiplexed digital-to-analog converter signals) are transferred and stored at the output of the multiplexers on each rising edge of the 6-MHz clock (from the microcomputer clock circuitry). The DACX1 thru DACX10 signals are sent to the digital-to-analog converters (DACs) in the X video output.

The DACX1 and DACX10 outputs represent the physical placement of the beam on the monitor. The far left of the monitor screen is 0, the center is 512, and the far right is 1023. Therefore, if the DACX1 thru DACX10 signal was greater than 1023, the monitor beam would go off the right side of the screen and start again on the left side of the screen, a "wraparound" condition. To prevent a wraparound, the multiplexers' select input from UNMDACX11 goes high when the count is greater than 1023 or less than 0. This selects UNMDACX12 to be output from the multiplexers to the DACs, forcing all zeros or all ones, and thus keeping the beam on the appropriate side on the screen, instead of allowing it to wraparound.

The XVLD and YVLD (X and Y valid) outputs from the X- and Y-position counter multiplexers are latched (F10) and gated together to enable the Z axis output, BVLD (beam valid).

ATARI®
A Warner Communications Company

Sheet 2, Side A

ASTEROIDS DELUXE™ Video Generator Section of 036471-01 and -02 B

DP-174-02 1st printing