B 9348-50
OPERATOR DISPLAY TERMINAL

TECHNICAL MANUAL
VOLUME 3:

THEORY OF OPERATION

BASIC PRINCIPLES

VIDEO AND DISPLAY

PROCESSOR AND STORAGE BOARD

SERIAL INPUT/OUTPUT BOARD

SWITCHING POWER SUPPLY

Burroughs

FIELD ENGINEERING

For Library Binder

Printed in U. S. America

June 1978

Form 1098399

NOTE

Information concerning the CDP1802D microprocessor chip has been included in this manual with permission of the Memory and Microprocessor Marketing Group of the RCA Solid State Division.

Correspondence regarding this document should be addressed directly to Burroughs Corporation, P.O. Box 4040, El Monte, California 91734, Attn: Publications Department, TIO–West.

LIST OF EFFECTIVE PAGES

TABLE OF CONTENTS

TABLE OF CONTENTS (Cont)

LIST OF ILLUSTRATIONS

SECTION 1
BASIC PRINCIPLES

GENERAL FUNCTIONAL OPERATION

The term B 9348 refers to a series of input and display terminals. The series of units are designated as styles B 9348-50, B 9348-51, B 9348-53, and B 9348-54. Refer to figure 1-1.

The B 9348 Terminal Input and Display unit is a free-standing, self contained, microprocessor based, input/output system. It can accept and display information entered locally from a keyboard or exchange and display information with a central processing unit.

C1000

Figure 1-1. B 9348 Input and Display Unit

Communications Procedures

The B 9348 uses standard Burroughs Communications procedures, such as Multipoint and Point-to-Point procedures.

Interface Description

The B 9348 style unit interfaces with the host system, transmitting in an asynchronous mode, using TDI (Burroughs Two-Wire Direct), or BDI (Burroughs Direct Connect) interface.

Optional Styles

The B 9348 series of terminals is one of the applications (or styles) that is planned for a basic set of hardware components. These basic components, referred to as the Modular Terminal System (MTS-1) hardware, acquire different application functions and capabilities each time a different set of masked Read Only Memory (ROM) is installed.

Internal MTS-1 Structure

Figure 1-2 provides an internal block diagram of the MTS-1 unit. As shown in the block diagram, the MTS-1 is made up of five printed circuit boards and a Cathode Ray Tube (CRT) for visual display purposes. The five boards are the Video board, the Processor and Storage (P/S) board, Serial Input/Output (SIO) board, Power Supply (PS) board, and the CRT Control or Monitor Electronic board. Refer to the following sub-sections for a general description of each board.

Processor and Storage Board

The Processor and Storage board contains the "Main" microprocessor MPU the Read-Only Memory (ROM), the Random Access Memory (RAM), and the Electrically Alterable Read-Only Memory (EAROM). The MPU on this board is referred to as the Main MPU because it provides the functional characteristics of the unit, controls the SIO and Data Communications interfaces, and provides the timing relationship used by the Video MPU in accessing the RAM through use of the interconnecting bus.

Video Board

The Video board contains the Display MPU, the video drive or display logic, and clock generation logic.

The clock oscillator output is gated to produce the display logic timing, the horizontal and vertical sync pulses, and the MPU clock. The MPU clock is used as master clock by the Video, the Processor and Storage, and the Serial Input/Output boards.

The Video MPU controls the display presentation. Its main function is to keep the character buffers full in the display logic.

The output of the video drive logic is a composite video signal that is fed to the CRT deflection and control logic on the CRT Control board. This composite signal contains character representation information, video blanking information, and horizontal and vertical sync pulses.

CRT Control Board

The CRT Control board contains the CRT Control circuits, Video Amplifier, Horizontal Control circuit, Vertical Control circuit, and High Voltage Supply. The purpose of the CRT Control board is to supply all the necessary controlling signals and voltages for proper operation of the CRT.

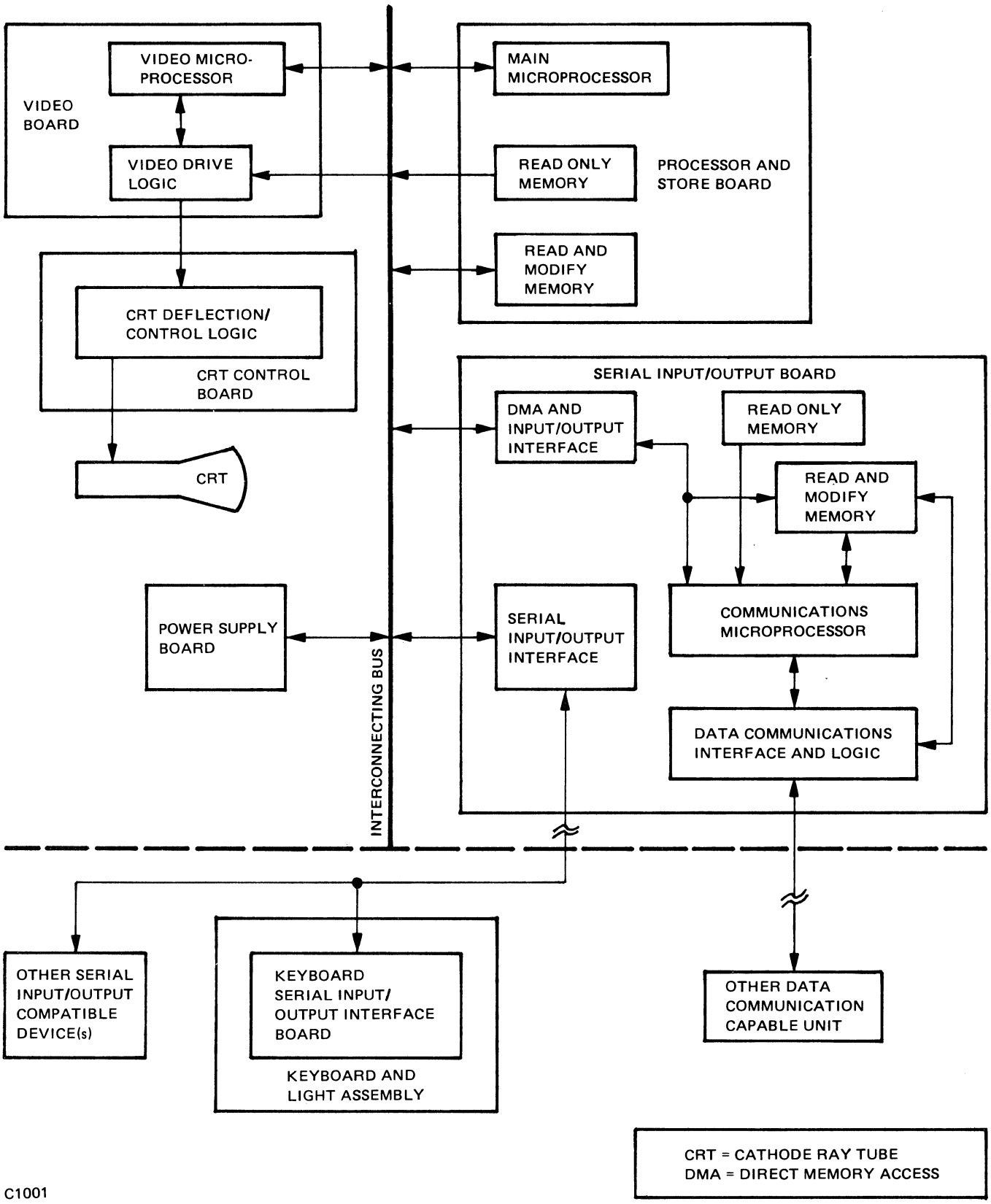


Figure 1-2. MTS-1 Functional Block Diagram

Serial Input/Output Board

The Serial Input/Output board contains both the Data Communications Control Logic and the Serial Input/Output Interface Logic.

The Serial Input/Output Interface controls the exchange of information between the MTS-1 and any terminals capable of using this interface. This logic is under the control of the Processor and Storage board microprocessor. The keyboard is the only device now using this interface.

The Data Communications Logic contains a third microprocessor. This Data Communications MPU monitors the communications interface and provides buffering between the Data Communications Line and the "Main" Microprocessor. The Data Communications MPU can communicate with the Main MPU, using an Input/Output interface or may take or store characters directly from the Processor and Storage (P/S) board RAM memory. The Data Communications MPU accesses the P/S RAM memory by using the Direct Memory Access (DMA) capabilities of the P/S MPU.

Power Supply Board

The Power Supply board provides the necessary voltages used on all the other logic boards except the CRT Control board. The CRT board uses the input voltage from the Power Supply board and develops the other voltages it needs.

SECTION 2
VIDEO AND DISPLAY

FUNCTIONAL OPERATION

This section describes the operation of the Video board and the Monitor. The Monitor is composed of Display board, Cathode Ray Tube, deflection coils, and the mounting frame.

DISPLAY DESCRIPTION

The MTS-1 display section is compatible with the display scheme used by a standard television receiver. The basic picture is divided into 525 vertical lines. Each of these vertical lines is divided into 780 horizontal Picture Elements (PELs). The combination of 525 lines by 780 PELs is referred to as the raster. The actual area of the picture used to display information is composed of 480 vertical lines each composed of 640 horizontal Picture Elements. (See figure 2-1.)

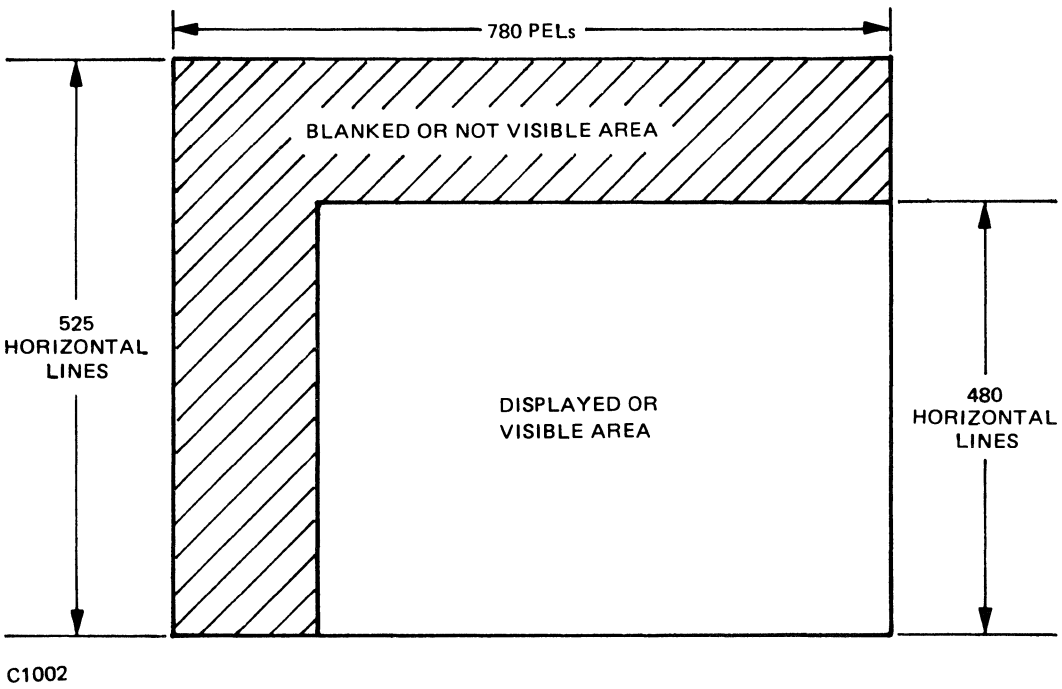


Figure 2-1. MTS-1 Raster Representation

This total picture of 525 lines by 780 PELs is presented to the viewer at a rate of 30 pictures per second. The human eye has a persistence of vision of 1/16 of a second. This means in order to produce a flicker-free image, it must be reconstructed at least 16 times per second. Even at this rate, some flicker will be noticeable. To overcome this indication, the standard television receiver reproduces the displayed picture 30 times in one second. The inside of the front of the Cathode Ray Tube or Picture Tube is coated with a phosphur or luminescent material. When this material is struck by an electron beam, it produces light of varying intensity, depending upon the strength of the electron beam. This beam is created from electrons leaving the cathode of the picture tube. Figure 2-2 illustrates a typical picture tube. The electrons are focused into a narrow beam. This beam is directed horizontally and vertically across the face of the picture tube to make up the horizontal lines. Figure 2-3 provides a sample of this result.

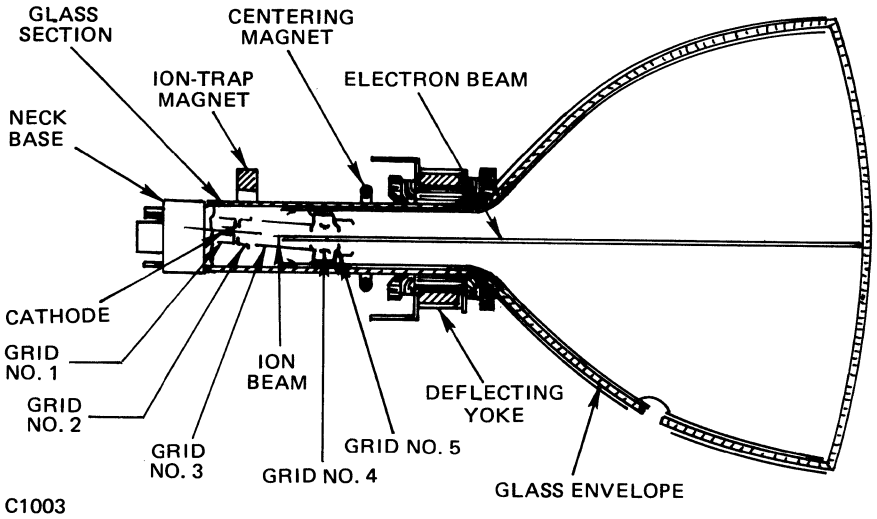


Figure 2-2. Typical Cathode Ray Tube

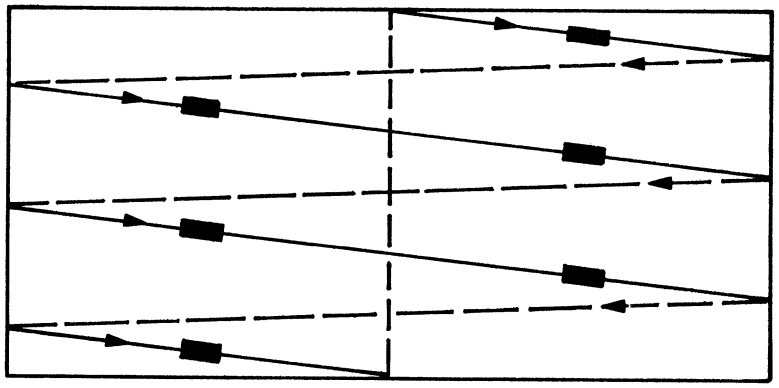


Figure 2-3. Typical Display Illustration

The solid lines in this figure represent the time the beam would be intensified to produce a picture representation and are called scan lines. The dashed lines represent the time the beam would be inhibited and no intensifying would take place. This is called retrace time. Note that the beam moves downward as it moves left to right across the picture. The dotted line is deflected only slightly. This is because the beam is deflected much more rapidly during retrace time than when the beam is scanning to produce an image. The retrace time is defined as the time when the beam is deflected from right to left to begin a new line or from bottom to top to begin a new field. Since these lines are very close together, if one were to view the darkened portion of the scan lines in figure 2-3, they would appear as two vertical bars.

The picture tube phosphor material does not have a very long persistence. If the picture is reproduced only every 1/30th of a second, the top of the picture would begin to fade. This fading would cause the image to flicker.

Interlaced Scanning

To reduce this flicker, the vertical deflection speed is doubled to produce an interlaced 525-line raster similar to the line raster shown in the "A" portion of figure 2-4.

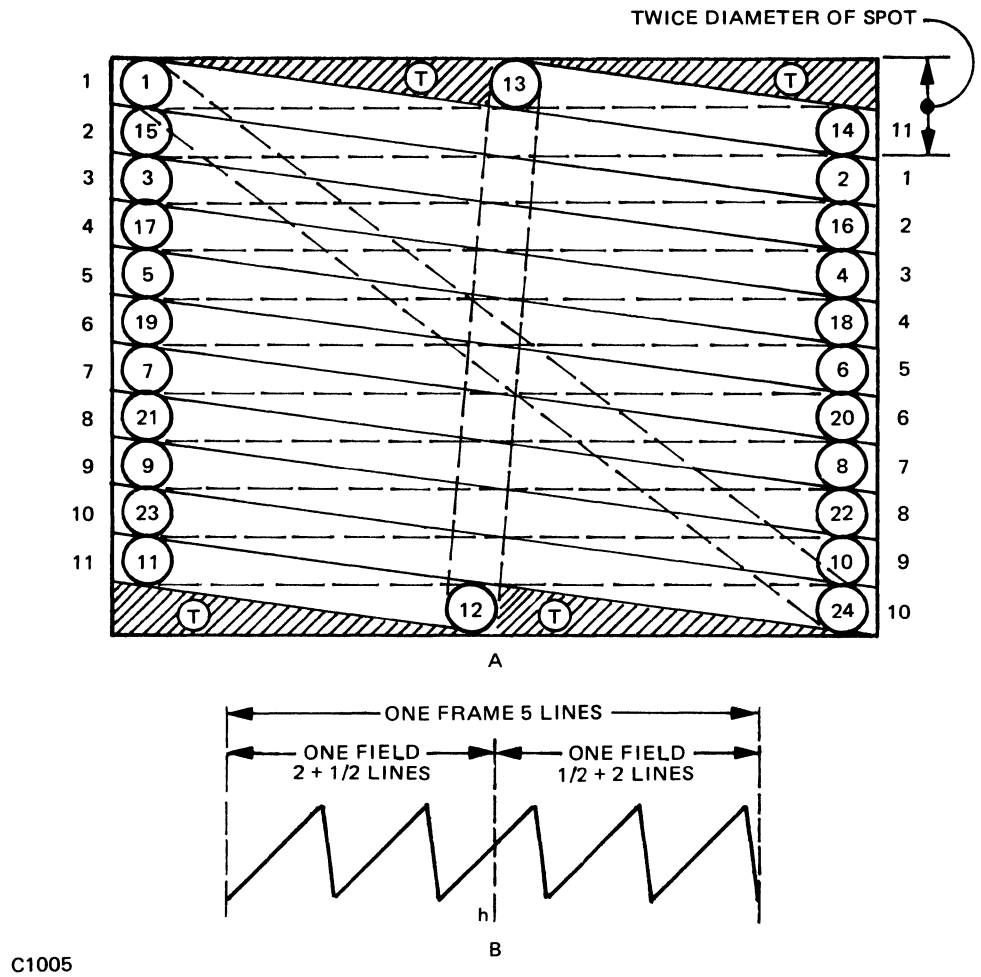


Figure 2-4. Frame and Field Representation

In this illustration, the beam (spot) is shown having a greatly exaggerated diameter. The various positions of the beam are indicated by the circles, such as those marked 1, 15, and 3, and so on, at the left edge of the figure. Those positions marked 14, 16, 2, and so on, are shown at the right edge of the figure. In addition, the 11 scanning lines are numbered from the top to the bottom of the raster on the outside edges of the illustration. The shaded T areas are the small parts of the screen not covered by the scanning beam.

NOTE

In the following discussion, the numbers in parentheses, for example, (1), represent the numbers shown within the circles of the "A" portion of figure 2-4. The numbers at the edges of the "A" portion of figure 2-4 represent the horizontal lines that comprise the raster.

Starting at position (1) in the upper left corner, the beam moves to position (2), tracing the top scanning line number (1). During this left-to-right movement, the beam travels downward also, for a distance equal to twice its diameter. Consequently, after retracing rapidly from position (2) to position (3), the beam is ready to trace scanning line number 3, to position (4), having skipped the scanning of line number 2. After moving to position (5), the spot scans lines 5, 7, and 9, and the first half of 11 in order, having skipped lines 4, 6, 8, and 10.

At the middle of line number 11, which is position (12), the beam has reached the end of vertical deflection and is suddenly vertically retraced to position (13), at the top, where it completes the tracing of the 11th scanning line. From position (13) to position (14), because the spot moves only half-way across the screen, it travels downward a distance equal only to its diameter. Therefore, retracing to the left side of the screen, it arrives in position (15), to begin scanning line number 2, which is in-between the lines scanned previously.

From position (15) to position (16), the spot traces line number 2 and again moves downward a distance equal to twice its diameter, so that retracing to position (17), it is ready to scan line 4. In a similar manner, lines 6, 8, and 10 are scanned, with lines 3, 5, 7, and 9 being skipped. When the end of line 10 at position (24) is reached, the beam is again at the end of vertical deflection and then retraces to position (1) in the upper left corner of the screen, ready to begin another frame.

Thus, in this interlace scanning arrangement, in order to trace all the lines of the raster once, the beam spot completes two vertical cycles of deflection. During the vertical scan "A" portion in figure 2-4, the beam traces 5-1/2 lines; on the second vertical cycle, the beam traces the other 5-1/2 lines, thereby completing one frame. Each vertical scan period is referred to as a field. Two fields are required to completely scan all the horizontal lines for one complete picture or frame. In the TD850, the first field is referred to as field 1 and the second field as field 0.

With this method of interlace scanning, as one set of lines of a field begins to diminish slightly in intensity, the second set of interlaced scanning lines is just being traced and will be at maximum brilliance. This action produces a more overall, evenly-illuminated frame, resulting in very little noticeable flicker.

The result of interlace action is obtained only if an odd number of scanning lines is used in the raster structure. The "B" portion of figure 2-4, composed of five horizontal sawtooth deflection cycles, represents the time for one frame in odd lines. Dividing line h represents the bottom of one field and the top of the next. Splitting the odd number 5 in two at point h of the figure will leave the half-line condition as shown. Whether that odd number is 5 or 525, the half line condition will always occur at the split.

Since dividing an odd number of lines by 2 will always result in a half line being left over, the beam will end at, and start from, a different point on the raster for each successive vertical scan or field (as shown in figure 2-4). An even number of lines divided into two fields would always have the beam finish and start a full line. Note that the bottom and top limits of the raster are the same for both fields, as shown in the "A" portion of figure 2-4. An even-numbered line raster arrangement, therefore, would make interlace action impossible; the fields would overlap rather than interlace.

Accordingly, the television system designates 525 lines per frame, interlaced 2 to 1, in successive fields. Approximately 60 fields are scanned in one second. This produces a flicker frequency of approximately 60 times per second, and requires the beam to travel from the top to the bottom of the raster in about 1/60 of a second. Since it takes two fields to make up one frame, 60 divided by 2 equals 30 frames per second.

In one vertical scan or field, 525 divided by 2 (or 262-1/2 horizontal lines) are traced in 1/60 of a second. In the next field, the in-between 262-1/2 horizontal lines are scanned. This completes a 525-line frame in 2/60, or 1/30 of a second, and is similar to the 11-line arrangement of the "A" portion of figure 2-4 or the 5-line time sketch of the "B" portion of figure 2-4.

Scanning Frequencies

Since 262-1/2 raster lines are scanned every field, with 60 fields per second there are 60 times 262-1/2 (or 15,750) horizontal lines traced each second. This means the frequency of the horizontal deflection system must be 15,750 Hertz per second, while the frequency of the vertical deflection system is 60 Hertz per second.

Composite Video Signal

The Video board produces a combination of synchronizing pulses and picture signal variations. This combination signal is called the Composite Video Signal and is used by the Display board to develop the necessary signals for beam deflection and intensity. Figure 2-5 illustrates a Composite Video Signal of one horizontal line, with relative voltage levels for that signal.

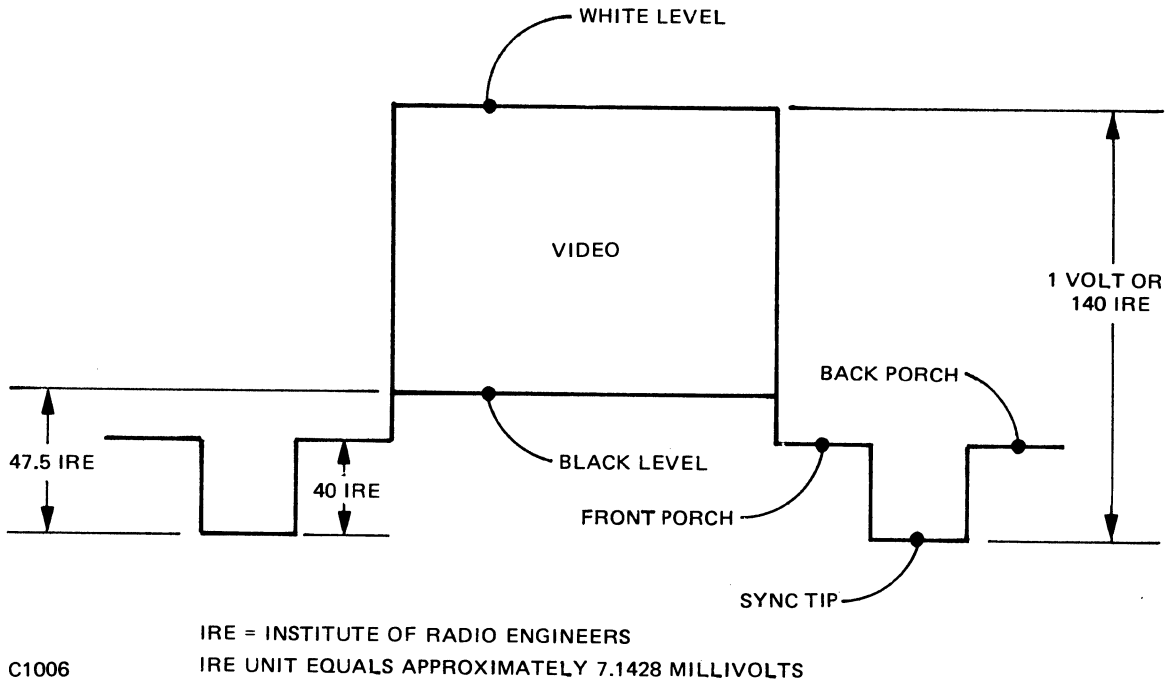


Figure 2-5. Composite Video Signal

The total signal amplitude is 1 volt in amplitude. This is divided into 140 IRE (Institute of Radio Engineers) units. Figure 2-5 describes standard video as opposed to negative, or inverted video. The Sync Tip (Horizontal Oscillator Synchronizing signal) is shown between two areas described as the "Front Porch" or "Back Porch." The "White" level is defined as the brightest intensity of a picture, and is the 1 volt or 140 IRE unit level of the signal. The "Black" level defines the darkest intensity of the picture and is the point the electron beam is cut off. The "Black" level is defined as being at the 47.5 IRE unit, or approximately 336 millivolt level. The "Front" and "Back" porches occur at the 40 IRE unit or approximately 280 millivolt level. The time represented by the Front Porch, Synch Tip, and Back Porch is the time electron beam is retracing to begin another scan line. Since all this area is below the Black Level, the movement will not be visible on the screen.

The MTS-1 divides the voltage between the Black and White levels into 16 levels. Refer to figure 2-6.

Note that less voltage difference is needed as the White level is approached in order to cause an equivalent brightness change from the last level of grey displayed.

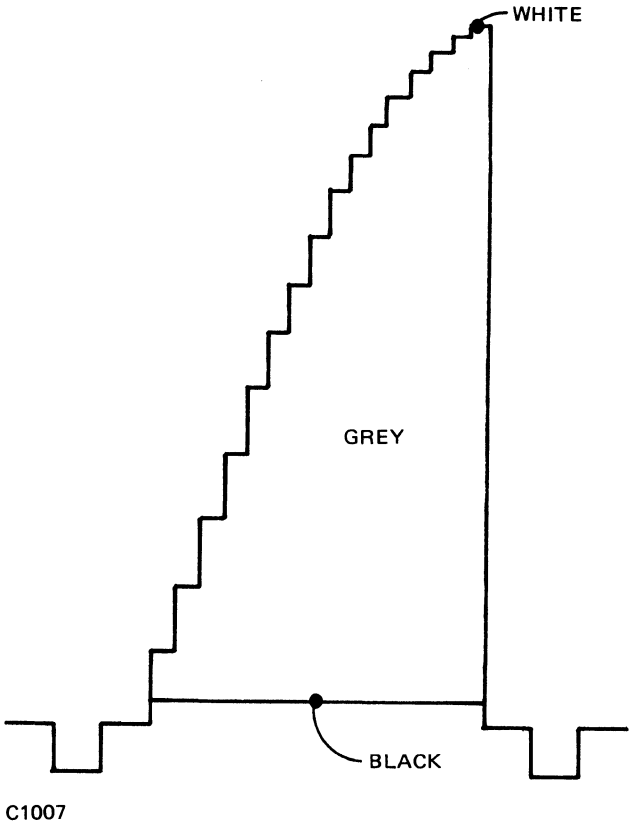


Figure 2-6. Grey Level Representation

Character Representation

A character space is defined in terms of horizontal Picture Elements (PELs) and scan lines. (See figure 2-7.) The displayable character space is eight horizontal PELs wide by 16 scan lines high. The sixteen scan lines are referred to as being eight scan line pairs. (Refer to Scan Line Counter and Decoder.) These eight pairs of scan lines are represented by Vertical Counter counts of 248 through 255. By adding the field designation, 16 scan lines are represented. If the character is displayed with an underline, the underline occurs in scan lines represented by field 0 of Vertical Counter count 254 and field 0 and 1 of Vertical Counter count 255.

To increase the vertical spacing between the displayed characters on successive display lines, the Vertical Counter can be preset to count up to eight additional scan line pairs. Changing the Vertical Counter preset will provide up to an additional 16 scan lines of vertical spacing between successive displayed character lines.

The visible displayed character cannot be increased in vertical size. Since the visible portion of the horizontal scan line is 640 PELs, there can be 80 standard or 40 wide characters. A wide character takes up two standard character widths or 16 horizontal PELs. The displayed wide character is the same vertical size; but is twice the width of a standard character.

The 480 displayable scan lines result in a maximum display capability of 30 character lines of 8 scan line pairs. If the maximum spacing of 16 scan lines is used, the maximum display capability is 15 character lines of 16 scan line pairs. The MTS-1 uses 24 character lines for operator data display and one for the display of status information.

VERTICAL COUNTER		FIELD	CHARACTER
DEC.	BINARY		
240	1111 0000	1	
		0	
241	1111 0001	1	
		0	
242	1111 0010	1	
		0	
243	1111 0011	1	
		0	
244	1111 0100	1	
		0	
245	1111 0101	1	
		0	
246	1111 0110	1	
		0	
247	1111 0111	1	
		0	
248	1111 1000	1	
		0	
249	1111 1001	1	
		0	
250	1111 1010	1	
		0	
251	1111 1011	1	
		0	
252	1111 1100	1	
		0	
253	1111 1101	1	
		0	
254	1111 1110	1	
		0	
255	1111 1111	1	
		0	

C1008

Figure 2-7. Character Representation

Video Board

Figures 2-8 and 2-9 show a block diagram of the sections of the Video Board. This board is grouped into two major sections. The first section is referred to as the Control Logic. The Control Logic controls the overall operation of the board. The second section controls the character conversion for input to the Display board as a composite video signal.

Timing Generation

Figure 2-10 is a block diagram of the Timing Generation logic. This logic develops master clock, horizontal synchronizing and blanking signals, and the vertical clock pulses. The frequency of the clock oscillator is 12.272726 megahertz. Its output, called BITCLOCK, is 40.7 nanoseconds wide and occurs every 81.48 nanoseconds. BITCLOCK is fed into a Master Clock Divider circuit. This circuit is a counter composed of

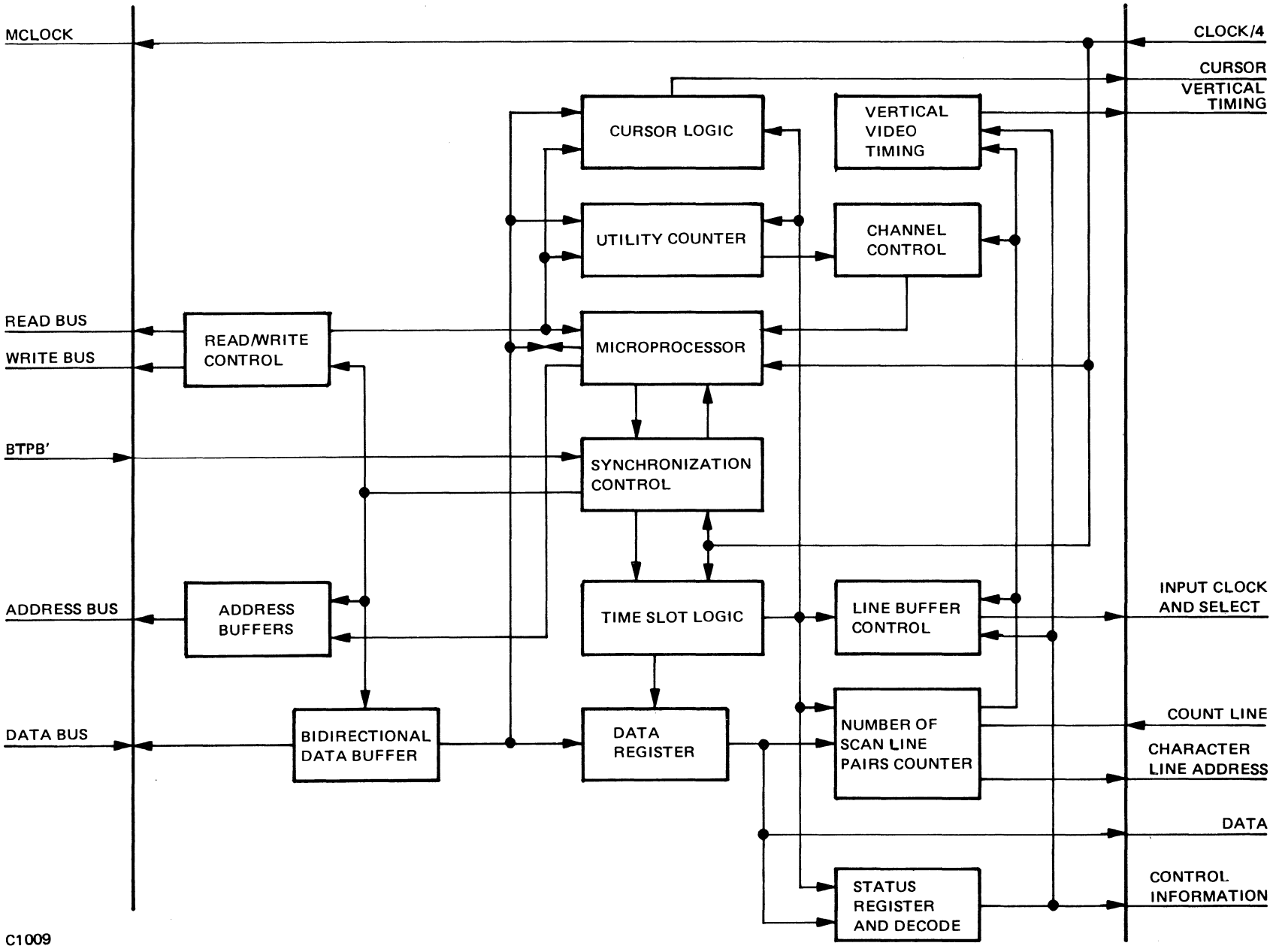


Figure 2-8. Video Board Block Diagram - Control Section

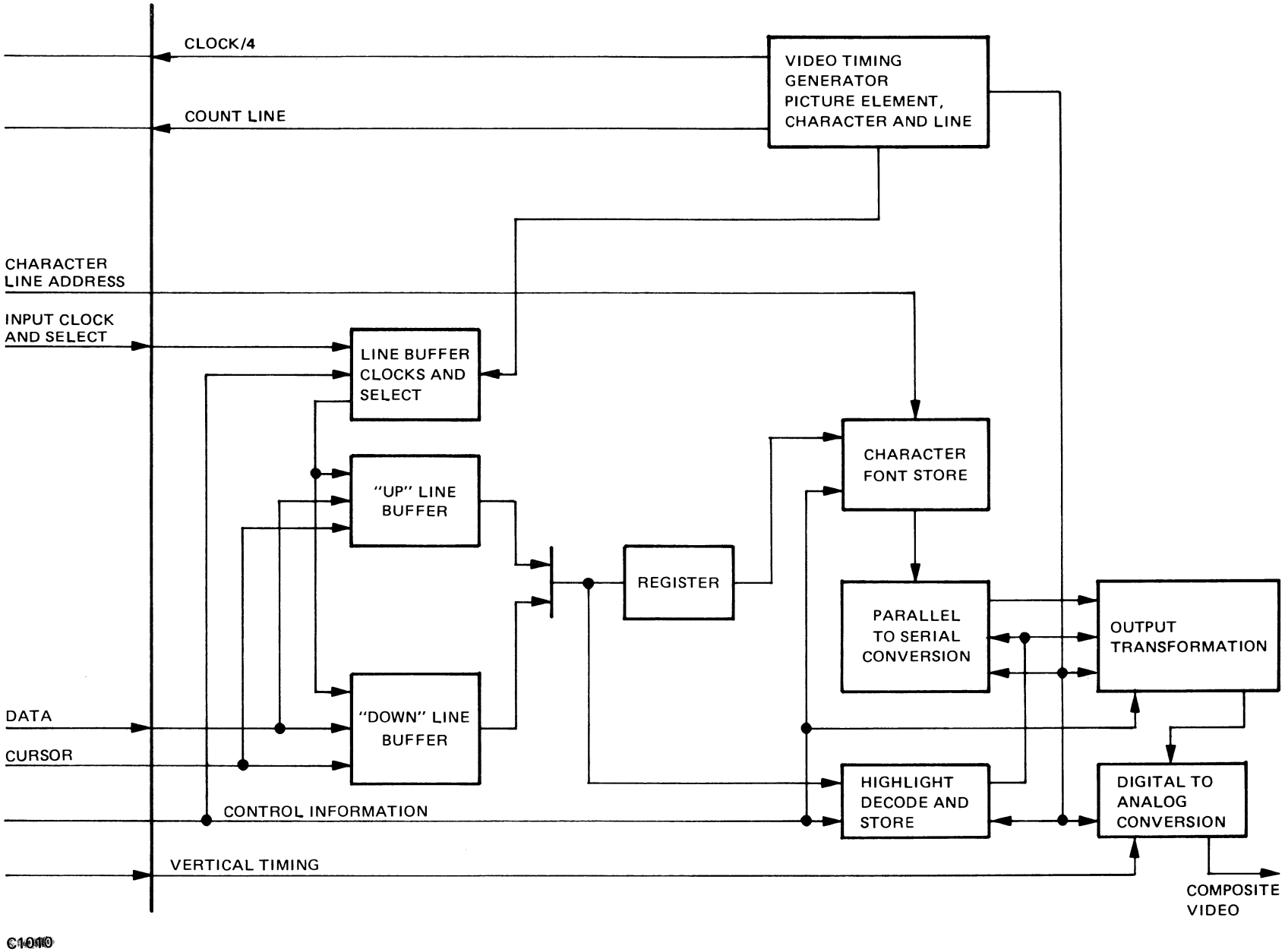


Figure 2-9. Video Board Block Diagram - Output Section

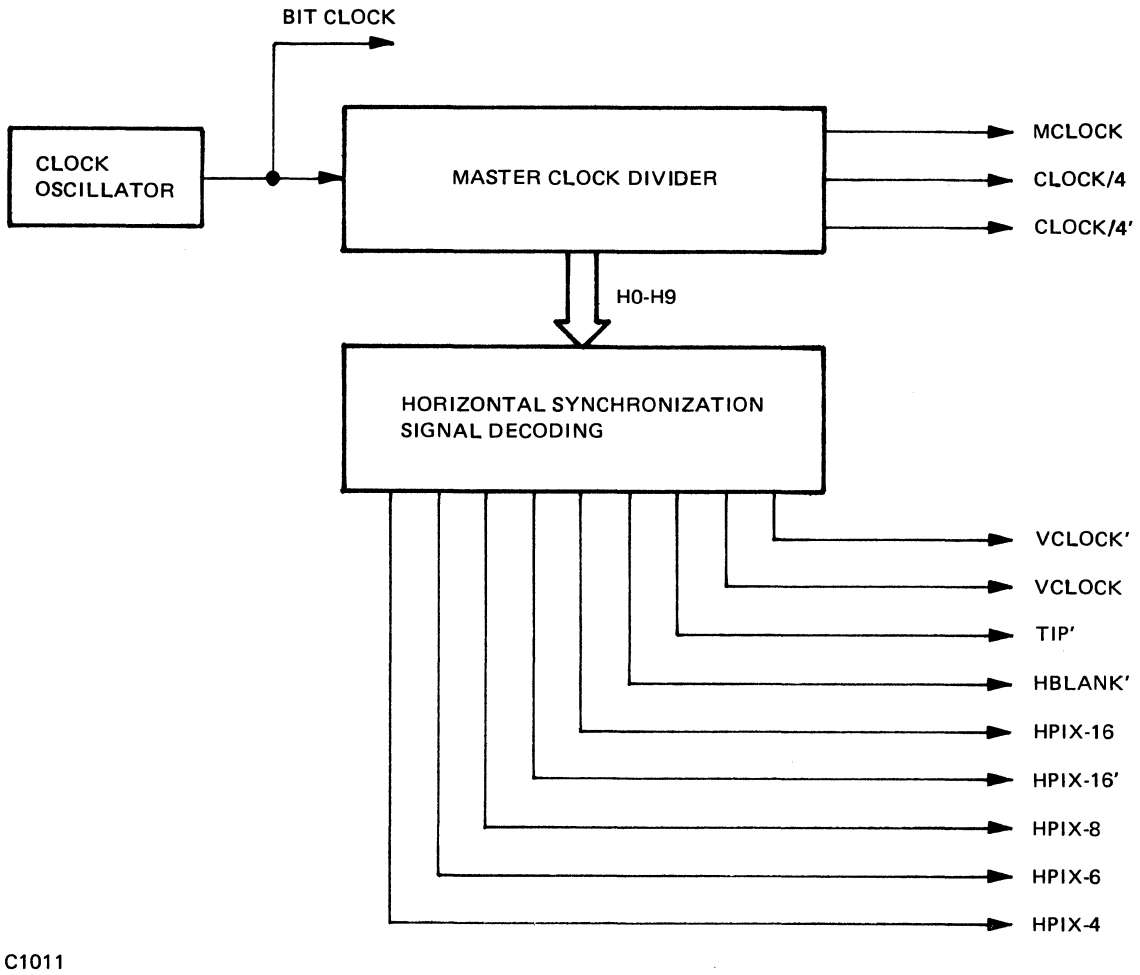


Figure 2-10. Video Clock and Decode Block Diagram

flip-flops designated as H0 through H9. The counter resets every 780 bit clocks. Each bit clock time is equivalent to one horizontal Picture Element (PEL). The 780-bit clock count is equivalent to one horizontal scan line.

Other outputs are MCLOCK, CLOCK/4, and CLOCK/4'. These signals all have a pulse width of approximately 163 nanoseconds and a repetition rate of approximately 326 nanoseconds. MCLOCK is routed to the backplane and becomes the master clock for the timing of actions on the other logic boards in the MTS-1. CLOCK/4 and its inverse signal CLOCK/4' are used for timing actions on the Video board. These signals provide a common clock synchronization for the MTS-1 logic.

Horizontal Timing Generation

H0 through H9 are fed to the Horizontal Synchronization Signal Decoding logic. The signals from this logic are used for horizontal synchronization, horizontal blanking, character decoding synchronization, and for clocking the vertical synchronization logic. See figure 2-11 A and 2-11B for a timing diagram of this operation and the relationship of the horizontal timing and character shift signals.

Vertical Timing Generation

This logic generates the Vertical Synchronization signal and the Vertical Picture Time signal, VPIX. The Vertical Synchronizing signal is used to reset the vertical oscillator on the Display board, thus starting the vertical retrace. The Vertical Picture Time signal is used to enable the display or disable it during the vertical retrace time.

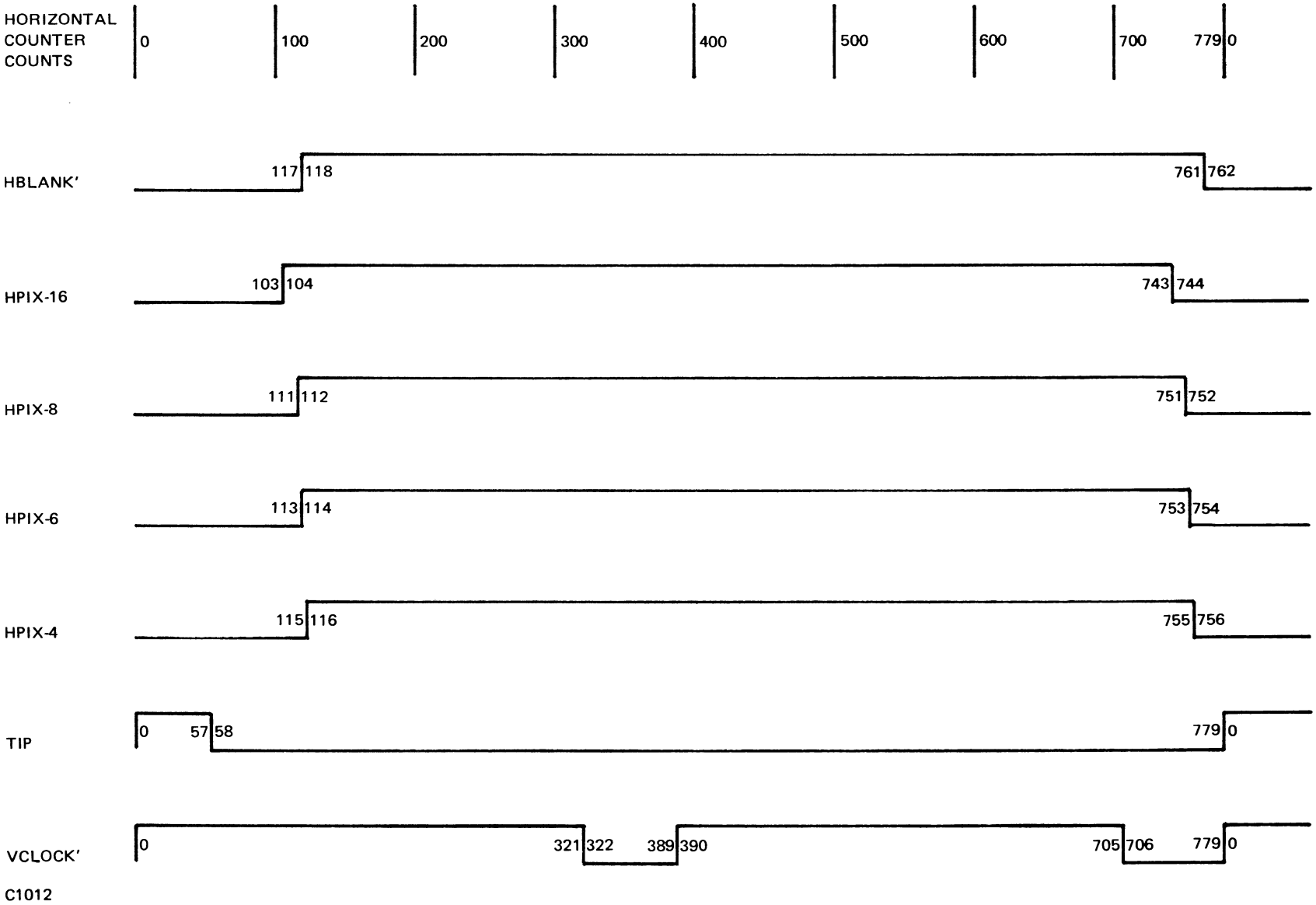


Figure 2-11A. Horizontal Synchronizing Signals

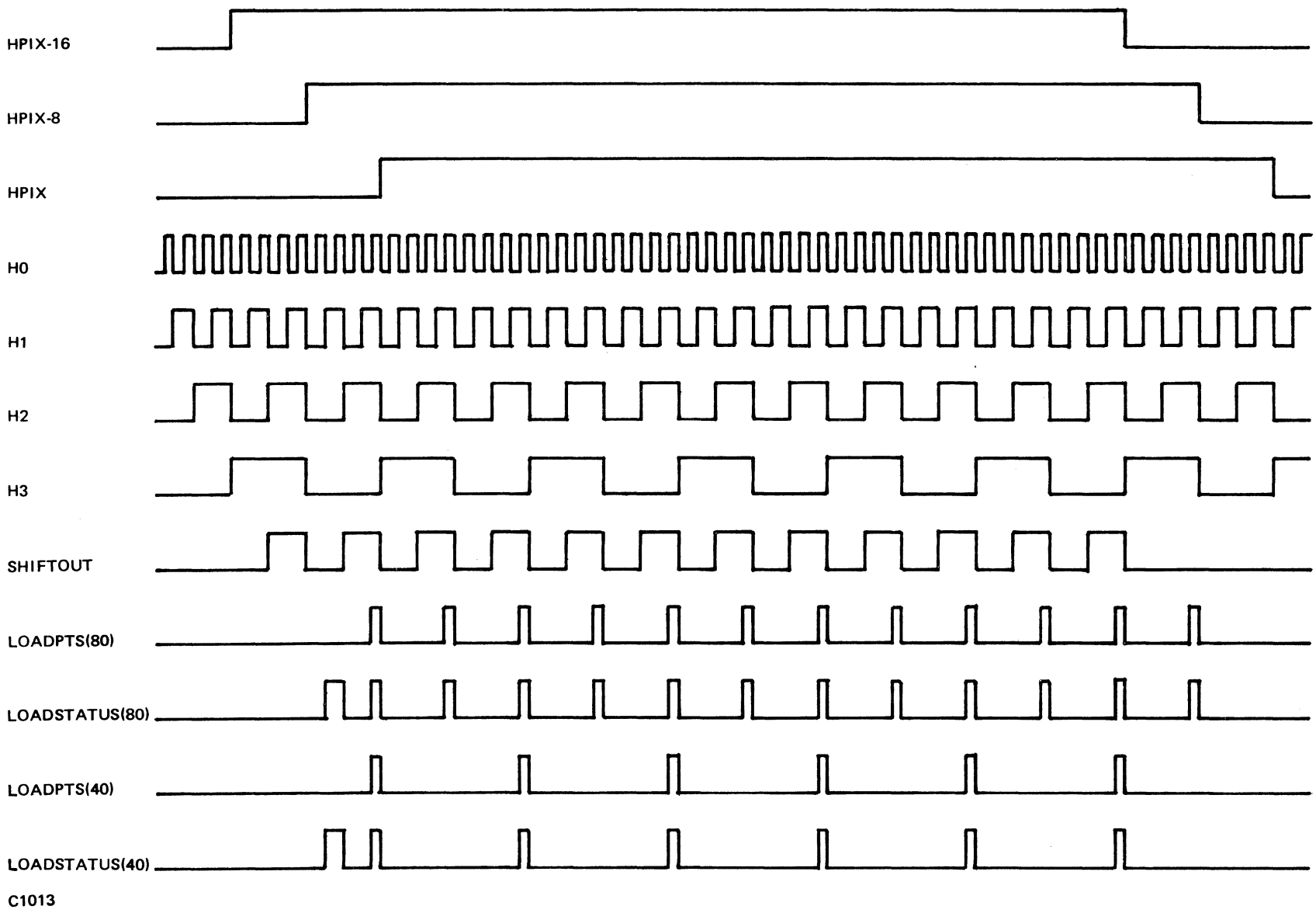


Figure 2-11B. Horizontal Timing and Character Shift Signals

The vertical synchronization logic is made up of a 4-bit counter and a flip-flop that decodes the counter's output. The counter is preset to a count of 0 or 1, depending upon the state of the Field flip-flop. This flip-flop is set during the time the first field of the picture is being presented and reset for the second field. When the display of the last scan line of the last character line occurs, the counter will then preset with the next VCLOCK signal. This signal occurs twice during one horizontal scan line. The decoded output of the counter occurs for counts 8 through 14. This corresponds to three horizontal line time periods. The pulse width is approximately 190.71 microseconds and occurs approximately every 16.687 milliseconds (or approximately 60 Hertz per second).

Presetting the Vertical Synchronization counter to 0 or 1 causes the Vertical Synchronization pulse to occur at the end or in the middle of a horizontal line. This action causes the interlacing of the picture fields. The counter is inhibited from counting until the next preset occurs, because the carry output of the counter is used to disable the counter.

Vertical blanking of the display is controlled by the signal VPIX. VPIX goes high at the beginning of the first displayable horizontal scan line at the top of the display. VPIX goes low to disable the display after the last scan line of the last character line of the display. The length of time (number of horizontal scan line periods) that it is low is controlled by the count placed in the Scan Line Pairs counter. If the Field flip-flop is 1, the count is 23. If the Field flip-flop is 0, the count is 22. See figure 2-12 for a timing diagram of this operation.

Microprocessor Synchronization

The MTS-1 has three microprocessors. Since two microprocessors can access memory on alternate memory cycles, it is necessary to ensure they are not requesting information at the same time. One microprocessor cycle is eight clock times in duration. These clock times for the Display microprocessor are designated as timing periods S0 through S7 and T0 through T7 for the Processor and Storage microprocessor. The clock used by the microprocessor chips is MCLOCK. (Refer to the Timing Generation heading in this section.) The microprocessors use data from memory during timing periods 4 through 7. To ensure there is no conflict, S0 through S3 of the Display microprocessor must occur during timing periods T4 through T7 of the Processor and Storage microprocessor.

This synchronization is achieved through the use of logic that compares timing pulses (TPB) from the Display microprocessor and from the Processor and Storage microprocessor. The results of this comparison are used to start and stop the Display microprocessor until the proper synchronization occurs. Figure 2-13 provides a simplified block diagram of synchronization and memory control.

Figure 2-14 illustrates logic operation for synchronizing two microprocessors. In this figure, "state" is defined to be the relative logic count of the synchronizing logic. TPB and BTPB occur at timing period 7 of the Display microprocessor and of the Processor and Storage microprocessor, respectively. Assume any binary value 00 through 11 for TPB and BTPB. Also assume any timing period from S0 through S7 for the Display microprocessor and T0 through T7 for the Processor and Storage microprocessor. The Processor and Storage microprocessor timing periods progress from T0 through T7 with each clock input. The Display microprocessor timing periods progress to the next timing period only if WAIT' is high and a clock pulse occurs. In this figure, the correct cycle through the state counts is 8, 9, 10, 11, 12, 13, 14, 15, and then back to 8. When this cycle occurs, the Processor and Storage microprocessor timing period T7 will be coincident with the Display microprocessor timing period S3; the Processor and Storage microprocessor timing period T3 will be coincident with the Display microprocessor timing period S7. The State logic also produces signals to enable the Address bus, Data bus, and the Memory Control signals.

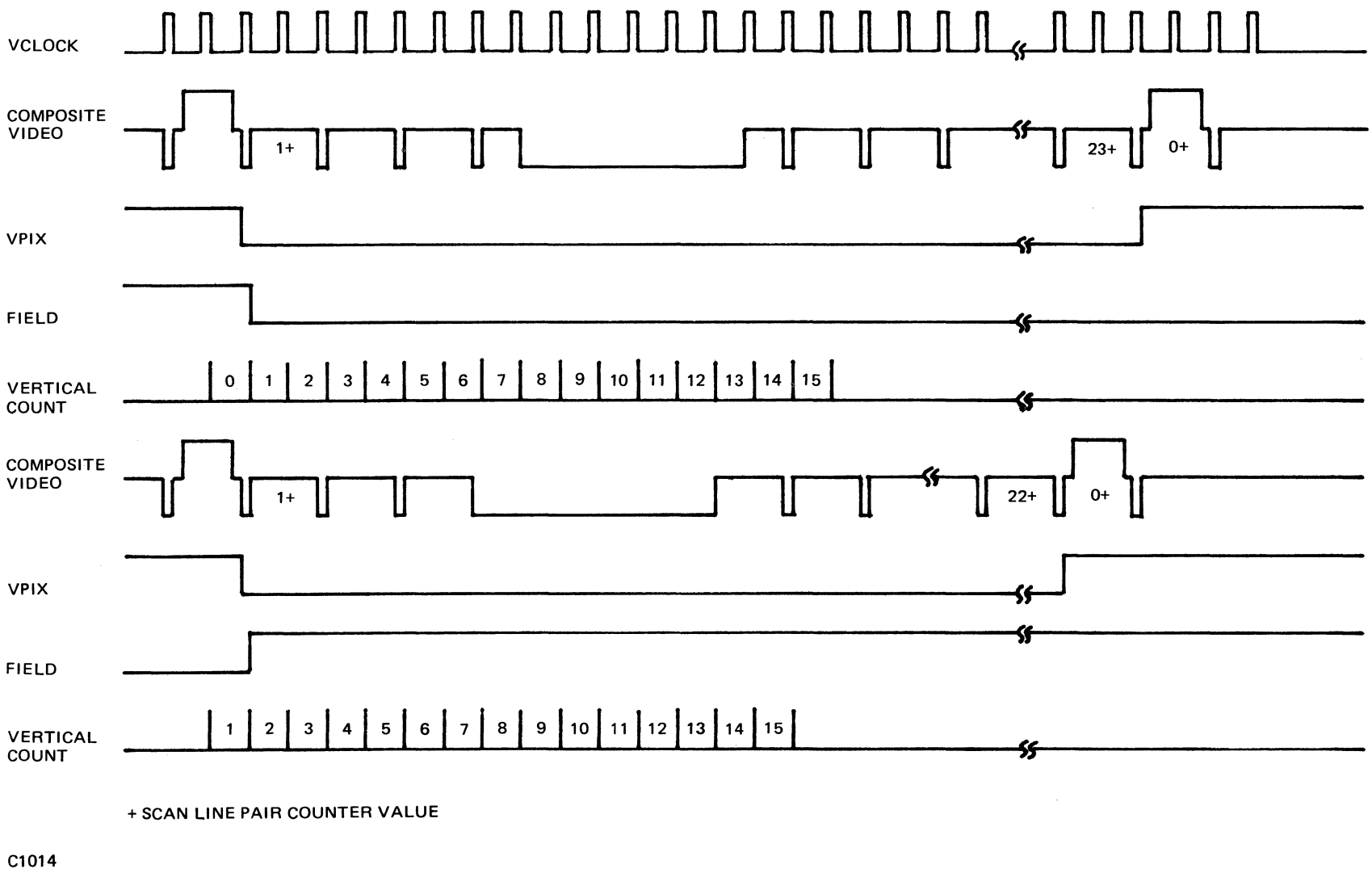


Figure 2-12. Vertical Synchronizing Timing

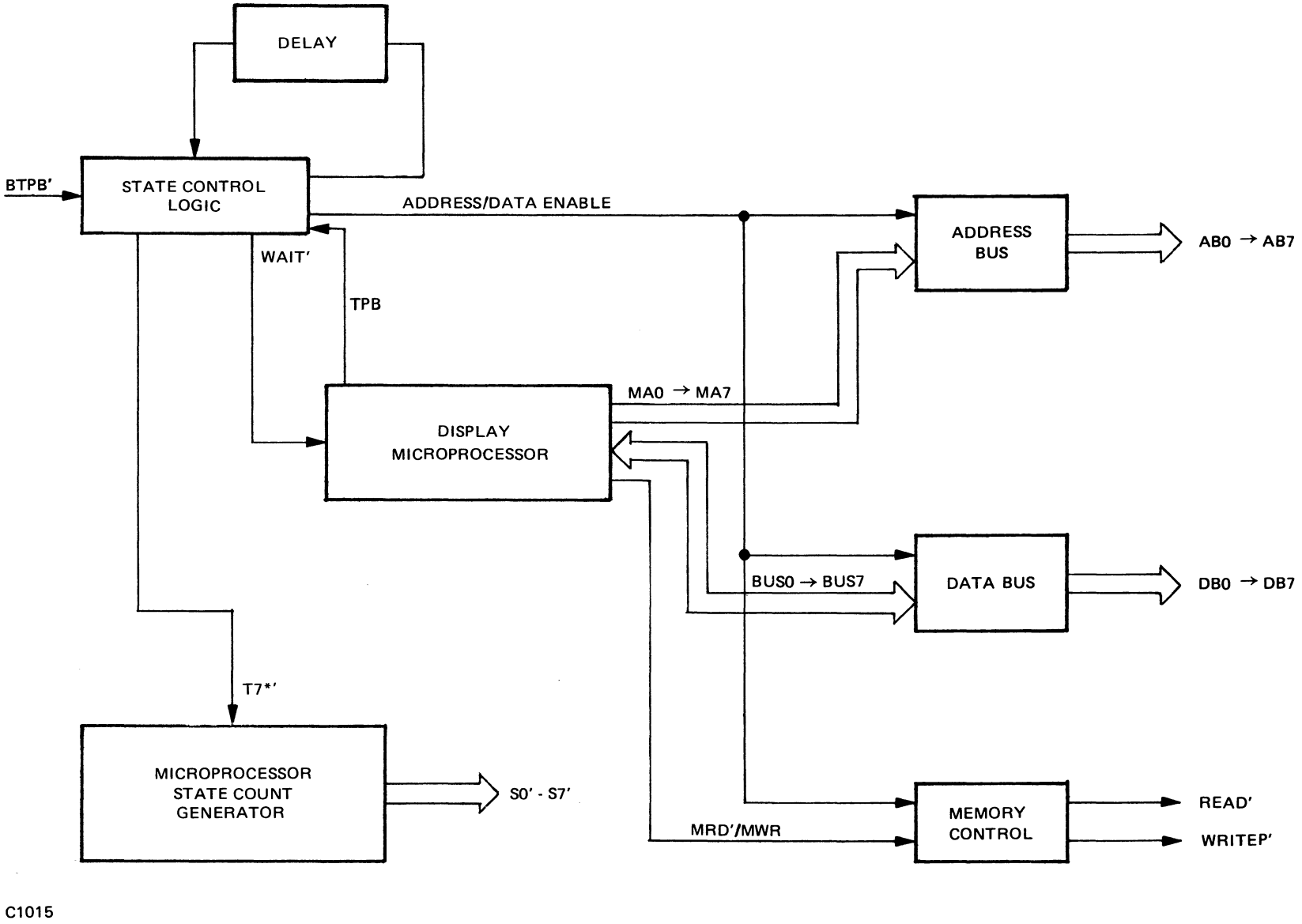


Figure 2-13. Microprocessor Synchronization and Memory Control Video Board

	STATE \ BTPB, TPB	00	MAIN T0	DISPLAY S0	01	MAIN T0	DISPLAY S0	11	MAIN T0	DISPLAY S0	10	MAIN T0	DISPLAY S0	GO	DATAEN	ADDEN	T7**
UNSYNCHRONIZED	0	0/100	?	?	6/000	?	7	2/000	7	7	0/100	7	?	1	0	0	1
	1	13/101	4	0	6/000	?	7	2/000	7	7	0/100	7	?	1	0	0	1
	2	3/000	0	0	0/100	?	7	0/100	7	7	0/100	7	?	0	0	0	1
	3	4/000	1	0	0/100	?	7	0/100	7	7	0/100	7	?	0	0	0	1
	4	5/000	2	0	0/100	?	7	0/100	7	7	0/100	7	?	0	0	0	1
	5	1/100	3	0	0/100	?	7	0/100	7	7	0/100	7	?	0	0	0	1
	6	6/000	?	0	0/100	?	7	0/100	7	7	2/000	7	0	0	0	0	1
	7																
SYNCHRONIZED	8	9/110	0	4	6/000	?	7	2/000	7	7	0/100	7	?	1	1	1	1
	9	10/110	1	5	6/000	?	7	2/000	7	7	0/100	7	?	1	1	0	1
	10	11/110	2	6	6/000	?	7	2/000	7	7	0/100	7	?	1	1	0	1
	11	0/100	?	?	12/100	3	7	2/000	7	7	0/100	7	?	1	1	0	0
	12	13/101	4	0	6/000	?	7	2/000	7	7	0/100	7	?	1	0	0	1
	13	14/101	5	1	6/000	?	7	2/000	7	7	0/100	7	?	1	0	1	1
	14	15/101	6	2	6/000	?	7	2/000	7	7	0/100	7	?	1	0	1	1
	15	0/100	?	?	6/000	?	7	2/000	7	7	8/111	7	3	1	0	1	1

BTPB = TPB PROCESSOR/STORE MICROPROCESSOR
TPB = TPB DISPLAY MICROPROCESSOR
NEXT STATE/EARLY GO, EARLY DATAEN, EARLY ADDEN
T0 = STATE COUNT (T0-T7) OF PROCESSOR/STORE MICROPROCESSOR
S1 = STATE COUNT (S0-S7) OF DISPLAY MICROPROCESSOR

C1016

Figure 2-14. State Diagram for Synchronization State Machine

Cursor Counter

The Cursor counter is made up of two 4-bit binary synchronous counters connected in series to provide a maximum count capability of 256 characters. This counter is loaded when the Display microprocessor executes an Out 1 instruction (defined in section 3). This counter is loaded with the 1's complement of the character position of the cursor within the character line. If the cursor is not to be displayed on this line, a value greater than the number of characters in the line is loaded. The counter is allowed to count under the following conditions: (1) when the Display microprocessor goes into a Direct Memory Access (DMA) cycle, and (2) it is not time to load the number of scan line pairs.

When the number of characters loaded into the Data register causes a carry level from the Cursor counter, this carry level signifies that this character position is where the cursor is to be displayed. This carry level is gated as the eighth bit of the data character into the line buffer, together with the seven bits of data from the Data buffer. When the data is then shifted out of the Line buffer to be displayed, this bit acts as a flag to cause the cursor to be displayed in its proper character position in the line.

Utility Counter

The Utility counter is made up of two 4-bit binary synchronous counters connected in series to provide a maximum count capability of 256 characters. This counter is loaded when the Display microprocessor executes an Out 2 instruction (defined in section 3). This counter is loaded to a value equal to the number of characters to be displayed on a line. The value loaded is the 1's complement of the desired number. Normally, this value is 80, but it will be 40 if wide characters are to be displayed on this line. This counter is used to generate a DMA request to the Display microprocessor to request character information to load the line buffer. Whenever the Utility counter is not equal to maximum or a carry has not occurred and the Utility counter has a valid count in it, a DMA request will be made. The Utility counter is considered to have a valid count in it if an Out 2 instruction has been issued since power was applied or a reset has occurred.

Scan Line Counter and Decoder

The Scan Line counter is comprised of two 4-bit binary synchronous counters connected in series to give a maximum count capability of 256. This counter can be reset by the Q signal from the Display microprocessor. The Scan Line counter is loaded with the 2's complement of the number of scan line pairs in the character line. Normally, the number of scan line pairs will be eight. The binary count that would be loaded in that case would be 11111000. If the counter is loaded with a value from 8 to 16, those scan line pairs corresponding to the values of 8 to 16 would appear as a blank area on the display. This is to allow greater flexibility in creating line spacing. The lower values of the Scan Line counter (V0, V1, and V2) are decoded to indicate the underline area of the character space. The Last Scan Line level is also generated from this counter. This level is used in the Vertical Synchronizing circuits. The signal used to count the Scan Line Counter is HPX-16' synchronized with the Display microprocessor time S4. This signal is gated with the carry output of the Scan Line Counter to develop a signal called END'. In a normal sequence of operation, the Display microprocessor loads one of the line buffers, finishes its processing, and goes into an idle state with its internal register pointing to the next number (value) of scan line pairs to be used. This END' signal causes a DMA request and that value is loaded into the Scan line counter for this character line to be displayed. This signal also causes the line buffer last loaded to be selected for display.

Character Handling and Transformation

Refer to Video Board Block Diagrams, figures 2-8 and 2-9. A character in 7-bit ASCII code is transferred from the memory on the Processor and Storage board during the Display microprocessor DMA Out memory cycle and placed in the Data register. This DMA cycle was requested through the channel control logic as a result of the Utility counter not being at its maximum count.

This 7-bit character code is then gated into either the "UP" or "DOWN" Line buffer, along with the cursor information, as bit 8 if required. The character is shifted through the Line buffer. From there it is gated by a multiplexor into three character generator ROMs. Other inputs to the ROMs include the field designation and scan line count information. The ROMs convert the parallel input character into three sets of 8-bit parallel information.

This information is gated into three parallel input-serial output registers. Now the information is shifted out serially from the three registers, one bit at a time. Each bit of this information corresponds to one third of the character representation of one horizontal PEL for one scan line. These three bits are applied, along with character status information, to the input of a Programmable Read-Only Memory (PROM) in the Output Transformation logic.

The output is a binary value of 0 to 16. This value corresponds to one of the 16 values of grey (or light intensity) that will appear on the display for that horizontal PEL of the characters to be displayed.

This 4-bit binary value is applied to the input of two 32 x 8 PROMs. These PROMs convert the 4-bit binary value into 16 separate levels. These 16 levels plus the Horizontal Synchronizing level or Vertical Synchronizing level are applied to a digital-to-analog converter. The output is a voltage level from 0 volt to 1 volt, depending upon the level of grey (light intensity) to be displayed on the Cathode Ray Tube (CRT). The signal at this point is called the Composite Video Signal and is applied to the input of the Display board.

Display Modes

There are two Video modes in which characters are displayed: Normal Video, where gray characters are displayed on a black background, and Negative Video, where black characters are displayed on a gray background.

To achieve the Negative Video effect, the gray code values from the Output Transformation circuit (Video Output Section Block Diagram, figure 2-9) Programmable Read-Only Memory (PROM) are reversed. The output values for Normal Video are a binary count from 0 to 7. Black results in an output value of 0, and light gray results in an output value of 7. For Negative Video, the black level, or background, output value from the PROM is 7 and the light gray output, or character, results in a value of 0. The effect on the display results in a gray background and black characters.

The information to be displayed is divided into logical pages. These pages are divided into character lines. The character lines are 0 to 80 standard width characters or 0 to 40 wide characters. The page may be composed of 4 to 24 lines of data information plus one 80 standard character line of status information.

Data information is the character information received over the Data Communications lines or the character information entered from the Keyboard. The status character information is generated as a communication between the MTS-1 and the operator.

The total number of pages depends upon the amount of Random Access Storage (RAM) that is available. The number of lines and pages are controlled by the operator through a change in the configuration control information. This configuration information may be changed temporarily by modifying the configuration control information stored in a firmware table in RAM. A more permanent change may be achieved if the operator enters the configuration control information into the Electronically Alterable Read-Only Memory (EAROM). Refer to EAROM in Section 3. Refer also to the TD730/TD830 system Reference Manual, form number 1093788, for additional information.

The Video mode specifies how the characters are to be displayed on each page before the addition of any highlight modification is applied. A highlight is a change to the way a character is displayed.

A character may be highlighted or changed in the following ways:

a. Reverse: If the Video mode is normal (gray characters on a black background), the character is displayed as a black character on a gray background. If the Video mode is negative, (black characters on a gray background), the character is displayed as a gray character on a black background.

b. Underline: An underlined character has a horizontal line added to the bottom of its display.

c. Blink: A blinking character will be alternated with a space character at a 1.5 Hertz per second rate. All the characters chosen to blink on a page will blink at the same time. This blinking rate is controlled by Firmware.

d. Secure: The character display is replaced by a solid block of white, gray, or black. The color is determined by the specific Video mode in which the character(s) would normally appear on this page.

e. Bright: A bright character has its gray code values from the Output Transformation circuit PROM increased by a factor of 2. Thus, the binary output values (instead of being 0 through 7) would be 0 through 15 or 0, 2, 4, 6, 8, 10, 12, and 14 for normal output counts of 0, 1, 2, 3, 4, 5, 6, and 7. This will have the effect on the display of creating a brighter visual character.

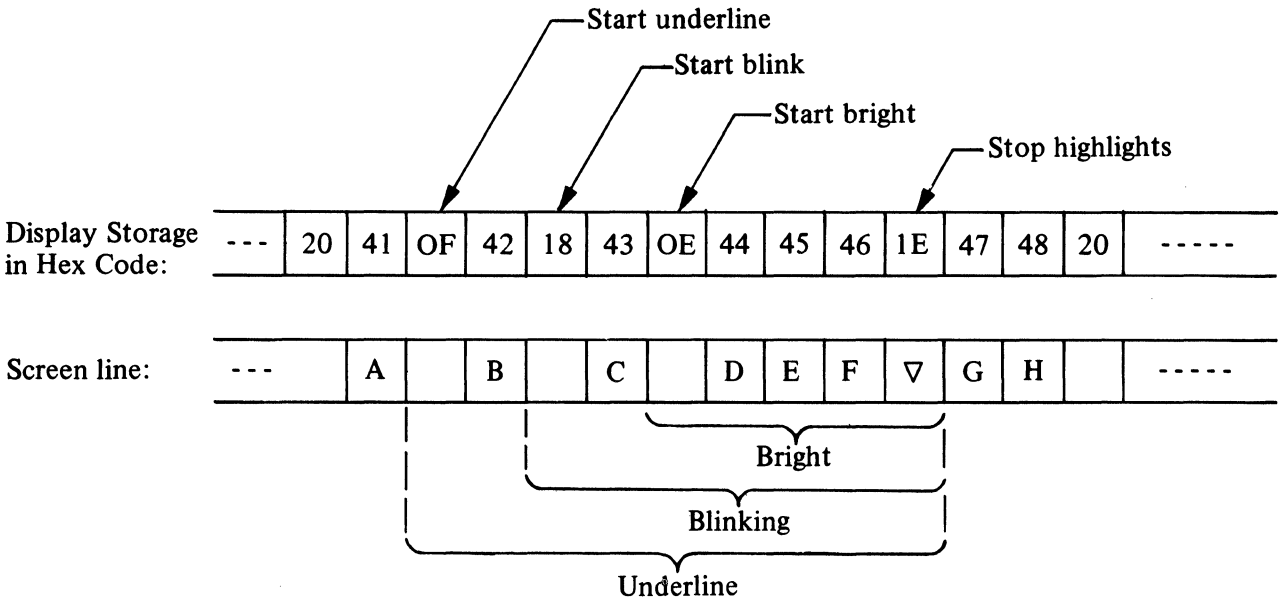
A character may have any number of the above highlights and in any combination. The designation of which characters will be highlighted is controlled by start highlight and stop highlight characters placed within the string of characters to be displayed on a character line. The highlight characters are usually not displayed, but appear as a space, or blank, in the characters line display. The highlight control characters are described below in terms of hexadecimal code representation, function, and displayed symbol (graphic).

Hex Code	Function	Graphic
OE	Start REVERSE highlight	blank
OF	Start UNDERLINE	blank
18	Start BLINK	blank
19	Start SECURE	blank
1A	Start BRIGHT	blank
1E	Start all highlights	◁ *

*May be blanked

The effect of highlight-start characters is to apply the specified highlight to the highlight character itself and to all characters to its right, up to and including either the first stop-highlight character, if any, or the last character on the same line.

For example:



Forms mode is used to send page information from the central system to assist the operator to input data in the proper format. In Forms mode, data are divided into unprotected and protected fields. The information in the unprotected field may be modified by the operator. The data in the protected field may not be accessed by the operator. These areas are outlined on the display by special symbols. Refer to TD730/TD830 reference manual for more information on Forms mode. When the MTS-1 is in Forms mode and negative display mode, two additional characters may cause a highlight effect. These characters are Hex code 1D and 1F with symbols of ▽ and ◁ . The effect will be to start a Reverse highlight. If the page is in Normal Video mode, or Normal Video mode and Forms mode, there will be no highlight effect.

Control Characters

Certain control characters can be sent by the microprocessor to the display logic. These characters must be transmitted before the line buffer for a line can be filled, and before the cursor position for that line is sent. These control characters differ from all other characters because their high-order bit is "1."

Status Control	Reset	Set
Field	F 0	F 1
Last Line	F 2	F 3
Character Blink Rate	F 4	F 5
Cursor Blink Rate'	F 6	F 7
Wide	F 8	F 9
Negative Video	F A	F B
Forms	F C	F D

Blinking Characters

If the highlight blink is on, the character will be displayed if character blink rate = 0, and be blanked if character blink rate = 1.

Blinking Cursor

The cursor is blanked if cursor blink rate' = 1. The blinking effect is under firmware control, which programmatically turns the blink status bit on and off.

Power-Up Sequence

When the Display module is powered-up, two things have to be done before normal operations can start.

a. The microprocessor has to be initialized and identified.

b. The display logic has to be started up.

Microprocessor Initialization

As part of the power-up sequence the microprocessor in the Display module is automatically reset, i.e., P=0, X=0, R (0) = 0.0 and Q=0. The interrupt line of the Display microprocessor has been wired to a logic low level. Therefore, it is immaterial for this processor if the interrupt is enabled or disabled.

During the start-up sequence the Display microprocessor has to be pointed to the display program, while the main microprocessor has to execute a different program. In order to be able to identify the Display microprocessor, its flag (EF1) has been wired to a logic high level: EF1'=0.

After the processor has been identified, the registers have to be assigned, and the Display microprocessor is off and running.

Display Logic Start-Up

After the microprocessor has been started up, the display logic has to be initialized. No vertical synchronization information is sent to the monitor until the logic has been completely initialized.

During the power-up sequence, the VPIX flip-flop is reset, thus blanking out the display. It is desirable to start up the display with the Last Line flip-flop set and the Field flip-flop in a known state. After setting the Last Line flip-flop and setting or resetting the Field flip-flop, R (0) must be set to point to the retrace parameters for the current field. At this point, the Q bit can be set, and the Display processor should go to the idle state. At the next "end of line" request the vertical retrace starts, and the display section is running.

Display Board

General

The purpose of the Display board is to take the information provided by the Composite Video signal and use this information to control the visual display on the CRT. Refer to figure 2-15 for a block diagram of the Display board and CRT.

Video Amplifier

The purpose of the Video Amplifier is to convert the small input signal information in the Composite Video signal to an amplitude large enough for the proper operation of the CRT.

The outputs of this circuit are to the Sync Stripper circuit, Control Grid circuit, and the cathode of the CRT. The inputs are the Composite Video signal and the control information from the Back Porch Clamping and Brightness circuit.

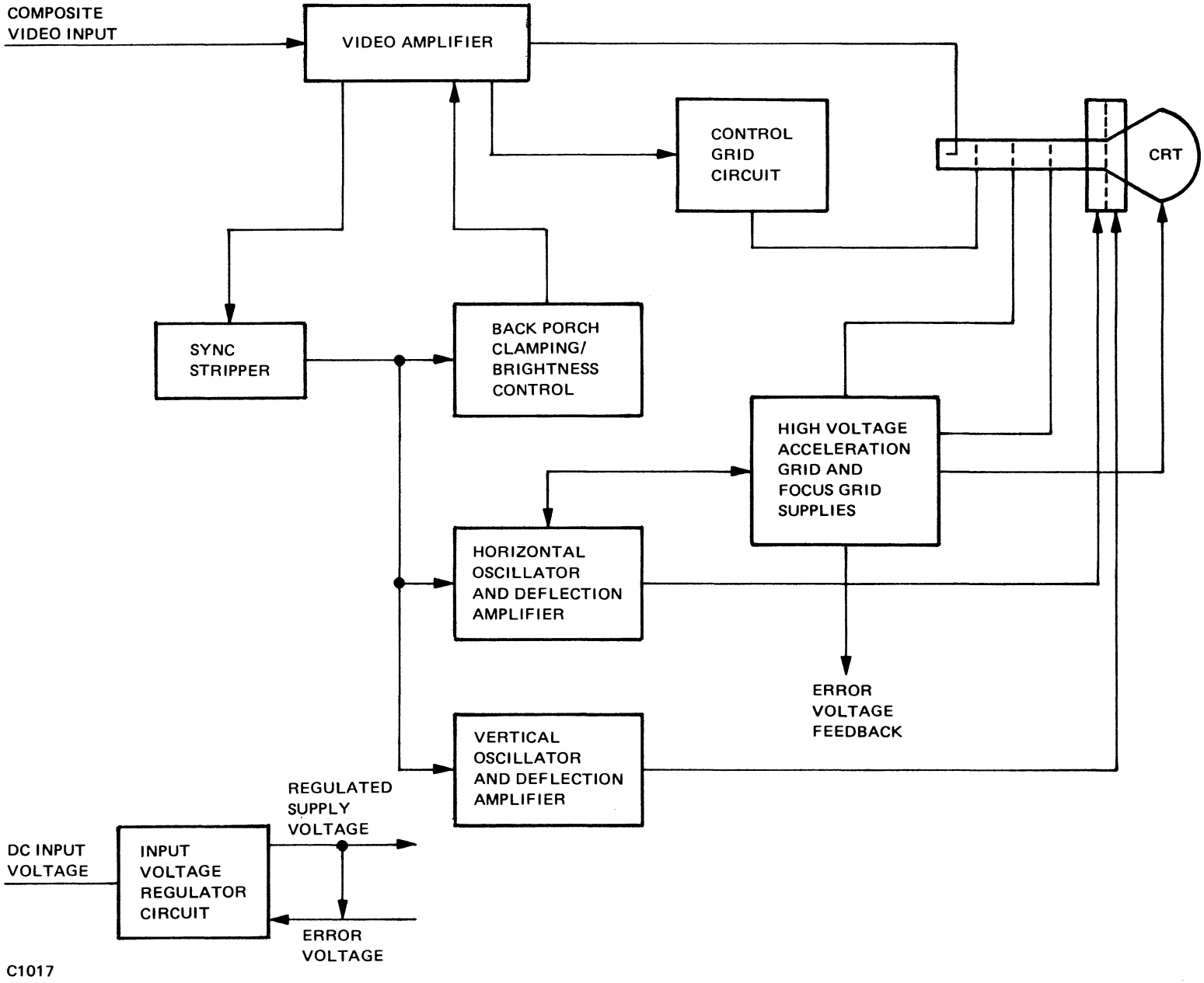


Figure 2-15. Display Monitor Control Board Block Diagram

The overall amplification factor of the Video Amplifier is controlled by the Contrast Control. A decrease in contrast means less difference in light intensity seen on the CRT for a given change in picture signal input amplitude. (The picture signal consists of the levels that occur between the Black and White signal reference levels of the Composite Video signal.) An increase in contrast means more difference in light intensity seen on the CRT for the same change in input signal amplitude.

Control Grid Circuit

This circuit develops the necessary Control Grid voltage for the proper operation of the CRT.

Sync Stripper

The Sync Stripper circuit removes the video from the Composite Video signal and applies the Horizontal and Vertical synchronizing pulses to the Back Porch Clamping and Brightness circuit, Horizontal Oscillator circuits, and Vertical Amplifier circuits.

Back Porch Clamping and Brightness Control

This circuit determines the brightness level which the entire display has when no picture signal is applied. When a signal is applied, the average voltage component of this input will add to this control to establish the brightness level of the picture background. Unlike the Contrast Control, the Brightness Control does not effect the amplitude of the Video signal. When the Contrast Control is advanced, the result on the display is a greater difference or range between the brightest and darkest elements of the picture. The bright areas become brighter and the dark areas darker. On the other hand, the Brightness Control changes all areas in the same direction; that is, when the Brightness Control is advanced, both the light and dark areas become lighter. When the control is turned the other way, all parts of the picture become darker.

This circuit also ensures that the electron beam is cut off during the Horizontal and Vertical Retrace Time by changing the brightness control voltage applied to the Video Amplifier. The time that this change is applied to the Video Amplifier is determined from the output signal of the Sync Stripper circuit.

Vertical Oscillator Circuits

The Vertical Oscillator circuits are made up of a special-purpose chip, external controlling components, and the Vertical Deflection Coils.

The special-purpose chip contains a Synchronizing Pulse Amplifier, Vertical Oscillator, Voltage Regulator for the Vertical Oscillator Pulse Amplifier and Shaper, and an Output Amplifier. This chip produces a sawtooth voltage/current waveform used by the Vertical Deflection Coils. Refer to figure 2-16.

The Vertical Deflection Coils transform the sawtooth waveform into a varying intensity magnetic field. The effect of this increasing field is to cause the electron beam to move slowly down the face of the CRT during the display time (the long side of the sawtooth waveform), and move rapidly back to the top of the CRT during the retrace time (the short side of the sawtooth waveform).

The input to the Vertical Oscillator circuits contains a signal integrating circuit. Its purpose is to detect when a Vertical Synchronizing pulse occurs. The narrow Horizontal Synchronizing pulse has no effect on the output of this circuit. When the wider Vertical Synchronizing pulse occurs, the circuit detects this and produces an output. This output is used to reset the Vertical Oscillator to begin the Vertical Retrace Time.

Horizontal Oscillator Circuits

The Horizontal Oscillator circuits include a special-purpose chip, external controlling components, and Horizontal Deflection Coils.

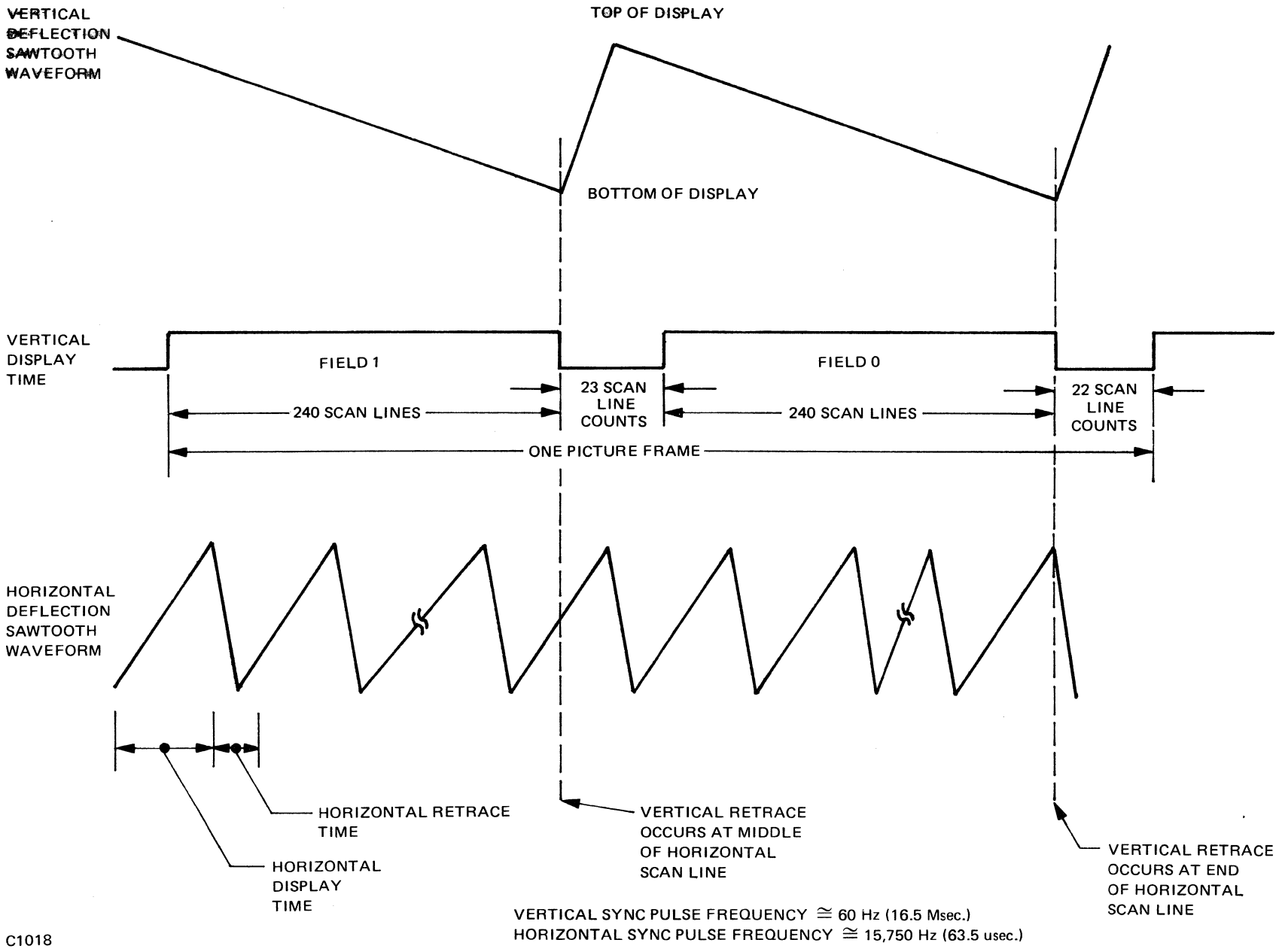


Figure 2-16. Horizontal and Vertical Deflection Waveforms

The special-purpose chip contains a Phase Detector, Voltage Regulator, Horizontal Oscillator, and an Output Amplifier circuit. This chip produces a sawtooth voltage/current waveform used by the Horizontal Deflection Coils and the High Voltage circuits. Refer to figure 2-16.

The Horizontal Deflection Coils transform the sawtooth waveform into a varying intensity magnetic field. The effect of this increasing field is to cause the electron beam to move slowly to the right side of the CRT (the long side of the sawtooth waveform), and move rapidly back to the left side of the CRT during the retrace time (the short side of the sawtooth waveform).

High Voltage, Acceleration and Focus Grid Supplies

The Horizontal Oscillator output is applied to a transformer. The transformer outputs are modified to satisfy several functions. One output is compared with the Horizontal Sync pulse by the Phase Detector whose output is used to control the Horizontal Oscillator frequency. A second output is rectified to provide the extremely high voltage needed by the CRT anode. This voltage is specified to be 13.0 $\pm$.8 kilovolts. A third output is rectified to provide the necessary Focus Control Grid voltages. A fixed voltage of 250 volts maximum and a variable voltage of 250 to 400 is used to provide the focusing action.

These two focus voltages are applied to separate grids within the CRT. The difference of voltage potential between these two grids causes an electrostatic field between them. The electron beam must pass through this field. If any of the electrons start to move away from this beam, they interact with the electrostatic field and are deflected back toward the beam. Increasing the voltage potential difference between the grids strengthens the electrostatic field and results in a greater convergence effect on the electron beam. This overall effect is referred to as Electrostatic Focus.

Input Voltage Regulator

The Input Voltage Regulator reduces the input voltage applied to the Display Board of 13.3 volts to 10.7 volts.

The error voltage is fed into the control circuit of the regulator and is proportional to any high voltage variation. If the high voltage varies, the output of the voltage regulator section will change to cancel the variation and cause the overall size of the displayed screen to remain constant.

SECTION 3
PROCESSOR AND STORAGE BOARD

FUNCTIONAL OPERATION

The Processor and Storage board contains a Processor and Storage Microprocessor Chip (PSMP), Electrically Alterable Read-Only Memory (EAROM), Random-Access Memory Storage (RAM), Bus Control logic, Timing and Control logic, and the Storage Address and Control logic. Refer to the block diagram provided in figure 3-1.

The microprocessor on the Processor and Storage board is the controlling microprocessor in the MTS-1. It accesses and executes the firmware program code stored in the ROM memory. Through the execution of this program code, the operation of the MTS-1 functions are accomplished. This includes operations such as communication with the keyboard and other Serial I/O capable devices, the controlling of the Data Communications microprocessor to communicate with a remote center by means of data communications procedures, and formatting character display lines in RAM memory.

Storage Allocation

The following is an overall view of the types of storage in terms of memory addressing.

Storage Address	Type
0000 - 7FFF	Masked Read-Only Storage
8000 - BFFF	External Storage (ROM or RAM)
8000 - FFFF	Read/Write Storage

The masked read-only storage is expandable starting at address 0000 in steps of 2 kilobytes for current maximum of 32 kilobytes. The RAM is expandable from address FFFF down in steps of 1 kilobyte to an address of C000. This provides a capability of 16 kilobytes. Currently a one-page MTS-1 would have 7 kilobytes and a two-page MTS-1 would have 10 kilobytes.

The capability of using external ROM or RAM storage exists. The addresses currently available for this are 8000 - BFFF for 16 kilobytes maximum.

Any read access mode in the address space 0000 - FFFF which does not contain a storage chip will result in all ones on the data bus.

Electrically Alterable-Read Only Memory

Another type of storage used is the Electrically Alterable Read-Only Memory (EAROM). (See figure 3-2.)

The EAROM chip is a serial input/output 1400-bit, electrically erasable, and reprogrammable ROM organized as 100 words of 14 bits each. Data and addresses are communicated in serial form by means of a one-pin bidirectional bus.

Addressing is by two consecutive one-of-ten codes. Mode selection is by a 3-bit code applied to C1, C2, and C3.

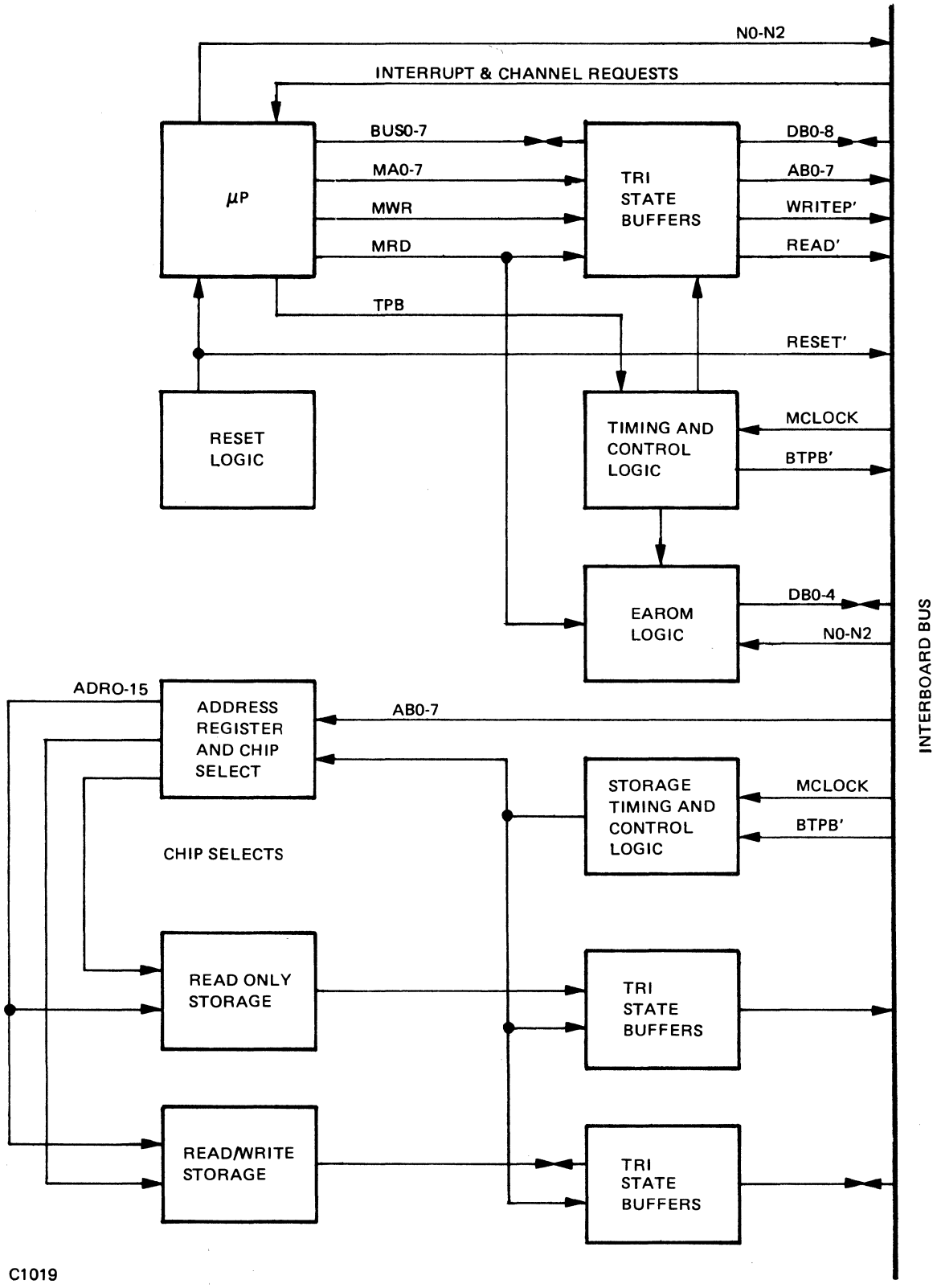


Figure 3-1. Processor and Storage Board Simplified Block Diagram

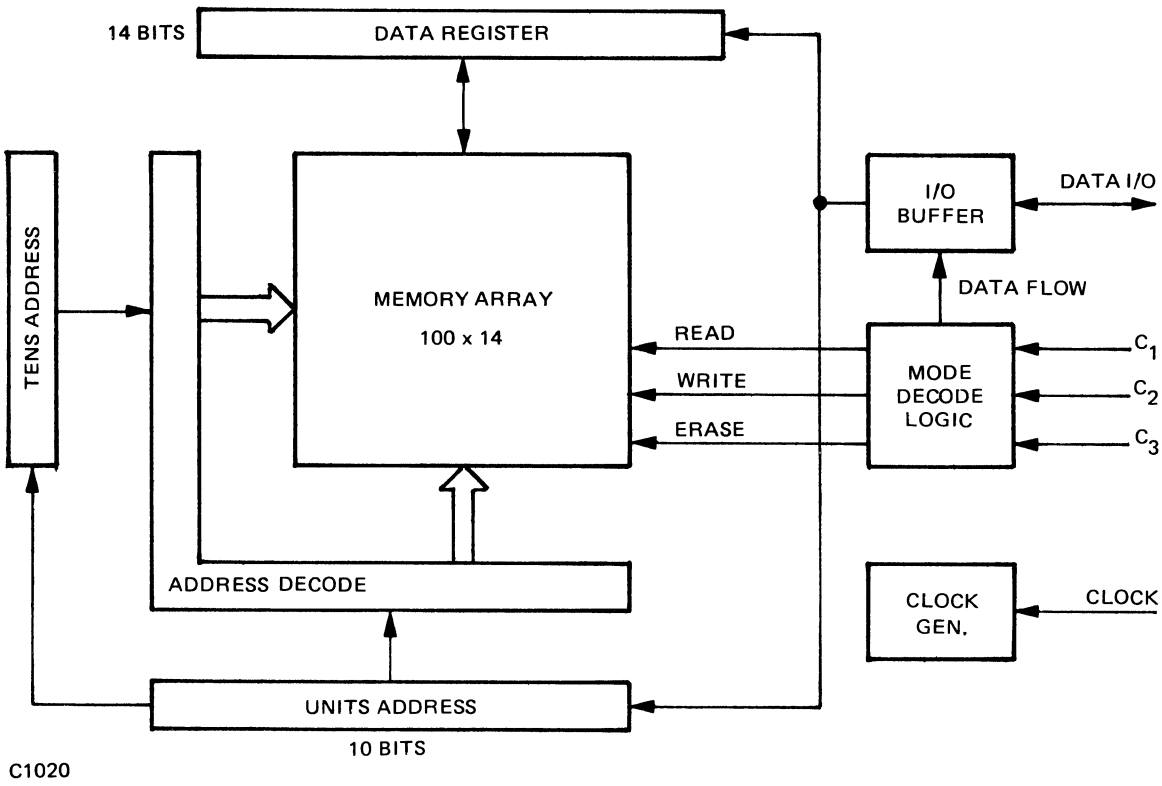


Figure 3-2. EAROM Block Diagram

Figure 3-3 outlines the function of each of the eight pins of the EAROM. Figure 3-4 illustrates some of the mode control functions in timing diagram form.

The EAROM is used for storing configuration control information. The MTS-1 contains 32 bytes of permanently stored data which define the terminal configuration.

During power-up or after a power reset pulse caused by an input line fluctuation, the stored data is transferred under Program control to RAM memory. The data stored in the RAM may be altered through ESC or CTRL functions. Any power-down will erase the temporarily stored data.

Microprocessor Description

Figure 3-5 shows a block diagram of a typical microprocessor system which is similar to the way the microprocessors are used in the MTS-1.

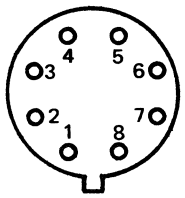
Figure 3-6 shows the signal-to-pin assignment for the RCA CDP1802 microprocessor.

Figure 3-7 shows the general timing describing the RCA CDP1802 microprocessor operation.

PIN NO.	NAME	FUNCTION
1	DATA	IN THE ACCEPT ADDRESS AND ACCEPT DATA MODES, THIS PIN IS AN INPUT PIN FOR ADDRESS AND DATA RESPECTIVELY. IN THE SHIFT DATA OUT MODE THIS PIN IS AN OUTPUT PIN DESIGNED TO DRIVE MOS. IN STANDBY, READ, ERASE AND WRITE, THIS PIN IS LEFT FLOATING.
2	V_M	USED FOR TESTING PURPOSES ONLY. SHOULD BE LEFT UNCONNECTED FOR NORMAL OPERATION.
3	V_{SS}	CHIP SUBSTRATE. NORMALLY CONNECTED TO GROUND.
4	V_{GG}	DC SUPPLY. NORMALLY CONNECTED TO -35 VOLT SUPPLY.
5	CLOCK	14KHZ TIMING REFERENCE.
6, 7, 8	C1, C2, C3	MODE CONTROL PINS. THEIR OPERATION IS AS FOLLOWS:

C1	C2	C3	FUNCTION
0	0	0	STANDBY - CONTENTS OF ADDRESS AND DATA REGISTER REMAINS UNCHANGED. OUTPUT BUFFER IS LEFT FLOATING.
0	1	1	ACCEPT ADDRESS - DATA PRESENTED AT THE I/O PIN IS SHIFTED INTO THE ADDRESS REGISTER WITH EACH CLOCK PULSE.
1	0	0	READ - THE ADDRESS WORD IS READ FROM MEMORY INTO THE DATA REGISTER.
1	0	1	SHIFT DATA OUT - THE OUTPUT DRIVER IS ENABLED AND THE CONTENTS OF THE DATA REGISTER ARE SHIFTED OUT ONE BIT WITH EACH CLOCK PULSE.
0	1	0	ERASE - THE WORD STORED AT THE ADDRESSED LOCATION IS ERASED TO ALL ZEROS.
1	1	1	ACCEPT DATA - THE DATA REGISTER ACCEPTS SERIAL DATA PRESENTED AT THE I/O PIN. THE ADDRESS REGISTER REMAINS UNCHANGED.
1	1	0	WRITE - THE WORD CONTAINED IN THE DATA REGISTER IS WRITTEN INTO THE LOCATION DESIGNATED BY THE ADDRESS REGISTER.
0	0	1	NOT USED.

BOTTOM VIEW



1. DATA I/O
2. V_M
3. V_{SS}
4. V_{GG}
5. CLOCK
6. C1
7. C2
8. C3

C1021

Figure 3-3. EAROM Functions

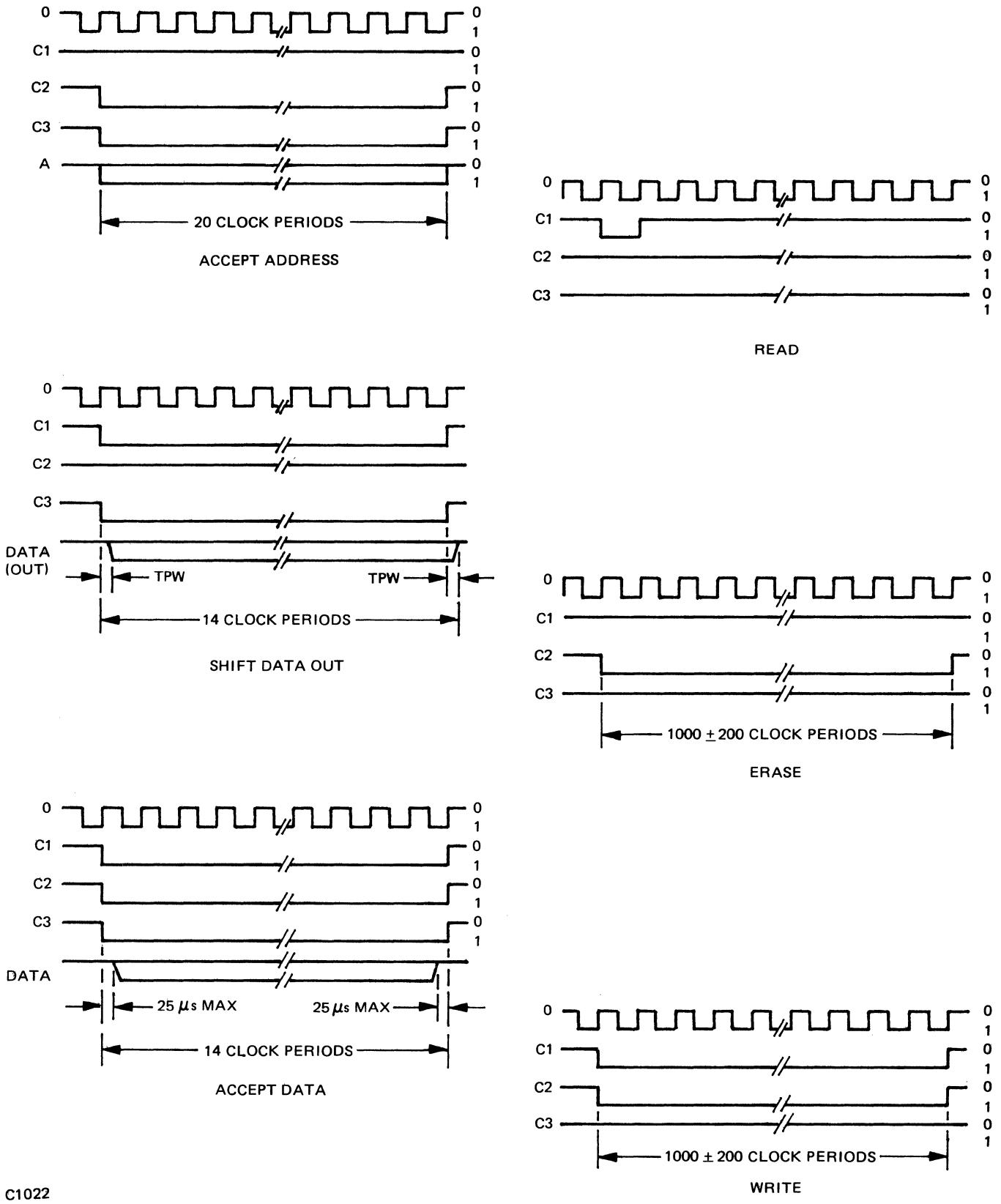
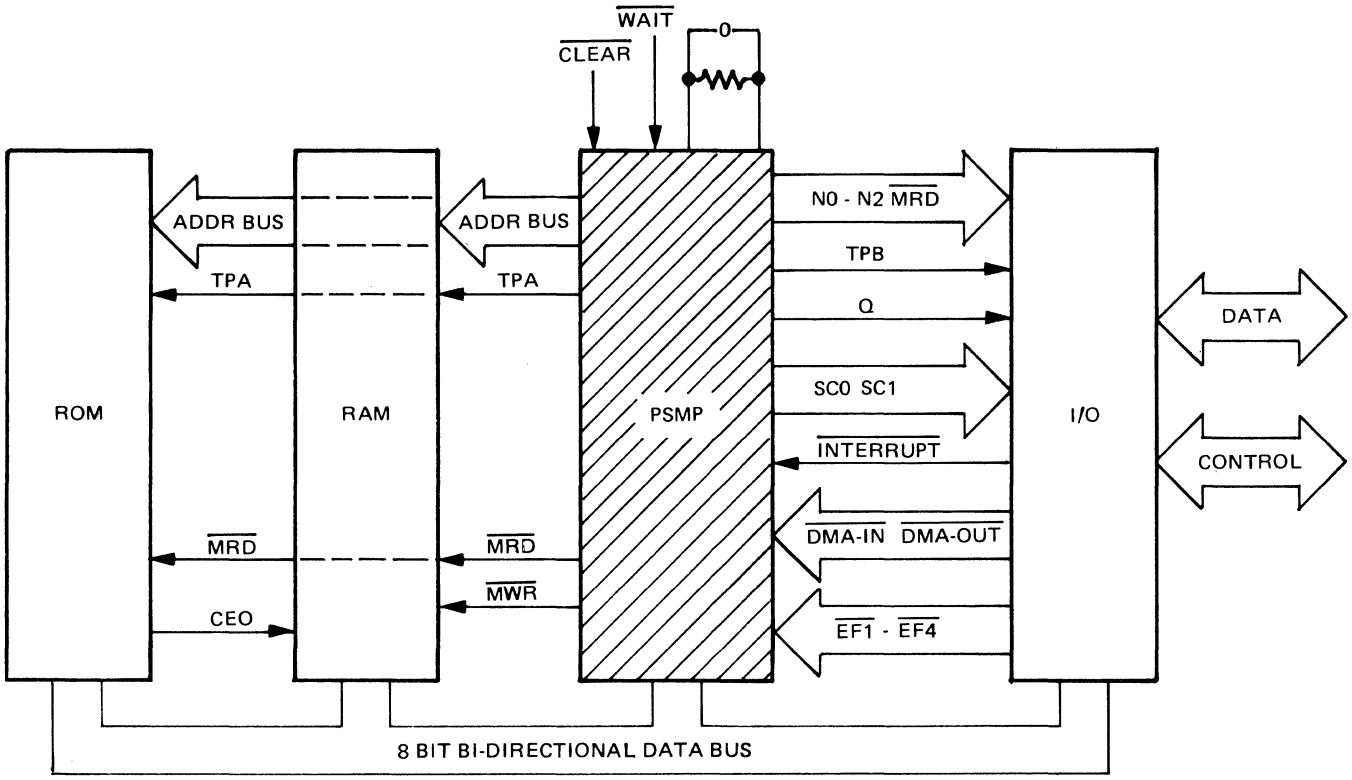
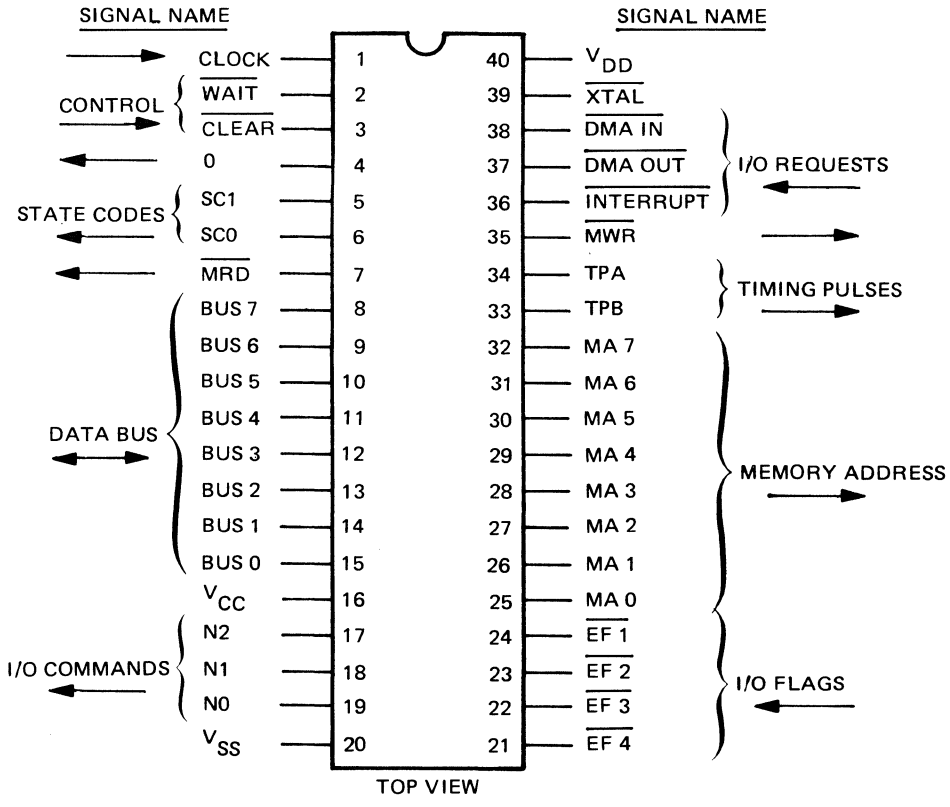


Figure 3-4. EAROM Mode Timing Diagrams



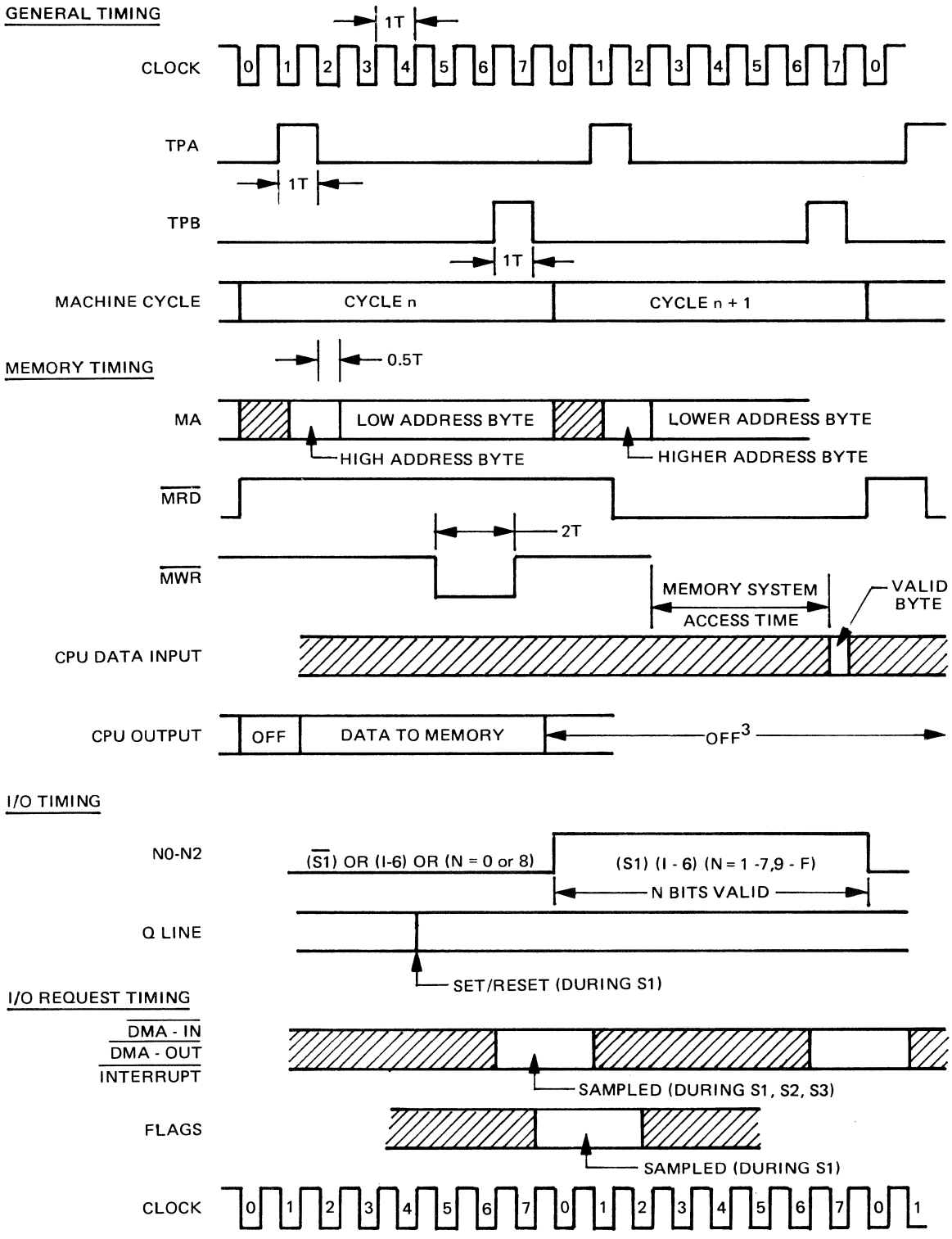
C1023

Figure 3-5. Typical Microprocessor System



C1024

Figure 3-6. Microprocessor Pin Assignment



NOTES:

1. USER GENERATED SIGNALS
2. SHADING INDICATES "DONT CARE" OR INTERNAL DELAY
3. "OFF" INDICATES HIGH-IMPEDANCE STATE

C1025

Figure 3-7. Timing Diagram

Internal Architecture of the Microprocessor

The microprocessor block diagram is shown in figure 3-8. The principal feature of this system is a register array (R) consisting of sixteen 16-bit scratchpad registers. Individual registers in the array (R) are designated (selected) by a 4-bit binary code from one of the 4-bit registers labeled N, P, and X. The contents of any register can be directed to any one of the following three paths:

a. The external memory (multiplexed, higher-order byte first, onto 8 memory address lines);

b. The D register (either of the two bytes can be gated to D);

c. The increment/decrement circuit where it is increased or decreased by one and stored back in the selected 16-bit register.

The three paths, depending on the nature of the instruction, may operate independently or in various combinations in the same machine cycle.

With two exceptions, microprocessor instructions consist of two 8-clock pulse machine cycles. The first cycle is the fetch cycle, and the second (and third, if necessary) are execute cycles. During the fetch cycle, the four bits in the P designator select one of the 16 registers R(P) as the current program counter. The selected register R(P) contains the address of the memory location from which the instruction is to be fetched. When the instruction is read out from the memory, the higher-order 4 bits of the instruction byte are loaded into the I register and the lower-order 4 bits into the N register. The content of the program counter is automatically incremented by one so that R(P) is now "pointing" to the next byte in the memory.

The X designator selects one of the 16 registers R(X) to "point" to the memory for an operand (or data) in certain Arithmetic Logic Unit (ALU) or I/O operations.

The N designator can perform the following five functions depending on the type of instruction fetched:

a. designate one of the 16 registers in R to be acted upon during register operations;

b. indicate to the I/O devices a command code or device-selection code for peripherals;

c. indicate the specific operation to be executed during the ALU instructions, types of tests to be performed during the Branch instructions, or the specific operation required in a class of miscellaneous instructions (70-73 and 78-7B);

d. Indicate the value to be loaded into P to designate a new register to be used as the program counter R(P);

e. indicate the value to be loaded into X to designate a new register to be used as data pointer R(X).

The registers in R can be assigned by a programmer in three different ways: as program counters, as data pointers, or as scratchpad locations (data registers) to hold two bytes of data.

Program Counters

Any register can be the main program counter; the address of the selected register is held in the P designator. Other registers in R can be used as subroutine program counters. By a single instruction, the contents of the P register can be changed to effect a "call" to a subroutine. When interrupts are being serviced, register R(1) is used as the program counter for the interrupt servicing routine. At all other times, the register designated as program counter is at the discretion of the user.

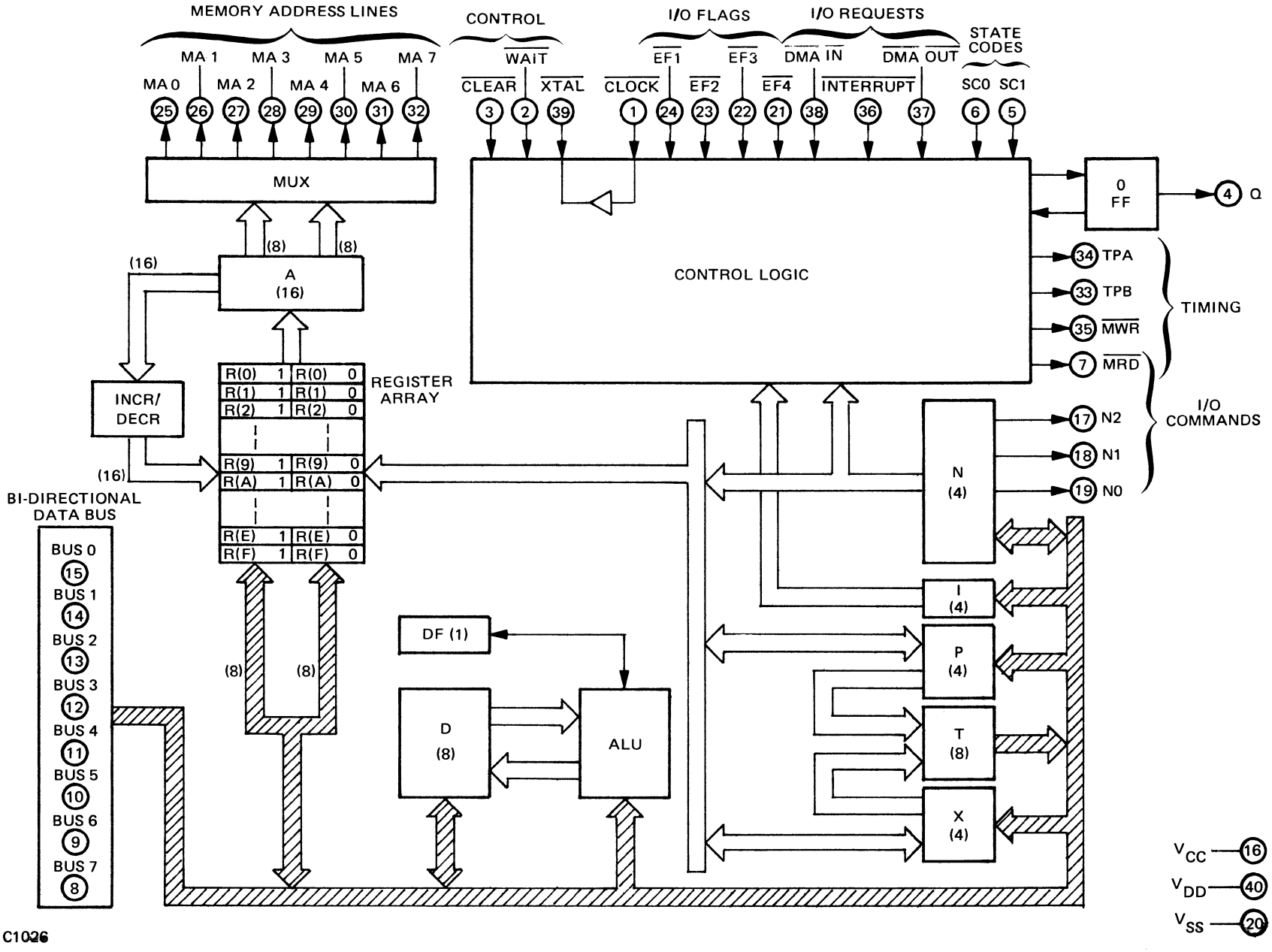


Figure 3-8. Microprocessor Block Diagram

Data Pointers

The registers in R may be used as data pointers to indicate a location in memory. The register designated by X (i.e., R(X)) points to memory for the following instructions (see figure 3-10):

a. ALU operations F0-F5, F7, 74, 75, 77;

b. output instructions 61 through 67;

c. input instructions 69 through 6F;

d. certain miscellaneous instructions - 70-73, 78.

The register designated by N (i.e., R(N)) points to memory for the "load D from memory" instructions ON and 4N and the "Store D" instruction 5N. The register designated by P (i.e., the program counter) is used as the data pointer for ALU instructions F8-FD, FF, 7C, 7D, 7F. During these instruction executions the operation is referred to as "data immediate."

Another important use of R as a data pointer supports the built-in Direct-Memory-Access (DMA) function. When a DMA-In or DMA-Out request is received, one machine cycle is "stolen." This operation occurs at the end of the execute machine cycle in the current instruction. Register R(0) is always used as the data pointer during the DMA operation. The data is read from (DMA-Out) or written into (DMA-In) the memory location pointed to by the R(0) register. At the end of the transfer, R(0) is incremented by one so that the processor is ready to act upon the next DMA byte transfer request. This feature in the architecture saves a substantial amount of logic when fast exchanges of blocks of data are required, such as with magnetic discs or during CRT-display-refresh cycles.

A program load facility, using the DMA-In channel, is provided to enable users to load programs into the memory. This facility provides a simple, one-step means for initially entering programs into the microprocessor system and eliminates the requirement for specialized "bootstrap" ROM's.

Data Registers

When registers in R are used to store bytes of data, four instructions are provided which allow D to receive from or write into either the higher-order-or lower-order-byte portions of the register designated by N. By this mechanism (together with loading by data immediate) program pointer and data pointer designations are initialized. Also, this technique allows scratchpad registers in R to be used to hold general data. By employing increment or decrement instructions, such registers may be used as loop counters.

The Q Flip-Flop

An internal flip-flop, Q, can be set or reset by instruction and can be sensed by conditional branch instructions. The output of Q is also available as a microprocessor output.

Interrupt Servicing

Register R(1) is always used as the program counter whenever interrupt servicing is initiated. When an interrupt request comes in and the interrupt is allowed by the program (again, nothing takes place until the completion of the current instruction) the contents of the X and P registers are stored in the temporary register T, and X and P are set to new values; hex digit 2 in X and hex digit 1 in P. Interrupt enable is automatically deactivated to inhibit further interruptions. The interrupt routine is now in control; the contents of T are saved by means of a single instruction (78) in the memory location pointed to by R(X). At the conclusion of the interrupt, the routine restores the pre-interrupted values of X and P with a single instruction (70 or 71). The interrupt-enable flip-flop can be activated to permit further interrupts or can be disabled to prevent them.

See figure 3-9 for a summary of the internal registers of the microprocessor.

D	8 BITS	DATA REGISTER (ACCUMULATOR)
DF	1 BIT	DATA FLAG (ALU CARRY)
R	16 BITS	1 OF 16 SCRATCHPAD REGISTERS
P	4 BITS	DESIGNATES WHICH REGISTER IS PROGRAM COUNTER
X	4 BITS	DESIGNATES WHICH REGISTER IS DATA POINTER

N	4 BITS	HOLDS LOW-ORDER INSTR. DIGIT
I	4 BITS	HOLDS HIGH-ORDER INSTR. DIGIT
T	8 BITS	HOLDS OLD X, P AFTER INTER RUPT (X IS HIGH BYTE)
IE	1 BIT	INTERRUPT ENABLE
Q	1 BIT	OUTPUT FLIP-FLOP

C1027

Figure 3-9. Internal Register Summary

State Transitions

The microprocessor state transitions when in the RUN mode are shown in figure 3-10. Each machine cycle requires the same period of time-8 clock pulses. The execution of an instruction requires either two or three machine cycles, S0 followed by a single S1 cycle or two S1 cycles. S2 is the response to a DMA request and S3 is the interrupt response.

Instruction Set

The microprocessor instruction summary is given in figure 3-11. Hexadecimal notation is used to refer to the 4-bit binary codes.

In all registers, bits are numbered from the least significant bit (LSB) to the most significant bit (MSB) starting with 0.

R(W): Register designated by W, where W=N or X, or P

R(W)0: Lower-order byte of R(W)

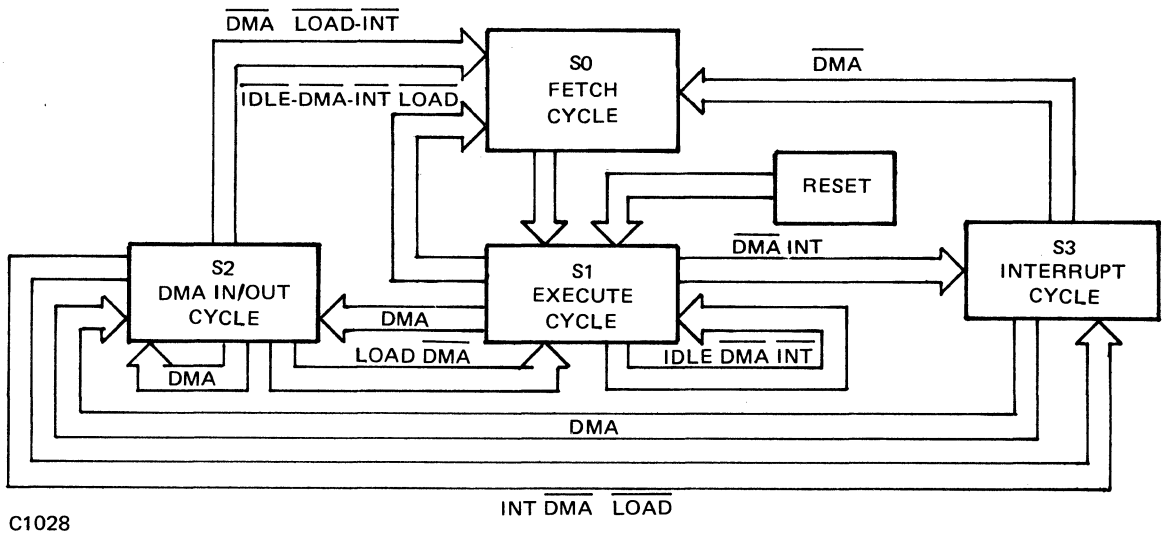
R(W)1: Higher-order byte of R(W)

N0 = Least significant Bit of N Register

Operation Notation

M(R(N)) → D; R(N) + 1

This notation means: The memory byte pointed to by R(N) is loaded into D, and R(N) is incremented by 1.



C1028

Figure 3-10. Microprocessor State Transistions (Run Mode)

INSTRUCTION	MNEMONIC	OP CODE	OPERATION
MEMORY REFERENCE			
LOAD VIA N	LDN	0N	M(R(N)) → D; FOR N NOT 0
LOAD ADVANCE	LDA	4N	M(R(N)) → D; R(N) +1
LOAD VIA X	LDX	F0	M(R(X)) → D
LOAD VIA X AND ADVANCE	LDXA	72	M(R(X)) → D; R(X) +1
LOAD IMMEDIATE	LDI	F8	M(R(P)) → D; R(P) +1
STORE VIA N	STR	5N	D → M(R(N))
STORE VIA X AND DECREMENT	STXD	73	D → (R(X)); R(X) −1
REGISTER OPERATIONS			
INCREMENT REG N	INC	1N	R(N) +1
DECREMENT REG N	DEC	2N	R(N) −1
INCREMENT REG X	IRX	60	R(X) +1
GET LOW REG N	GLO	8N	R(N).0 → D
PUT LOW REG N	PLO	AN	D → R(N).0
GET HIGH REG N	GHI	9N	R(N).1 →D
PUT HIGH REG N	PHI	BN	D →R(N).1
LOGIC OPERATIONS**			
OR	OR	F1	M(R(X)) OR D →D
OR IMMEDIATE	ORI	F9	M(R(P)) OR D →D; R(P) +1
EXCLUSIVE OR	XOR	F3	M(R(X)) XOR D →D
EXCLUSIVE OR IMMEDIATE	XRI	FB	M(R(P)) XOR D →D; R(P) +1
AND	AND	F2	M(R(X)) AND D →D
AND IMMEDIATE	ANI	FA	M(R(P)) AND D →D; R(P) +1
SHIFT RIGHT	SHR	F6	SHIFT D RIGHT, LSB(D) →DF, 0 →MSB(D)
SHIFT RIGHT WITH CARRY	SHRC	76*	SHIFT D RIGHT, LSB(D) →DF, DF →MSB(D)
RING SHIFT RIGHT	RSHR		
SHIFT LEFT	SHL	FE	SHIFT D LEFT, MSB(D) →DF, 0 →LSB(D)
SHIFT LEFT WITH CARRY	SHLC	7E*	SHIFT D LEFT, MSB(D) →DF, DF →LSB(D)
RING SHIFT LEFT	RSHL		

*NOTE: THIS INSTRUCTION IS ASSOCIATED WITH MORE THAN ONE MNEMONIC. EACH MNEMONIC IS INDIVIDUALLY LISTED.

**NOTE: THE ARITHMETIC OPERATIONS AND THE SHIFT INSTRUCTIONS ARE THE ONLY INSTRUCTIONS THAT CAN ALTER THE DF.
AFTER AN ADD INSTRUCTION:
DF = 1 DENOTES A CARRY HAS OCCURRED
DF = 0 DENOTES A CARRY HAS NOT OCCURRED
AFTER A SUBTRACT INSTRUCTION:
DF = 1 DENOTES NO BORROW. D IS A TRUE POSITIVE NUMBER
DF = 0 DENOTES A BORROW. D IS TWO'S COMPLEMENT
THE SYNTAX "−(NOTE DF)" DENOTES THE SUBTRACTION OF THE BORROW

C1029/SHEET 1 OF 4

Figure 3-11. Instruction Summary (Sheet 1 of 4)

INSTRUCTION	MNEMONIC	OP CODE	OPERATION
ARITHMETIC OPERATIONS**			
ADD	ADD	F4	M(R(X)) +D →DF, D
ADD IMMEDIATE	ADI	FC	M(R(P)) +D →DF, D; R(P) +1
ADD WITH CARRY	ADC	74	M(R(X)) +D +DF →DF, D
ADD WITH CARRY, IMMEDIATE	ADCI	7C	M(R(P)) +D +DF →DF, D R(P) +1
SUBTRACT D	SD	F5	M(R(X))–D →DF, D
SUBTRACT D IMMEDIATE	SDI	FD	M(R(PP)–D →DF, D; R(P) +1
SUBTRACT D WITH BORROW	SDB	75	M(R(X))–D–(NOT DF) →DF, D
SUBTRACT D WITH BORROW, IMMEDIATE	SDBI	7D	M(R(P))–D–(NOT DF) →DF, D; R(P) +1
SUBTRACT MEMORY	SM	F7	D–M(R(X)) →DF, D
SUBTRACT MEMORY IMMEDIATE	SMI	FF	D–M(R(P)) →DF, D; R(P) +1
SUBTRACT MEMORY WITH BORROW	SMB	77	D–M(R(X))–NIT DF) →DF, D
SUBTRACT MEMORY WITH BORROW, IMMEDIATE	SMBI	7F	D–M(R(PP))–(NOT DF) →DF, D R(P) +1
BRANCH INSTRUCTIONS–SHORT BRANCH			
SHORT BRANCH	BR	30	M(R(P)) →R(P).0
NO SHORT BRANCH (SEE SKP)	NBR	38*	R(P) +1
SHORT BRANCH IF D=0	BZ	32	IF D=0, M(R(P)) →R(P).0 ELSE R(P) +1
SHORT BRANCH IF D NOT 0	BNZ	3A	IF D NOT 0, M(R(P)) →R(P).0 ELSE R(P) +1
SHORT BRANCH IF DF=1	BDF	33*	EF DF=1, M(R(PP)) →R(P).0
SHORT BRANCH IF POS OR ZERO	BPZ		ELSE R(P) +1
SHORT BRANCH IF EQUAL OR GREATER	BGE		
SHORT BRANCH IF DF=0	BNF	3B*	IF DF=0, M(R(P)) →R(P).0
SHORT BRANCH IF MINUS	BM		ELSE R(P) +1
SHORT BRANCH IF LESS	BL		
SHORT BRANCH IF Q=1	BQ	31	IF Q=1, M(R(P)) →R(P).0 ELSE R(P) +1
SHORT BRANCH IF Q=0	BNQ	39	IF Q=0, M(R(P)) →R(P).0 ELSE R(P) +1
SHORT BRANCH IF EF1=1	B1	34	IF EF1=1, M(R(P)) →R(P).0 ELSE R(P) +1
SHORT BRANCH IF EF1=0	BN1	3C	IF EF1=0, M(R(P)) →R(P).0 ELSE R(P) +1
SHORT BRANCH IF EF2=1	B2	35	IF EF2=1, M(R(P)) →R(P).0 ELSE R(P) +1
SHORT BRANCH IF EF2=0	BN2	3D	IF EF2=0, M(R(P)) →R(P).0 ELSE R(P) +1
SHORT BRANCH IF EF3=1	B3	36	IF EF3=1, M(R(P)) →R(P).0 ELSE R(P) +1
SHORT BRANCH IF EF3=0	BN3	3E	IF EF3=0, M(R(P)) →R(P).0 ELSE R(P) +1
SHORT BRANCH IF EF4=1	B4	37	IF EF4=1, M(R(P)) →R(P).0 ELSE R(P) +1
SHORT BRANCH IF EF4=0	BN4	3F	IF EF4=0, M(R(P)) →R(P).0 ELSE R(P) +1

C1029/SHEET 2 OF 4

Figure 3-11. Instruction Summary (Sheet 2 of 4)

INSTRUCTION	MNEMONIC	OP CODE	OPERATION
BRANCH INSTRUCTIONS–LONG BRANCH			
LONG BRANCH	LBR	C0	M(R(P)) →R(P).1 M(R(P) +1) →R(P).0
NO LONG BRANCH (SEE LSKP)	NLBR	C8*	R(P) +2
LONG BRANCH IF D=0	LBZ	C2	IF D=0, M(R(P)) →R(P).1 M(R(P) +1) →R(P).0 ELSE R(P) +2
LONG BRANCH IF D NOT 0	LBNZ	CA	IF D NOT 0, M(R(P)) →R(P).1 M(R(P) +1) →R(P).0 ELSE R(P) +2
LONG BRANCH IF DF=1	LBDF	C3	IF DF=1, M(R(P)) →R(P).1 M(R(P) +1) →R(P).0 ELSE R(P) +2
LONG BRANCH IF DF=0	LBNF	CB	IF DF=0, M(R(P)) →R(P).1 M(R(P) +1) →R(P).0 ELSE R(P) +2
LONG BRANCH IF Q=1	LBQ	C1	IF Q=1, M(R(P)) →R(P).1 M(R(P) +1) →R(P).0 ELSE R(P) +2
LONG BRANCH IF Q=0	LBNQ	C9	IF Q=0, M(R(P)) →R(P).1 M(R(P) +1) →R(P).0 ELSE R(P) +2
SKIP INSTRUCTIONS			
SHORT SKIP (SEE NBR)	SKP	38*	R(P) +1
LONG SKIP (SEE NLBR)	LSKP	C8*	R(P) +2
LONG SKIP IF D=0	LSZ	CE	IF D=0, R(P) +2 ELSE CONTINUE
LONG SKIP IF D NOT 0	LSNZ	C6	IF D NOT 0, R(P) +2 ELSE CONTINUE
LONG SKIP IF DF=1	LSDF	CF	IF DF=1, R(P) +2 ELSE CONTINUE
LONG SKIP IF DF=0	LSNF	C7	IF DF=0, R(P) +2 ELSE CONTINUE
LONG SKIP IF Q=1	LSQ	CD	IF Q=1, R(P) +2 ELSE CONTINUE
LONG SKIP IF Q=0	LSNQ	C5	IF Q=0, R(P) +2 ELSE CONTINUE
LONG SKIP IF IE=1	LSIE	CC	IF IE=1, R(P) +2 ELSE CONTINUE

C1029/SHEET 3 OF 4

Figure 3-11. Instruction Summary (Sheet 3 of 4)

INSTRUCTION	MNEMONIC	OP CODE	OPERATION
CONTROL INSTRUCTIONS			
IDLE	IDL	00#	WAIT FOR DMA OR INTERRUPT; M(R(0)) →BUS
NO OPERATION	NOP	C4	CONTINUE
SET P	SEP	DN	N →P
SET X	SEX	EN	N →X
SET Q	SEQ	7B	1 →Q
RESET Q	REQ	7A	0 →Q
SAVE	SAV	78	T →M(R(X))
PUSH X, P TO STACK	MARK	79	(X, P) →T; (X, P) →M(R(2)) THEN P →X; R(2) – 1
RETURN	RET	70	M(R(X)) →(X, P); R(X) +1 1 →IE
DISABLE	DIS	71	M(R(X)) →(X, P); R(X) +1 0 →IE
INPUT–OUTPUT BYTE TRANSFER			
OUTPUT 1	OUT 1	61	M(R(X)) →BUS; R(X) +1; N LINES = 1
OUTPUT 2	OUT 2	62	M(R(X)) →BUS; R(X) +1; N LINES = 2
OUTPUT 3	OUT 3	63	M(R(X)) →BUS; R(X) +1; N LINES = 3
OUTPUT 4	OUT 4	64	M(R(X)) →BUS; R(X) +1; N LINES = 4
OUTPUT 5	OUT 5	65	M(R(X)) →BUS; R(X) +1; N LINES = 5
OUTPUT 6	OUT 6	66	M(R(X)) →BUS; R(X) +1; N LINES = 6
OUTPUT 7	OUT 7	67	M(R(X)) →BUS; R(X) +1; N LINES = 7
INPUT 1	INP 1	69	BUS →M(R(X)); BUS →D; N LINES = 1
INPUT 2	INP 2	6A	BUS →M(R(X)); BUS →D; N LINES = 2
INPUT 3	INP 3	6B	BUS →M(R(X)); BUS →D; N LINES = 3
INPUT 4	INP 4	6C	BUS →M(R(X)); BUS →D; N LINES = 4
INPUT 5	INP 5	6D	BUS →M(R(X)); BUS →D; N LINES = 5
INPUT 6	INP 6	6E	BUS →M(R(X)); BUS →D; N LINES = 6
INPUT 7	INP 7	6F	BUS →M(R(X)); BUS →D; N LINES = 7

#AN IDLE INSTRUCTION INITIATES A REPEATING S1 CYCLE. THE PROCESSOR WILL CONTINUE TO IDLE UNTIL AN I/O REQUEST (INTERRUPT, DMA-IN, OR DMA-OUT) IS ACTIVATED. WHEN THE REQUEST IS ACKNOWLEDGED, THE IDLE CYCLE IS TERMINATED AND THE I/O REQUEST IS SERVICED, AND THEN NORMAL OPERATION IS RESUMED.

C1029/SHEET 4 OF 4

Figure 3-11. Instruction Summary (Sheet 4 of 4)

Additional Instruction Information

Refer to figure 3-12 for additional instruction information.

Microprocessor Signal Descriptions

Refer to figure 3-13 for a description of the microprocessor input/output signals.

Input/Output Instructions

When I = 6 and N = 1, 2, 3, 4, 5, 6, or 7, the memory byte addressed by Register (R) designated by Register (X) is placed on the data bus. The three N Output Lines are low at all times except when an I/O instruction is being executed. The I/O system recognizes these conditions and reads the output byte from the memory data lines during PSMP timing counts T4 through T7. The Registers (R) and (X) are within the PSMP chip. Register (R) is incremented by 1 during the operation to point to the next memory location.

When I = 6 and N = 9, A, B, C, D, E, or F, an input byte replaces the memory byte addressed by register (R) designated by Register (X). The input byte is also placed in the (D) register of the PSMP. Register (R) is not incremented during this instruction. The three bits of N are sent from the PSMP to the I/O system. The I/O system should gate an input byte onto the data lines during the PSMP timing counts T4 through T7.

Refer to figure 3-14 for a description of the I/O commands used by the PSMP.

INTERBOARD SIGNALS

All the pins on the output of the MTS-1 logic boards are connected to the corresponding pins on all the other logic boards. This arrangement constitutes a bus with common signals between the logic boards. The following signals are shared, used or generated by the Processor and Storage board. Refer to Interboard Signals, figure 3-15.

The two video signals are not used on the Processor and Storage board. The following definitions pertain to figure 3-15.

QuP is the Q bit from the Processor Storage Microprocessor (PSMP). N0, N1, and N2 are the I/O command output signals, CMOS levels, from the PSMP. INTERRUPT' is an input to the PSMP.

SHIFTCYCLE' and DATACOMINT' are inputs from the Serial I/O board and go to EF2' and EF4' respectively on the PSMP.

RESET' is a P/S output signal which remains low during the power-up condition.

IOCLOCK and IODATA are not used.

CHANIN' and CHANOUT' are inputs to the DMAIN' and DMAOUT' pins of PSMP.

CHANACK is the acknowledge that PSMP has entered a DMA state. The channel request signal should then be removed if only a single transfer is to occur.

MCLOCK is a 3.06 MHz system clock generated on the Video board.

BTPB' is the inverted TPB signal from PSMP. It is to be used for microprocessor synchronization.

READ' and WRITEP' are sourced by the Processor and Storage board and the Video board according to the interboard timing convention.

1. Long-Branch, Long-Skip and No Op instructions are the only instructions that require three cycles to complete (1 fetch + 2 execute).

   Long-Branch instructions are three bytes long. The first byte specifies the condition to be tested; and the second and third byte, the branching address.

   The long-branch instructions can:

   a) Branch unconditionally
   b) Test for D=0 or D≠0
   c) Test for DF=0 or DF=1
   d) Test for Q=0 or Q=1
   e) Effect an unconditional no branch

   If the tested condition is met, then branching takes place; the branching address bytes are loaded in the high-and-low-order bytes of the current program counter, respectively. This operation effects a branch to any memory location.

   If the tested condition is not met, the branching address bytes are skipped over, and the next instruction in sequence is fetched and executed. This operation is taken for the case of unconditional no branch.

2. The short-branch instructions are two bytes long. The first byte specifies the condition to be tested, and the second specifies the branching address.

   The short-branch instructions can:

   a) Branch unconditionally
   b) Test for D=0 or D≠0
   c) Test for DF=0 or DF=1
   d) Test for Q=0 or Q=1
   e) Test the status (1 or 0) of the four EF flags
   f) Effect an unconditional no branch

   If the tested condition is met, then branching takes place; the branching address byte is loaded into the low-order byte position of the current program counter. This effects a branch with the current 256-byte page of the memory, i.e., the page which holds the branching address. If the tested condition is not met the branching address byte is skipped over, and the next instruction in sequence is fetched and executed. This same action is taken in the case of unconditional no branch.

3. The skip instructions are one byte long. There is one Unconditional Short-Skip (SKP) and eight Long-Skip instructions.

   The Unconditional Short-Skip instruction takes 2 cycles to complete (1 fetch + 1 execute). Its action is to skip over the byte following it. Then the next instruction in sequence is fetched and executed. This SKP instruction is identical to the unconditional no-branch instruction (NBR) except that the skipped-over byte is not considered part of the program.

Figure 3-12. Additional Instruction Information (Sheet 1 of 2)

The Long-Skip instructions take three cycles to complete (1 fetch + 2 execute).

They can:

a) Skip unconditionally
b) Test for D=0 or D≠0
c) Test for DF=0 or DF=1
d) Test for Q=0 or Q=1
e) Test for 1E=1

If the tested condition is met, then Long-Skip takes place; the current program counter is incremented twice. Thus two bytes are skipped over and the next instruction in sequence is fetched and executed. If the tested condition is not met, then no action is taken. Execution is continued by fetching the next instruction in sequence.

Figure 3-12. Additional Instruction Information (Sheet 2 of 2)

BUS 0 to BUS 7 (Data Bus)	8-bit directional DATA BUS lines. These lines are used for transferring data between the memory, the microprocessor, and I/O devices.
N0 to N2 (I/O Command)	Issued by an I/O instruction to signal the I/O control logic of a data transfer between memory and I/O interface. These lines can be used to issue command codes or device selection codes to the I/O devices (independently or combined with the memory byte on the data bus when an I/O instruction is being executed). The N bits are low at all times except when an I/O instruction is being executed. During this time their state is the same as the corresponding bits in the N register. The direction of data flow is defined in the I/O instruction by bit N3 and is indicated by the level of the MRD signal. $\overline{MRD} = V_{CC}$: Data from I/O to CPU and Memory $\overline{MRD} = V_{SS}$: Data from Memory to I/O
$\overline{EF1}$ to $\overline{EF4}$ (4 Flags)	These levels enable the I/O controllers to transfer status information to the processor. The levels can be tested by the conditional branch instructions. They can be used in conjunction with the INTERRUPT request line to establish interrupt priorities. These flags can also be used by I/O devices to "call the attention" of the processor, in which case the program must routinely test the status of these flag(s). The flag(s) are sampled at the beginning of every S1 cycle.

Figure 3-13. Microprocessor Signal Descriptions (Sheet 1 of 4)

$\overline{\text{INTERRUPT}}$, $\overline{\text{DMA-IN}}$, $\overline{\text{DMA-OUT}}$
(3 I/O Requests)

These signals are sampled by the microprocessor during the interval between the leading edge of TPB and the leading edge of TPA.

Interrupt Action: X and P are stored in T after executing current instruction; designator X is set to 2; designator P is set to 1; interrupt enable is reset to 0 (inhibit); and instruction execution is resumed.

DMA Action: Finish executing current instruction; R(0) points to memory area for data transfer; data is loaded into or read out of memory; and increment R(0).

NOTE: In the event of concurrent DMA and INTERRUPT requests, DMA-IN has priority followed by DMA-OUT and then INTERRUPT.

SC0, SC1,
(2 State Code Lines)

These lines indicate that the CPU is: 1) fetching an instruction, or 2) executing an instruction, or 3) processing a DMA request, or 4) acknowledging an interrupt request. The levels of state code are tabulated below. All states are valid at TPA.

$H = V_{CC}$ $L = V_{SS}$

State Type	State Code Lines	
	SC1	SC0
S0 (Fetch)	L	L
S1 (Execute)	L	H
S2 (DMA)	H	L
S3 (Interrupt)	H	H

TPA, TPB
(2 Timing Pulses)

Positive pulses that occur once in each machine cycle (TPB follows TPA). They are used by I/O controllers to interpret codes and to time interaction with the data bus. The trailing edge of TPA is used by the memory system to latch the higher-order byte of the 16-bit memory address. TPA is suppressed in IDLE when the CPU is in the load mode.

MA0 to MA7
(8 Memory Address Lines)

The higher-order byte of a 16-bit memory address appears on the memory address lines MA0-7 first. Those bits required by the memory system are strobed into external address latches by timing pulse TPA. The low-order byte of the 16-bit address appears on the address lines after the termination of TPA. Latching of all 8 higher-order address bits would permit a memory system of 64K bytes.

$\overline{\text{MWR}}$ (Write Pulse)

A negative pulse appearing on a memory-write cycle, after the address lines have stabilized.

Figure 3-13. Microprocessor Signal Descriptions (Sheet 2 of 4)

$\overline{MRD}$ (Read Level)

A low level on $\overline{MRD}$ indicates a memory read cycle. It can be used to control three-state outputs from the addressed memory which may have a common data input and output bus. If a memory does not have a three-state high-impedance output, $\overline{MRD}$ is useful for driving memory/bus separator gates. It is also used to indicate the direction of data transfer during an I/O instruction:

$\overline{MRD} = V_{CC}$: Data from I/O to CPU and memory

$\overline{MRD} = V_{SS}$: Data from Memory to I/O

Q

Single bit output from the CPU which can be set or reset under program control. During SEQ or REQ instruction execution, Q is set or reset between the trailing edge of TPA and the leading edge of TPB.

CLOCK

Input for externally generated single-phase clock. A typical clock frequency is 6.4 MHz at $V_{CC} = V_{DD} = 10$ volts.

The clock is counted down internally to 8 clock pulses per machine cycle.

$\overline{XTAL}$

Connection to be used with clock input terminal, for an external crystal, if the on-chip oscillator is utilized. The crystal is connected between terminals 1 and 39 (CLOCK and $\overline{XTAL}$) in parallel with a resistance (10 megohms typ.). Frequency trimming capacitors may be required at terminals 1 and 39.

$\overline{WAIT}$, $\overline{CLEAR}$
(2 Control Lines)

Provide four control modes as listed in the following truth table:

$\overline{CLEAR}$	$\overline{WAIT}$	MODE
L	L	Load
L	H	Reset
H	L	Pause
H	H	Run

The function of the modes are defined as follows:

Load

Holds the CPU in the IDLE execution state and allows an I/O device to load the memory without the need for a "bootstrap" loader. It modifies the IDLE condition so that DMA-IN operation does not force execution of the next instruction.

Figure 3-13. Microprocessor Signal Descriptions (Sheet 3 of 4)

Reset

Registers I, N, Q are reset, IE is set and 0's (V_{SS}) are placed on the data bus. TPA and TPB are suppressed while reset is held and the CPU is placed in S1. The first machine cycle after termination of reset is an initialization cycle. During this cycle the CPU remains in S1 and registers X, P, and R(0) are reset. Interrupt and DMA servicing are suppressed during the initialization cycle.

The next cycle is an S0, S1, or an S2 but never an S3. With the use of a 71 instruction followed by 00 at memory locations 0000 and 0001, this feature may be used to reset IE, so as to preclude interrupts until ready for them. Power-up reset can be realized by connecting an external RC to $\overline{\text{CLEAR}}$.

Pause

Stops the internal CPU timing generator on the first negative high-to-low transition of the input clock. The oscillator continues to operate, but subsequent clock transitions are ignored.

Run

May be initiated from the Pause or Reset mode functions. If initiated from Pause, the CPU resumes operation on the first negative high-to-low transition of the input clock. When initiated from the Reset operation, the first machine cycle following Reset is always the initialization cycle. The initialization cycle is then followed by a DMA (S2) cycle or fetch (S0) from location 0000 in memory.

Figure 3-13. Microprocessor Signal Descriptions (Sheet 4 of 4)

Microprocessor Instruction	Function
OUTPUT 1	Load SERIAL I/O data register.
INPUT 1	Read SERIAL I/O data register.
OUTPUT 2	Load SERIAL I/O I-CODE and initiate a shift.
INPUT 2	Turn off SERIAL I/O Invitation to Interrupt.
INPUT 3	Initiate an alarm signal.
OUTPUT 4	Not used.
INPUT 4	Not used.
OUTPUT 5	Not used.
INPUT 5	Not used.
OUTPUT 6	Load the Data Comm input data register and set EF1 of the SIO microprocessor.
INPUT 6	Read the Data Comm output Data register, and clear the interrupt flag.
OUTPUT 7	Send data, clock, and control bits to EAROM.
INPUT 7	Read EAROM data bit.

Figure 3-14. PSMP Input/Output Instructions Definitions

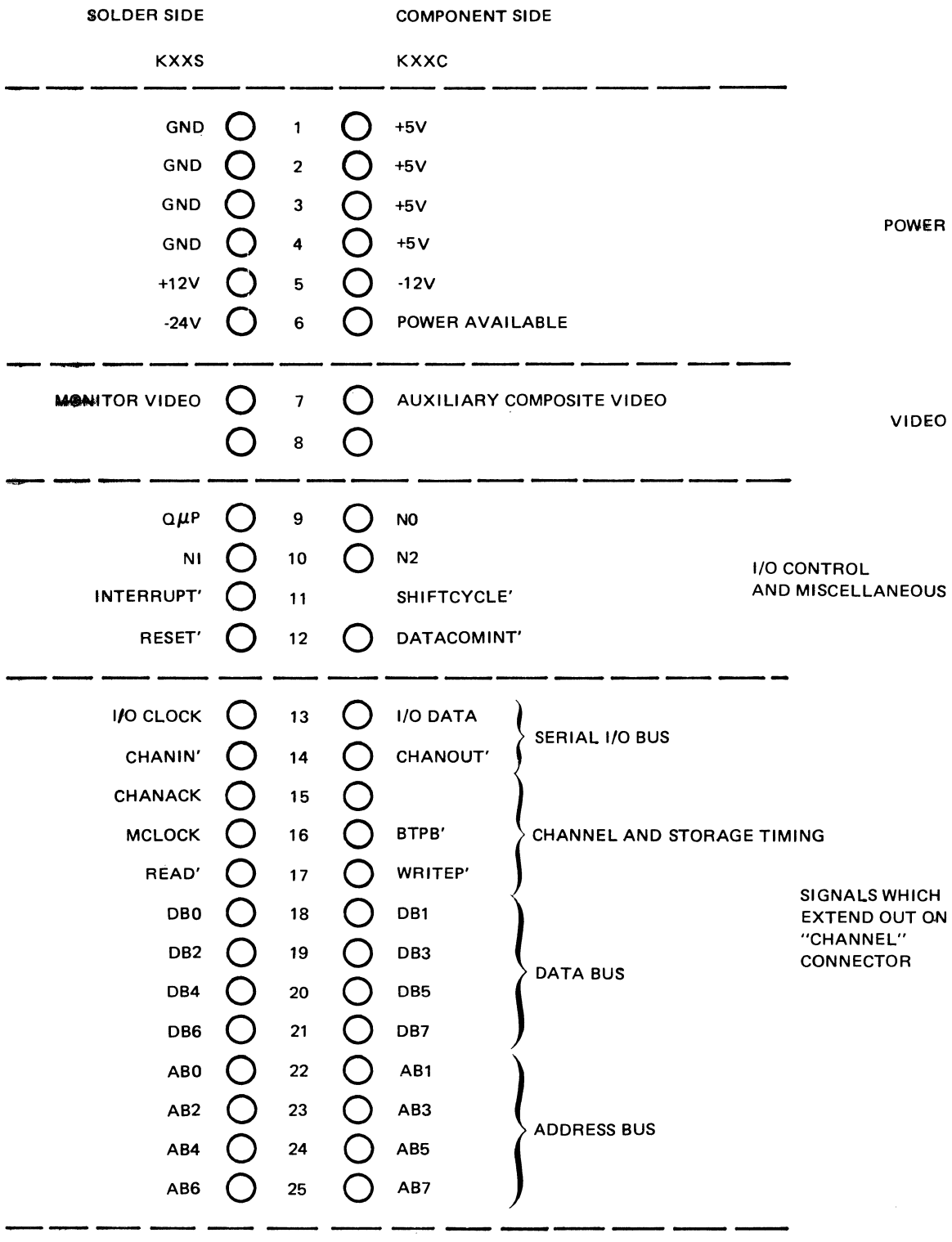


Figure 3-15. Interboard Signals

DB0-DB7 is a tristate bus shared by the Processor and Storage board, the Video board, I/O devices, channel devices, and external storage.

AB0-AB7 are sourced by the Processor and Storage board and the Video board according to the interboard timing convention.

Interboard Bus Timing Convention

The basic reference for timing within the MTS-1 is the MCLOCK Signal generated on the Video logic board. Figure 3-16, System Timing Convention, illustrates the following information. MCLOCK is a square wave with a repetition rate of approximately 326 nanoseconds. Eight clock periods are required for one microprocessor machine cycle. These periods are called T0 through T7.

The clock edge that causes signat TPB to be generated in the microprocessor defines the beginning of period T7. Thus each logic board may use the PSMP T7 signal and MCLOCK for synchronization. Each logic board will generate levels corresponding to time periods T0 through T7.

The microprocessors are designed to accept from, or provide, information to memory at TPB time. This time is defined a time period T7.

By selecting memory (either ROM or RAM) that has a Read or Write cycle time of less than one-half of the full cycle time of the microprocessor and controlling the memory timing signals, two microprocessors may share the same memory with no reduction in their processing capabilities. This concept is used in the MTS-1.

By synchronizing the Display microprocessor machine cycle out of phase by one-half of the PSMP, its S7 time will occur during T3 time of the PSMP. (Refer to section 2, Microprocessor Synchronization.) Therefore, the Display microprocessor may access memory during its timing periods S4 - S7, which correspond to the PSMP timing periods T0 - T3. The PSMP will access memory during the timing periods T4 - T7, which correspond to the Display microprocessor timing periods S0 through S3. In figure 3-16, System Timing Convention, the microprocessor signals illustrate the timing signals for the PSMP. The interboard signals illustrate the timing relationship for the interaction of the Display microprocessor and the PSMP. Figure 3-17 defines the timing convention to achieve the interaction of the processor.

All input, output, and external data transfers occur under the control of the PSMP. Any device requesting an input to memory only gates its information onto the data lines at timing periods T4 through T7. Any device requesting an output from memory will be provided valid data at the end of timing period T7.

The microprocessor N lines will become valid at approximately T0 and remain valid through T7. Requests for memory access may occur at the falling edge of any MCLOCK pulse; however, the Channel Acknowledge signal will occur in the T4 through T7 interval. BTPB' may not be used as a timing signal, as it occurs approximately 200 nanoseconds after timing period T7 starts. BTPB' may be used to generate local synchronous timing signals T0 through T7. The decoded outputs from the generated timing signals T4 through T7 are used for gating input, output, and external data transfers requested via the PSMP.

Figure 3-18, Storage Timing - Processor and Storage Board, illustrates the relationship between the interboard signals and the memory storage timing signals.

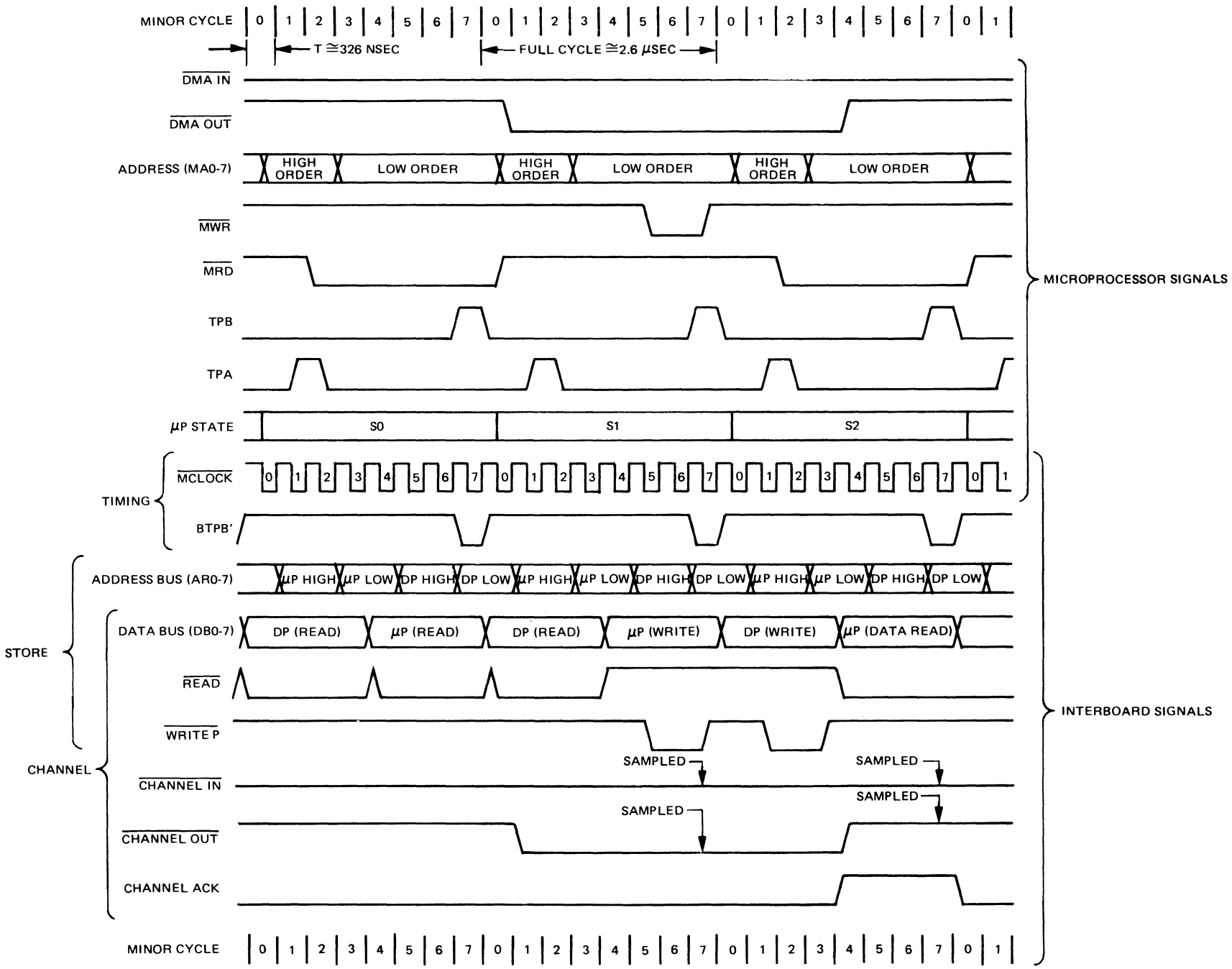


Figure 3-16. System Timing Convention

Interval	Data Bus Control, READ' and WRITEP' Source
TØ-T3	Video board
T4-T7	Processor Store
	Address Bus Source
T5-T6	Video board high address
T7-TØ	Video board low address
T1-T2	Processor Store high address
T3-T4	Processor Store low address

Figure 3-17. Timing Period Definition

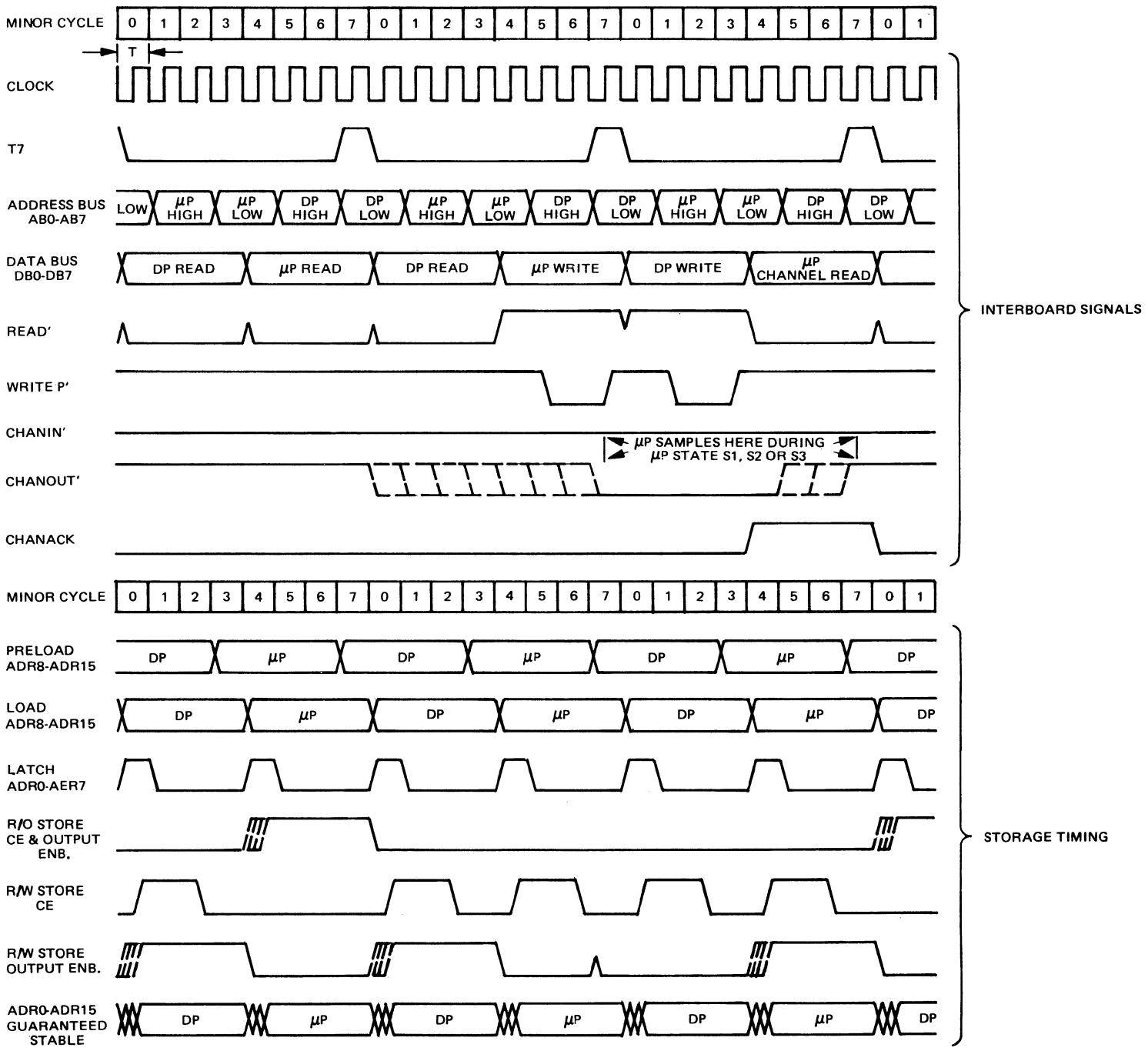


Figure 3-18. Storage Timing - Processor and Storage Board

SECTION 4
SERIAL INPUT/OUTPUT BOARD

FUNCTIONAL DESCRIPTION

The Serial Input/Output (SIO) board is comprised of two sections. One section contains the Serial Input/Output Generator logic; the other contains the Data Communications Control logic. The Serial Input/Output Generator logic provides the Serial Input/Output Interface between the Processor and Storage Microprocessor (PSMP) and the Serial Input/Output Interface-compatible peripheral units. The Data Communications Control logic provides the interface necessary for the MTS-1 to communicate either asynchronously or synchronously with a data communications network.

Serial Input/Output Interface

The Serial Input/Output Interface is designed to provide a simple, low-speed, short distance transmission interface. Peripheral units using the Serial Input/Output Interface are connected in a series fashion.

The interface has the capability of addressing up to 255 individual units. It transfers information at a maximum rate of 500 characters a second for distances to a maximum of 200 feet.

The interface cable consists of a clock signal, a data signal, and signal ground. Figure 4-1 illustrates the information on these two lines.

Clock Signal

The clock signal is a square wave with a frequency of 384 kHz, or having a repetition rate of 2606 nanoseconds. The clock signal is provided by the MTS-1 for use by the peripheral units for synchronization of the data signal. The clock signal is developed in the SIO Generator logic using BTPB' signal and the 3.06 megahertz clock signal provided by the Video board. As shown in figure 4-1, the trailing edge of the clock pulse is coincident with the leading edge of each data pulse.

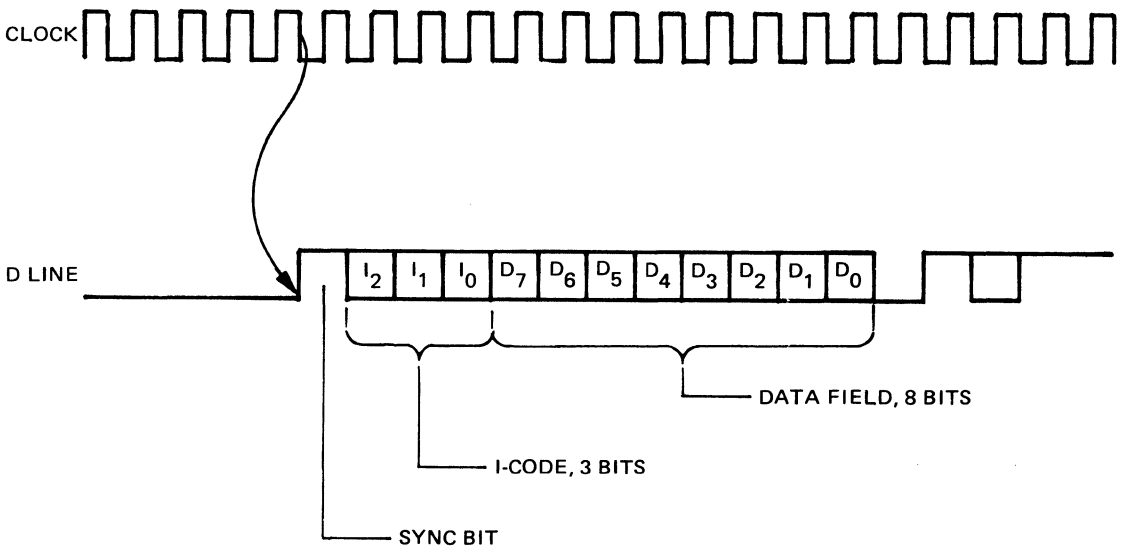


Figure 4-1. Character Transmission Data

Data Signal Line

The information on the data signal line is transmitted in bit serial form, and is divided into three sections. The sections consist of the Synchronizing bit, the Instruction code (I-code of three bits), and the Data field (eight bits).

SYNCHRONIZING BIT

The synchronizing bit represents a signal transition from a 0, or low signal, to a 1, or high signal, and signifies the beginning of a transmission.

DATA FIELD

The Data field is eight bits in length. The information in this field is clock synchronous and is provided by either the TD850 or the peripheral units, depending upon I code. D7 is the most-significant bit and D0 is the least significant bit.

INSTRUCTIONS (I-CODE)

Each transmission contains an instruction code. This code determines the direction data transfer and type of data being transferred. The bit referred to as I2 in figure 4-1 is the most-significant bit, and I0 is the least significant bit. The following are all of the I-codes and the description of each.

a. Invitation to Interrupt (I-code = 000)

   This instruction requests all the peripheral units which have a pending interrupt, to send their 8-bit unit number in the Data field. Contention is resolved by requiring each peripheral unit to monitor the Data line as it applies its 8-bit unit number to the Data line. If another peripheral unit has put a 1, or High, on the Data line while this peripheral unit is trying to insert a 0, or Low, this peripheral unit must stop trying to put its unit number on the Data line. This is because a unit with a higher priority (indicated by a larger unit number) is also trying to interrupt. A successful interrupt is indicated in subsequent instructions issued by the MTS-1.

b. Invitation to Select (I-code = 001)

   This instruction selects a device for data transmission activity. Every peripheral unit compares its unit number with the unit number inserted by the MTS-1 in the Data field. The peripheral unit which successfully compares the two numbers becomes selected. Once the peripheral unit becomes selected, it responds to the transmission activity instructions GET and SEND. The peripheral unit remains selected until it detects receipt of either the Invitation to Interrupt instruction or the Invitation to Select instruction.

c. GET (I-code = 010, 100, 110)

   The GET instructions cause a selected peripheral unit to transmit data to the MTS-1. The peripheral unit transmits one byte of data (most significant bit first) to the MTS-1. Note that only one 8-bit byte of information is transferred with each command. There are three codes that correspond to the GETA, GETB, and GETC instructions. This allows the data being transmitted to be arranged into three different fields.

d. SEND (I-code = 011, 101, 111)

   The SEND instructions cause a selected peripheral unit to receive data from the MTS-1. The peripheral unit takes the input data from the Data field, the most significant bit first. There are three codes that correspond to the SENDA, SENDB, and SENDC instructions. This allows the separation of the data being received into three fields or groups.

The block diagrams shown in figures 4-2 and 4-3 illustrate the Serial Input/Output logic used on the SIO board (Serial Input/Output Generator, figure 4-2), and in the Keyboard (Keyboard Serial Input/Output, figure 4-3). The PSMP communicates with the SIO Generator logic using Input/Output microprocessor instructions. The PSMP loads eight bits of data into the SIO Generator shift register by executing an Output 1 instruction. The PSMP takes eight bits of data information from the SIO Generator shift register by executing an Input 1 instruction. The PSMP loads the three bits of I-code into the SIO Generator shift register by executing an OUTPUT 2 instruction. The PSMP can interrupt the Invitation to Interrupt operation by executing an INPUT 2 instruction. The SIO Generator logic can generate an interrupt signal to the PSMP when it requires servicing. To begin operation, the PSMP loads 0's into the SIO Generator shift register data portion by executing an OUTPUT 1 instruction. The PSMP then loads the I-code 000 for Invitation to Interrupt instruction into the I-code portion of the SIO Generator shift register by executing an OUTPUT 2 instruction. This action allows the shift register to begin serially shifting the information just loaded out on the data line. This data output will also be applied to the signal input of the SIO Generator shift register. This serial shift will continue until interrupted by the request for interrupt from the Keyboard or upon the receipt of an INPUT 2 instruction from the PSMP.

Assume a normal key is pressed on the keyboard (figure 4-3). A normal key is other than one of the following: the Reverse Tabulation Key (RTAB), which generates signal F1; the Character Insert Key (CHARINS), which generates signal F2'; or a blue control key, which only generates a 10 microsecond strobe. For the non-blue keys, the strobe pulse width is equal to the length of time the key is pressed. The key decode signals, B1' through B9', are fed to the inputs of the data multiplexors to be gated through the Transmitter and Receiver logic to the interface Data line. The strobe signal causes the interrupt signal, which indicates to the keyboard select logic the need to respond to the Invitation to Interrupt instruction by placing the keyboard unit number on the Data line.

With the receipt of the Invitation to Interrupt instruction, the I-code Select and the Keyboard Select logic cause the Keyboard unit number to be gated onto the Data line. When the SIO Generator logic detects a 1 on the Data line, it senses that a 1 occurred, causes the SIO Generator shift register to stop shifting at the end of this cycle, and causes an interrupt signal to be sent to the PSMP.

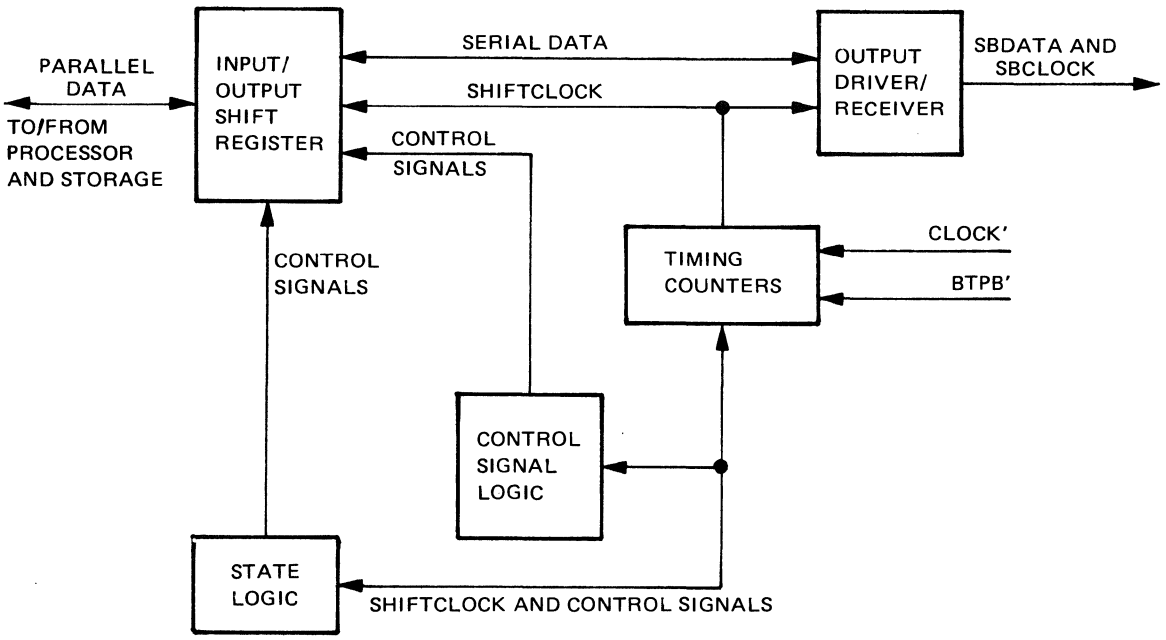


Figure 4-2. Serial Input/Output Generator Simplified Block Diagram

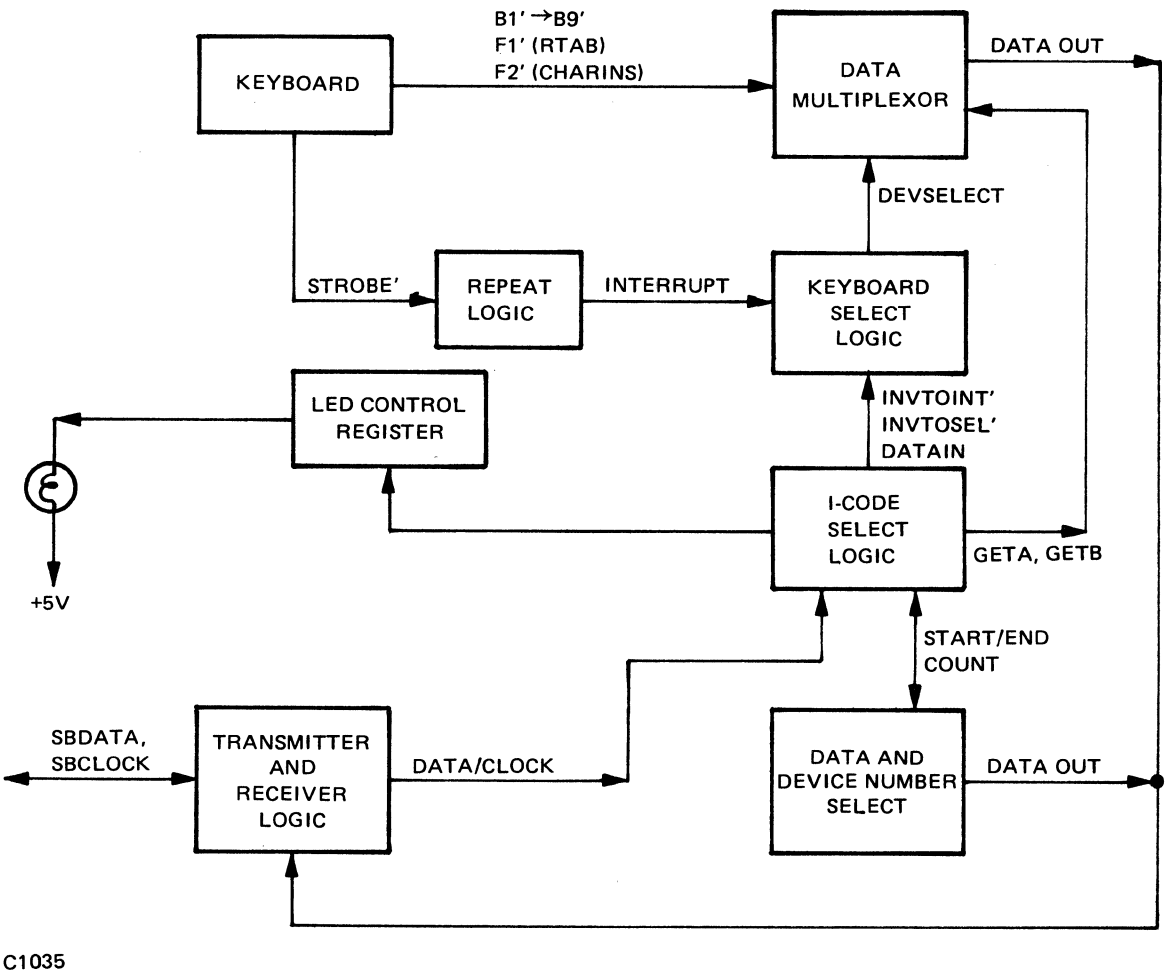


Figure 4-3. Keyboard Serial Input/Output Simplified Block Diagram

The PSMP then executes an INPUT 1 instruction to obtain the Keyboard unit number from the data portion of the SIO Generator shift register. The PSMP then executes an OUTPUT 1 instruction to load the Keyboard unit number back into the SIO Generator shift register. It then executes an OUTPUT 2 instruction to load the Invitation to Select (I-code = 001) into the I-code portion of the SIO Generator shift register. The loading of an I-code other than Invitation to Interrupt will cause the SIO Generator shift register to only shift through its serial shift cycle one time. The Keyboard I-code and Select logic compare the Keyboard unit number with the number received in the Data Field. If they compare, the Keyboard sets its selected logic and will now respond to either the GET or SEND instructions.

When the PSMP detects the end of the Invitation to Select shift cycle, it sends an OUTPUT 1 instruction, with 0's in the information, to clear the data portion of the SIO Generator shift register. The PSMP then sends a GETA instruction (I-Code = 010) to the SIO Generator shift register I-code portion by means of an OUTPUT 2 instruction. The SIO Generator shift register then shifts the GETA instruction out to the data line.

The Keyboard I-code Select logic decodes the GETA instruction and gates the first eight bits of the keyboard character decode information to the transmitter and receiver logic to be gated onto the data line.

The PSMP senses the end of the shift cycle and executes an INPUT 1 instruction to obtain the A portion of the character information. The PSMP executes an OUTPUT 1 instruction to clear the data portion of the SIO Generator shift register. The PSMP executes an OUTPUT 2 instruction to send a GETB instruction (I-code = 100) to the I-code portion of the SIO Generator shift register. The shift register starts a serial shift cycle. The I-code Select logic in the Keyboard decodes the GETB instruction, and gates the second eight bits of the character information to the Transmitter and Receiver logic to be placed on the data line.

The PSMP senses the end of the shift cycle and executes an INPUT 1 instruction to obtain the B portion of the character information. If the PSMP wishes to change the particular Keyboard Light-Emitting Diodes (LEDs) to be turned on at this time, it executes an OUTPUT 1 instruction to load the data corresponding to the desired light configuration into the data portion of the SIO Generator shift register. The PSMP then executes an OUTPUT 2 instruction to load the I-code portion of the SIO Generator shift register with the SENDA (I-code = 011). The shift register then begins another serial shift cycle. When the Keyboard I-code Select logic decodes the SENDA instruction and finds it is already selected, it then collects the incoming data in a serial-to-parallel register; at the completion of the serial shift cycle, the output of the register causes the desired LEDs to light.

When the PSMP detects the completion of the serial shift cycle, it executes an OUTPUT 1 instruction to clear the data portion of the SIO Generator shift register. The PSMP then executes an OUTPUT 2 instruction to load the Invitation to Interrupt (I-code = 000) instruction into the I-code portion of the SIO Generator shift register. The shift register shifts out the Invitation to Interrupt instruction until another peripheral unit number is inserted in the data field or the PSMP executes an INPUT 2 instruction. When the Keyboard I-code Select logic detects the Invitation to Interrupt instruction, it clears its selected indication and is ready for another sequence of events.

As indicated by the above information, the Keyboard sends two 8-bit bytes of data for each Keyboard key that is pressed. The first word contains Keyboard decoder output bits B8' through B1' shifted out most significant bit, B8', first. The second 8-bit byte contains the following information:

Bit	Value
8	1
7	1
6	1
5	
4	RTAB (signal F2)
3	CHARINS (signal F1)
2	Strobe' (Z506H)
1	Repeat' (NEWCHAR)

The values of bits 1 and 2 are decoded by firmware as follows:

Bit 2	Bit 1	Definition
0	0	Repeat
0	1	Normal Hit
1	0	Blue Key
1	1	N/A

Data Communications Control

The B 9348 will have an asynchronous Serial Input/Output (SIO) interface board. The MTS-1 hardware is capable of interfacing in an asynchronous or synchronous mode by using either an asynchronous or synchronous Serial Input/Output board. Further reference to synchronous hardware is not applicable to the B 9348 Operator Display System. The difference between the two versions is in the Data Communication line interface components that are installed on the board. The control logic is made up of the Data Communications Microprocessor (DCMP), two kilobytes of RAM, 32 bytes of ROM, Bus and PSMP interface control logic, and the Bus and control logic necessary to interface to the Data Communications line.

The DCMP communicates with the PSMP through a hardware interface. The ROM contains a small bootstrap program to establish the initial communication of the DCMP with the PSMP. Once this communication is established, the DCMP transfers the firmware necessary for the DCMP operation from the Processor and Storage board ROM to the RAM on the SIO board. After this transfer is complete, the bootstrap program causes the DCMP to branch to the firmware loaded in the SIO RAM to begin the Data Communication functions. Figures 4-4 and 4-5 give a block diagram of the SIO board.

PSMP and DCMP Hardware Interface

This interface handles control signals, flags, and information transfer between the microprocessors. The signals used are shown in figure 4-4 going to the Interboard bus. Figure 3-15, Interboard signals, shows the backplane locations of the signals. Data registers are used to transfer data information between the microprocessors. The flags are used to develop the request and acknowledge signals.

There are two modes for passing data between the two microprocessors. The first uses Input/Output (I/O) instructions of each microprocessor to pass the data. The second uses the Input/Output instructions of the DCMP and the DMA facility of the PSMP. The SIO Interface Control register is used to define which mode of transfer is going to be used.

INPUT/OUTPUT TRANSFER MODE

In this mode, the I/O instructions of each processor are used to transfer data. To configure the interface hardware for the mode, bit 0, SETINT, of the SIO Interface Control register must be set and all other bits cleared.

Data From the DCMP To the PSMP

Data to be sent to the PSMP is loaded in the SIO Output register by a DCMP OUTPUT 1 instruction. In this mode of transfer, loading the SIO Output register automatically causes the interrupt flag to the PSMP to be set. (Note that this flag, in addition to causing an interrupt of PSMP, if PSMP is interrupt enabled, can also be tested by PSMP as flag EF4.)

The PSMP responds to the interrupt by reading the SIO Output register with an INPUT 6 instruction. The occurrences of a PSMP INPUT 6 instruction automatically clears the interrupt flag.

Data From the PSMP To the DCMP

Data to be sent from the PSMP to the DCMP in this mode is loaded in the SIO Input register with a PSMP OUTPUT 6 instruction. The occurrence of this instruction automatically sets a flag (EF1) to the DCMP. (See EXTDATRDY.) The DCMP then reads the SIO Input register with an INPUT 1 instruction, which automatically clears the flag.

Hardware Handshaking Input/Output Mode

Note that no real hardware handshaking takes place in this mode of transfer. In each case where a processor loads the data to be sent to the other processor, there is no way for the sending processor to detect the fact that the receiving processor has taken the data; i.e., Processor DCMP cannot detect that the interrupt flag has been cleared, and processor PSMP cannot detect that the flag it caused to set (EXTDATRDY) has been cleared.

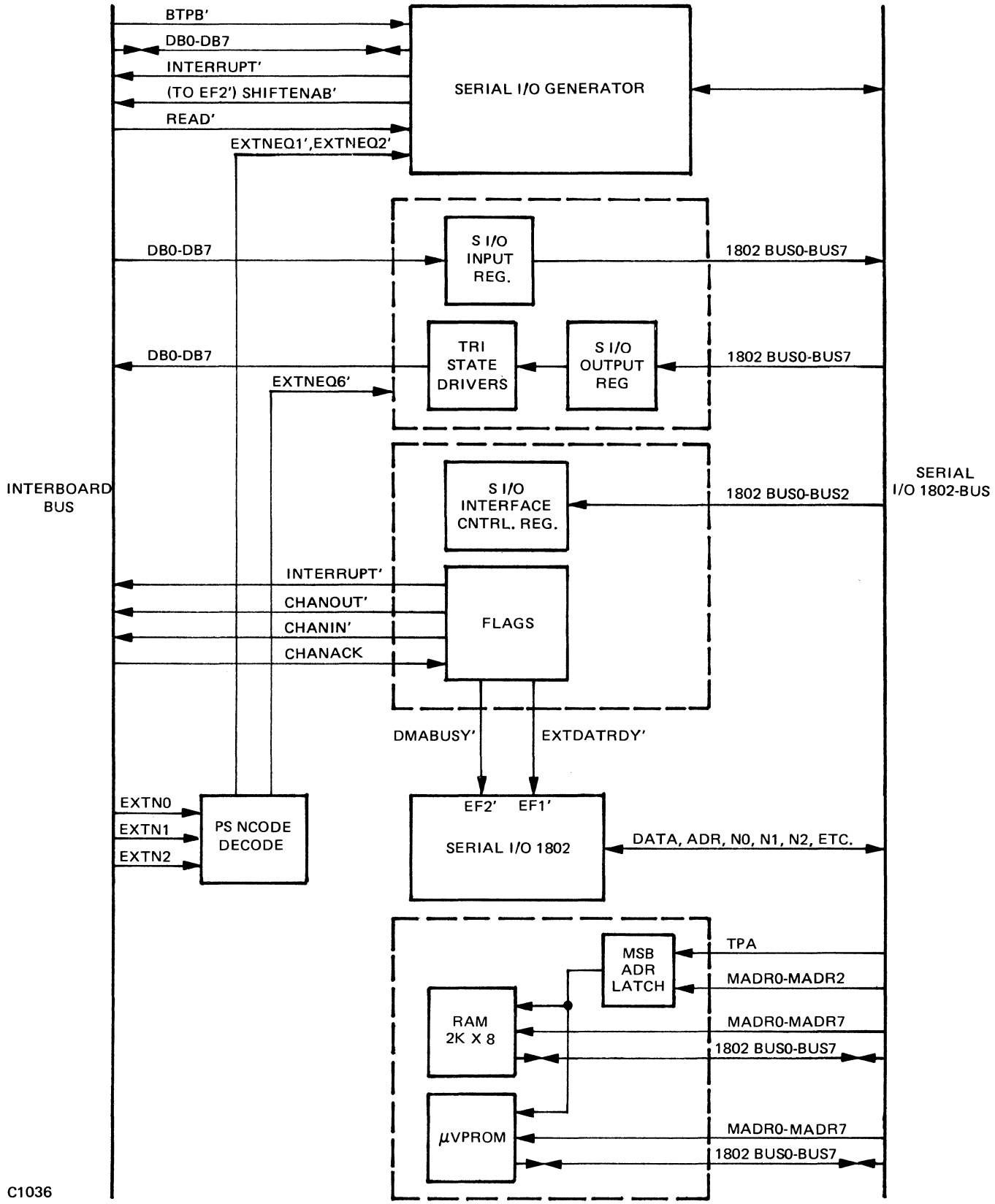


Figure 4-4. SIO Microprocessor and Control Logic Block Diagram

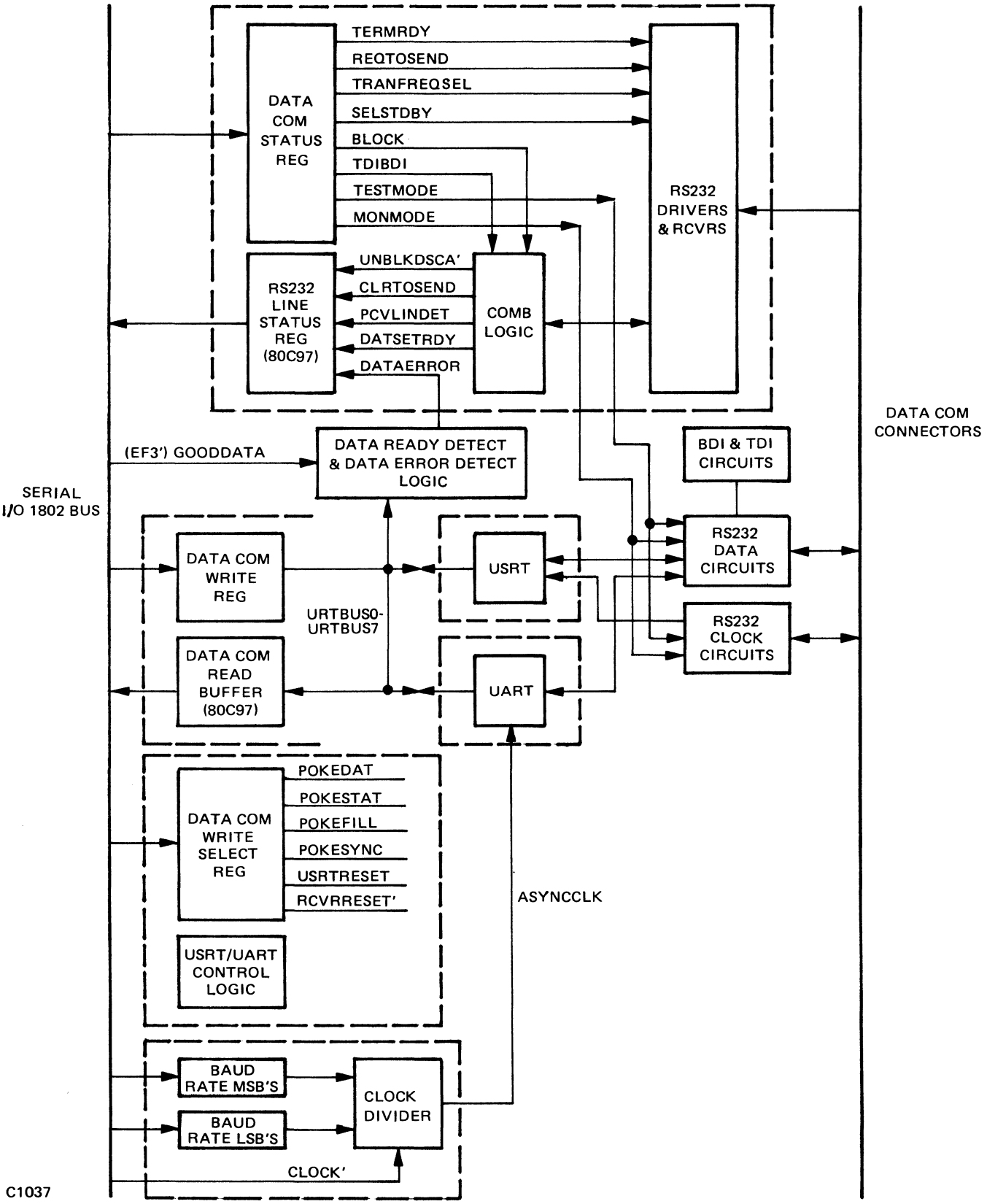


Figure 4-5. SIO Data Communications Interface Logic Block Diagram

In normal operation of the SIO board, the absence of hardware handshaking presents no problem because the firmware handshaking protocol that has been established assures that no problems will occur. The basic philosophy is that one processor will never attempt to send more than one word without then receiving a response from the other processor, in the form of a data transfer in the opposite direction.

DMA Transfer Mode

This mode of data transfer uses I/O instructions of the DCMP and the DMA facility of the PSMP. To configure the interface hardware for this mode, the SIO Interface Control register should be set as follows:

a. Bit 0, SETINT: Cleared.

b. Bit 1, DMAOUT: Set if data from PSMP.

c. Bit 2, DMAIN: Set if data to PSMP.

Note that in the DMA mode of transfer, the PSMP must have its R(0) register loaded with the address of the memory area to be used by the DMA mechanism. (In normal operation, the operation of setting up R(0) will be initiated by a communication between the two processors via the I/O mode of transfer.)

Data From the DCMP to the PSMP

Data to be sent to PSMP is loaded in the SIO Output register by a DCMP OUTPUT 1 instruction. In this mode of transfer (data from DCMP to PSMP via DMA) loading the SIO Output register automatically causes the CHANIN line to PSMP to be developed, which in turn will automatically cause a DMA cycle in PSMP. CHANIN also causes a DMABUSY signal to be developed. The DMA BUSY is tested as flag EF2 of the DCMP; thus the DCMP can test to see if a DMA operation is in progress. When the DMA cycle of PSMP generates CHANACK, the CHANIN, and thus DMABUSY are cleared. The DCMP can then determine that the DMA operation is complete and the SIO Output register is available for loading to initiate a subsequent DMA cycle if desired.

Data From the PSMP to the DCMP

This mode is unique from the ones previously defined in that even though the direction of data transfer is from the PSMP, it is the task of the DCMP to initiate the actual data transfer. (Note, however, in normal operation the PSMP will probably issue a request to the DCMP, via the I/O mode, that will cause the DCMP to then start requesting data via the DMA mode of transfer.)

To initiate a transfer in this mode, the DCMP Output register causes CHANOUT to the PSMP to be set. (Note that the data loaded in SIO Output register is meaningless; it just causes CHANOUT to be set.) The PSMP responds by entering a DMA cycle. CHANOUT also asserts DMABUSY, which can be tested as EF2 by the DCMP to determine if a DMA cycle is in progress. When CHANACK occurs, it causes the SIO Input register to be loaded with the desired data, and clears CHANOUT, thus causing DMABUSY to clear, allowing the DCMP to sense that the DMA operation is complete. The DCMP must then read the SIO Input register and, if desired, initiate another transfer by loading the SIO Output register again.

Hardware Handshaking DMA Mode

In the DMA mode of transfer, there is an actual hardware handshaking mechanism in the form of the DMABUSY signal. Its reliability stems from the fact both processors run at the same clock rate; however, it is not really a good asynchronous link.

Programming the Data Communications Logic

The following are general techniques that should be used when programming the Data Communications logic on the SIO board. Included are general descriptions of the logic with respect to Input and Output instructions, initialization of the UART or USRT, transmit mode programming, receive mode programming, test mode programming and monitor mode programming.

Refer to figure 4-6 for DCMP Input/Output instruction assignment and flags. Figure 3-15 gives the PSMP Input/Output instructions assignment. Figure 4-7 contains a brief description of the registers that are loaded and read by the DCMP Input/Output instructions. Figure 4-8 gives detailed descriptions of the various SIO Input/Output registers.

LOGIC OVERVIEW

The Data Comm logic that must be manipulated through use of DCMP Input/Output instructions can be divided into three functional categories:

a. RS232 control circuits.

b. UART and USRT logic.

c. Asynchronous clock generation logic.

Instruction	Function
OUTPUT 1	Loads SIO Output register; causes one of the following to be set: INTERRUPT, CHANOUT, or CHANIN.
INPUT 1	Reads SIO Input register, and clears flag EF1.
OUTPUT 2	Loads SIO Interface Control register.
INPUT 2	Not implemented.
OUTPUT 3	Loads Data Communications Status register.
INPUT 3	Reads RS232 line status.
OUTPUT 4	Loads Data Communications Write Buffer.
INPUT 4	Reads Data Communications Data.
OUTPUT 5	Loads Data Communications Write Select register.
INPUT 5	Reads USRT/UART status.
OUTPUT 6	Loads ASYNC Baud Rate register (8 least significant bits).
INPUT 6	Not implemented.
OUTPUT 7	Loads ASYNC Baud Rate register (4 most significant bits).
INPUT 7	Not implemented.
FLAGS	
$\overline{EF1}$	(PSMP DATA READY) '
$\overline{EF2}$	(DMA BUSY) '
$\overline{EF3}$	(GOOD DATA) '
$\overline{EF4}$	(LAST CHAR SENT) '

Figure 4-6. DCMP Input/Output Instruction and Flags

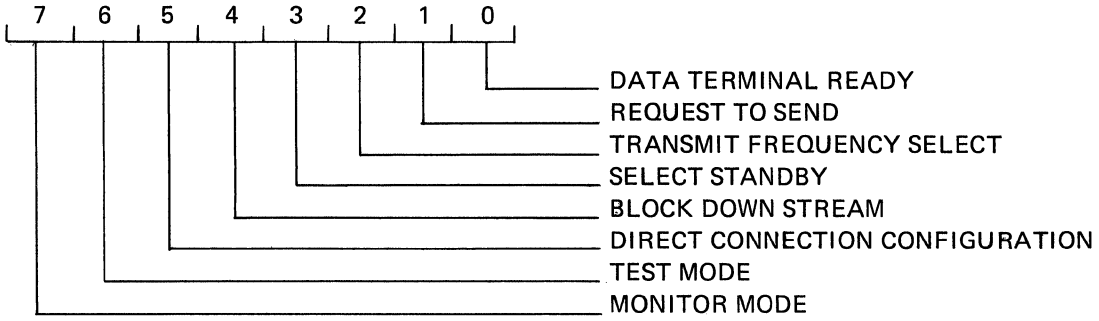
OUTPUT 1	Loads SIO Output register. This register is used to pass data to the PSMP. Loading it causes INTERRUPT, CHANIN or CHANOUT to be developed, depending on the contents of the SIO Interface control register.
INPUT 1	Reads SIO Input register. The SIO Input register is used to receive data from the PSMP. Reading this register clears DCMP flag EF1.
OUTPUT 2	Loads SIO Interface Control register. The contents of the SIO Interface Control register are used to specify which of INTERRUPT, CHANIN or CHANOUT is developed when an Output 1 is done.
INPUT 2	Not implemented.
OUTPUT 3	Loads Data Communications Status register. The Data Communications Status register is used to set certain RS232 and CCITT control circuits, and also to configure the SIO data communications circuitry in various special modes.
INPUT 3	Reads RS232 Line Status This data (not really a register) reflects the status of certain RS232 and CCITT control circuits, as well as the state of the receive data error detection circuits.
OUTPUT 4	Loads Data Communications Write Buffer. The Data Communications Buffer is loaded with data or status destined for either the USRT or UART. The ultimate destination of the contents must have been previously designated by loading the Data Communications Write Select register.
INPUT 4	Reads Data Communications Data. This instruction reads the contents of the UART Receiver Buffer register or the USRT Receiver Output register.

Figure 4-7. General DCMP Input/Output Register Descriptions (Sheet 1 of 2)

OUTPUT 5	Loads Data Communications Write Select register. The contents of the Data Communications Write Select register are used either to specify the destination of data loaded in the Data Communications Write Buffer or to directly cause resets of the UART or USRT.
INPUT 5	Reads USRT/UART Status. This instruction reads the contents of the USRT or UART internal status conditions.
OUTPUT 6	Loads Asynchronous Baud Rate Least Significant Bit register (LSB). The Asynchronous Baud Rate LSB register contains the eight least significant bits of the 12-bit word that specifies the asynchronous baud rate.
INPUT 6	Not implemented.
OUTPUT 7	Loads Asynchronous Baud Rate Most Significant Bit (MSB) register. The Asynchronous Baud Rate MSB register contains the four most significant bits of the 12-bit word that specifies the async baud rate.
INPUT 7	Not implemented

Figure 4-7. General DCMP Input/Output Register Descriptions (Sheet 2 of 2)

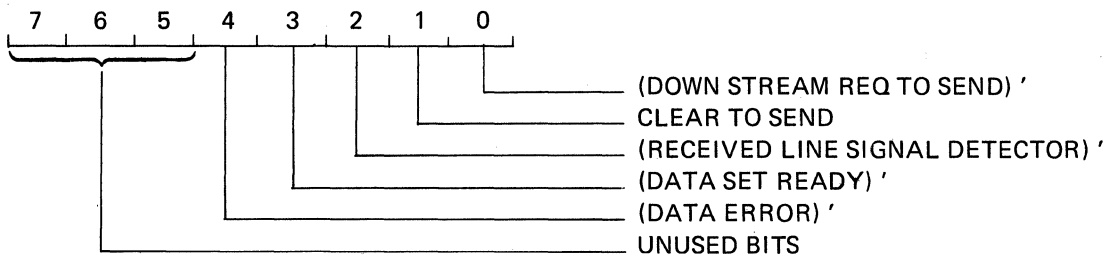
DATA COMM STATUS REG. (OUTPUT 3)



Bit	Name	Function
0	DATA TERMINAL READY	Controls the state of RS 232 circuit CD.
1	REQUEST TO SEND	Controls the state of RS 232 circuit CA.
2	TRANSMIT FREQUENCY SELECT	Controls the state of RS 232 circuit CH.
3	SELECT STANDBY	Controls the state of CCITT circuit 116.
4	BLOCK DOWN STREAM	When set blocks the "downstream" CA circuit and CB signals. It prevents the downstream CA from being passed "upstream" and prevents CB from being passed "downstream."
5	DIRECT CONNECTION CONFIGURATION	Should be set whenever the terminal is part of a "direct connection" data comm network (BDI or TDI).
6	TEST MODE	When set causes all data sent to the USRT or UART to be "looped-back" and treated as if it were data being received from the data communication cables.
7	MONITOR MODE	When set, it configures the terminal such that all data passing to and from the downstream terminals will be treated as receive data.

Figure 4-8. Detailed SIO Input/Output Register Descriptions (Sheet 1 of 6)

RS 232 LINE STATUS (INPUT 3)



Bit	Name	Function
0	(DOWNSTREAM REQ TO SEND) '	If the downstream CA circuit is true and BLOCK DOWNSTREAM (bit 4 of the DATA COM STATUS REG) is false this bit will read as false (0). Under all other conditions it will read true(1).
1	CLEAR TO SEND	Reads the state of RS 232 circuit CB.
2	(RECEIVED LINE SIG. DETECTOR) '	Reads the complement of the RS 232 circuit CF.
3	(DATA SET READY) '	Reads the complement of the RS 232 circuit CC.
4	(DATA ERROR) '	Reads the complement of the state of the receive data error detection logic. If the signal is false, it indicates that a character has been received by the USRT or UART and that character has an error associated with it.

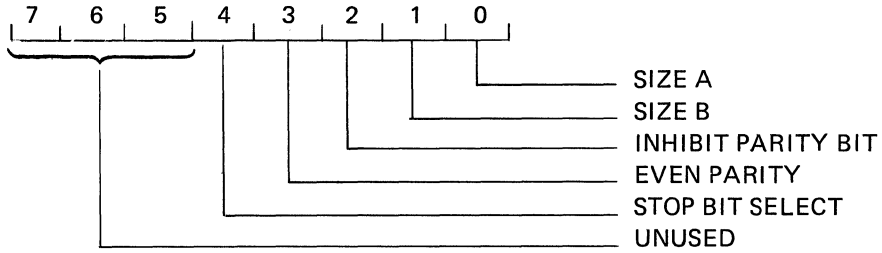
Figure 4-8. Detailed SIO Input/Output Register Descriptions (Sheet 2 of 6)

DATA COM WRITE BUFFER (OUTPUT 4)

NOTE: This register is used to transfer data and status to the USRT and UART. The destination of the bits loaded is determined by the contents of the Data Communications Write Select register.

DATA: If the buffer is loaded with data to be transmitted the bits 7 through 0 are the actual character to be sent.

STATUS: If the buffer is loaded with status the table below describes the interpretation of the bits.



Bit	Name	Function
0	SIZE A	These bits select the number of character bits to be transmitted or received:
1	SIZE B	SIZE B / SIZE A / No. of Bits: 0 / 0 / 5 bits; 0 / 1 / 7 bits; 1 / 0 / 6 bits; 1 / 1 / 8 bits
2	INHIBIT PARITY BIT	When set the UART or USRT will not include a parity bit in the transmitted word, and will not expect a parity bit in received data.
3	EVEN PARITY	If bit 2 is low, then if this bit is set even parity will be generated and expected, else odd parity.
4	STOP BIT SELECT	(Note: This is for async only). If it is set, selects 1.5 stop bits for five character format, 2 stop bits for all other lengths. If it is clear, selects one stop bit for all lengths.

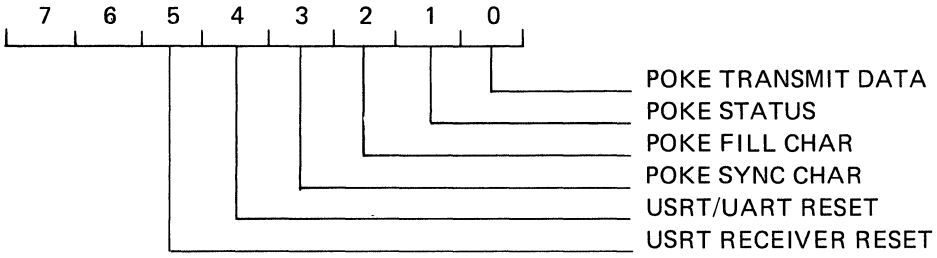
Figure 4-8. Detailed SIO Input/Output Register Descriptions (Sheet 3 of 6)

DATA COM DATA (INPUT 4)

The INPUT 4 instruction is used to read the data that has been received by the USRT or UART.

DATA COM WRITE SELECT REG. (OUTPUT 5)

NOTE: This register is used to control the loading and initialization of the USRT and UART. It is used to specify the destination of data loaded into the DATA COM WRITE BUFFER.

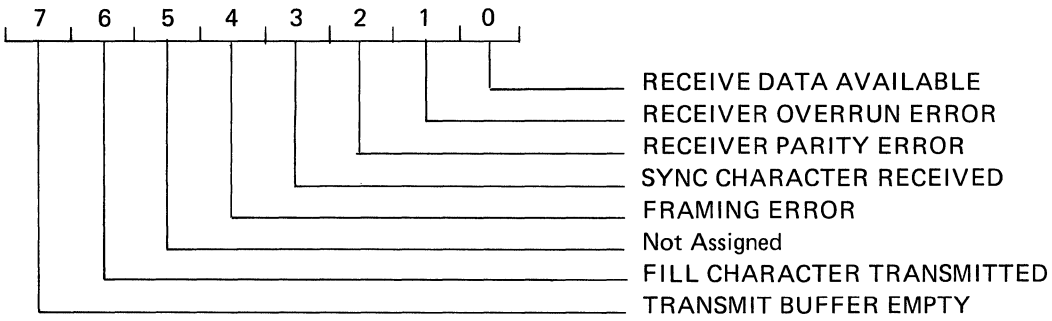


Bit	Name	Function
0	POKE TRANSMIT DATA	If this bit is set when an OUT 4 is executed, the OUT 4 data will be loaded into the USRT Transmitter Holding Register, or UART Transmitter Buffer register, and then transmitted.
1	POKE STATUS	If this bit is set when an OUT 4 is executed, the OUT 4 data will be loaded into the USRT/UART status registers.
2	POKE FILL CHAR	If this bit is set when an OUT 4 is executed, the OUT 4 data will be loaded into the USRT fill character register. (Note: This bit has no effect if async mode, i.e., UART)
3	POKE SYNC CHAR	Same as bit 2, except loads SYNC CHAR into sync char register of USRT.
4	USRT/UART RESET	When set this bit causes a reset of the USRT or UART.
5	USRT RECEIVER RESET	When set this bit causes a reset of the USRT receiver.

Figure 4-8. Detailed SIO Input/Output Register Descriptions (Sheet 4 of 6)

USRT/UART STATUS (INPUT 5)

NOTE: The bits read by the INPUT 5 are not the same as those loaded by an OUTPUT 4 with bit 1 of DATA COMM WRITE SELECT REG set.



Bit	Name	Function
0	RECEIVE DATA AVAILABLE	If this bit is set there is a received character available in the USRT or UART. (The character can be read with an INPUT 4 instruction).
1	RECEIVER OVERRUN ERROR	If this bit is set it indicates that additional characters have been received by the USRT/UART before the previous receive character has been read with an INPUT 4.
2	RECEIVER PARITY ERROR	A true indicates that the current receive character has a parity error.
3	SYNC CHARACTER RECEIVED	A true indicates that the current receive character is a SYNC character (USRT only).
4	FRAMING ERROR	A true indicates that the current receive character has a stop-bit error. (UART only).
5	Not Assigned	
6	FILL CHARACTER TRANSMITTED	A true indicates a fill character was transmitted (USRT only).
7	TRANSMIT BUFFER EMPTY	A true indicates the URST/UART is ready to accept a data character to be transmitted.

Figure 4-8. Detailed SIO Input/Output Register Descriptions (Sheet 5 of 6)

ASYNC BAUD RATE LSB REG. (OUTPUT 6)

The ASYNC BAUD RATE LSB REG. contains the eight least significant bits of the 12-bit word that specifies the async baud rate.

ASYNC BAUD RATE MSB REG. (OUTPUT 7)

The ASYNC BAUD RATE MSB REG. contains the four most significant bits of the four-bit word that specifies the async baud rate. (1802 bits 0.3).

Figure 4-8. Detailed SIO Input/Output Register Descriptions (Sheet 6 of 6)

RS232 Control Circuits

There are several RS232 circuits that must be programmed for the particular type transfer that is to take place on the data comm lines. These circuits are controlled using INPUT 1 and OUTPUT 1 instructions.

UART and USRT Logic

The actual serial data communication logic on the SIO board consists of either a UART or USRT chip; the type chip depends on whether the board is equipped for asynchronous (UART) or synchronous (USRT) data communications.

Each of these chips has certain global status information, such as number of bits per character, and type of parity, that must be loaded into them. There are also status bits that must be sampled on a character by character basis, such as Receive Data Available. The chips also require some initialization in the form of Reset and Restart bits. Finally, transmit data must be loaded in, and received data read from the chips.

All information, whether global status or transmit data, loaded into the UART or USRT is loaded via the Data Communications Write Buffer, which is loaded with an OUTPUT 4 instruction. The destination of the information loaded with an OUTPUT 4 is specified by the contents of the Data Communications Write Select register (loaded by an OUTPUT 5). The Data Communications Write Select register also contains bits that are used to "master reset" the UART or USRT, and also to restart the USRT circuit that searches for a sync character.

An INPUT 5 instruction is used to read the status of the UART or USRT.

An INPUT 4 instruction is used to read the data that has been received by the UART or USRT.

Logic also exists that continually monitors the status of the UART and USRT receive circuit status to determine when a character has been received, and to determine whether there is an error associated with the character. This logic produces a flag that can be tested as EF3 of the DCMP. If the flag is true, there is error-free data available for reading. A second signal is produced which can be sampled as bit 4 of the data read with an INPUT 3 instruction. If this bit is false, it means an error has been detected in the receive data.

There is logic on the SIO board that produces the data clock for asynchronous Data Communications. The circuitry includes a 12-bit register that must be loaded with the necessary value to produce a desired data rate. This 12-bit register is loaded with two output instructions:

a. OUTPUT 6 loads the eight least significant bits of the register.

b. OUTPUT 7 loads the four most significant bits of the register.

The clock generation circuit divides the 3.06 MHz CLOCK down to 16 times the desired asynchronous data rate. (For example, if 38.4 K baud is desired, the frequency generated by the circuit must be 16 times 38.4 kHz, or 614.4 kHz.)

The clock circuit contains a 12-bit register that is used to specify the desired rate. This register is loaded using OUTPUT instructions of the SIO. The value loaded is calculated using the formula:

$$\text{"12-bit value"} = 4096 - \frac{306818.5}{16 \times (\text{data rate in Hz})}$$

(The quotient should be rounded to the nearest integer.)

Example:

data rate desired = 38.4 K baud

$$\text{12-bit value} = 4096 - \frac{306818.5}{16 \times 38400}$$

$$= 4096 - 4.99$$

$$= 4096 - 5$$

12-bit value = 4091

INITIALIZATION OF UART AND USRT

Both the UART and USRT require initializing action. They both require a "master reset." This is accomplished by executing two OUTPUT 5 instructions, first with bit 4 of the data set, and then with bit 4 of the data clear.

Both the UART and USRT require that status words be loaded which specify things such as character size, parity configuration. This is accomplished by first doing an OUTPUT 5 instruction with only bit 1 set, then doing an OUTPUT 4 instruction with the bits configured as necessary for the desired configuration.

The USRT requires some additional initialization. A "fill character" must be loaded in it. This is done by doing an OUTPUT 5 instruction with only bit 2 set, then doing an OUTPUT 4 instruction with the desired data for the fill character. Before the USRT can be put in receive mode, it must know what character it should look for as a sync character. A sync character must therefore be loaded in the USRT. This is done by first doing an OUTPUT 5 instruction with only bit 3 set, then doing an OUTPUT 4 instruction with the desired data for the sync character. Once the sync character has been loaded, the receive circuitry of the USRT can be activated by doing a receiver restart; two OUTPUT 5 instructions are executed, first with bit 5 set, then with bit 5 clear.

TRANSMIT MODE PROGRAMMING

Once the USRT or UART has been initialized, all that is necessary to transmit characters is to set up the RS232 control circuits for transmit mode, and then load characters to the UART or USRT. An INPUT 5 must be done to determine if the UART or USRT is ready to accept a new character; bit 7 being set indicates it is ready. To load a character, bit 0 of the Data Communications Write Select register must be set. For each character to be loaded, an OUTPUT 4 must be done. (Note it is necessary to set bit 0 of the Data Communications Write Select register only once, not for each character.)

Testing bit 7 of the data received by an INPUT 5 instruction only indicates that the USRT or UART is ready to accept a new character; it is not an indication that the last character has been completely transmitted. To determine this, it is necessary to test flag EF4 of the DCMP. The interpretation of this flag depends on whether a UART or USRT is being used. If it is a UART, it means that the last character has been transmitted. If it is a USRT, it means that a fill character has been transmitted, but this implies that the last character has been transmitted since a fill character is transmitted only when no new character is specified in time.

RECEIVE MODE PROGRAMMING

Once the UART or USRT has been initialized, all that is necessary to receive characters is to set up the RS232 control circuits for receive mode and to test flag EF3. When EF3 goes true, the data can be read with an INPUT 4 instruction, then EF3 can be tested again. Note that this method will result in a loop waiting for EF3 if a receive error ever occurs. The loop should therefore include a test of the (DATA ERROR) ' bit of the RS232 Line Status (with INPUT 3 instruction). This test need only be made if EF3 is false.

TEST MODE PROGRAMMING

Hardware facilities exist on the SIO board to allow testing of the UART and USRT. This mode is known as test mode, and is entered by setting bit 6 of the Data Communications Status register (OUTPUT 3 instruction). In this mode, all data transmitted by the UART or USRT is "looped-back" to the receive section of the UART or USRT and treated as received data. The UART and USRT should be programmed in the normal fashion while in this mode. No data will be sent to the data communications cable driving circuits while in this mode. (Note that with a USRT a modem must be connected to the terminal for this mode to work since the circuitry relies on the data comm supplied clock to drive the USRT.)

MONITOR MODE PROGRAMMING

To configure the terminal to monitor mode bit 7 of the DATA COMM STATUS register must be set (with an OUTPUT 3 instruction).

SECTION 5
SWITCHING POWER SUPPLY

GENERAL DESCRIPTION

The power supply is the MTS-1 and is called a switching power supply because the output voltage amplitude of a series of rectifier and filter assemblies is controlled by varying the length of time the input voltage to the rectifier and filter assemblies is applied. The removal and application of the voltage is controlled by causing a transistor to conduct heavily for a period of time and then causing it to cease conducting for a different period of time. This transition of the transistor from conduction to non-conduction, and back to conduction, is referred to as "switching." The outputs of the rectifier and filter assemblies are roughly shown by the following equation:

$$V_o = Vin \frac{Ton}{Ton + Toff}$$

where:

V_o = The output voltage

Vin = Input voltage amplitude to the rectifier and filter assembly

Ton = Time that Vin is present

Toff = Time that Vin is removed

There are other factors involved, but it can be seen by the above equation that the output is determined mainly from the ratio of the time the input voltage is applied to the sum of the time the voltage is removed and applied times the input voltage.

The switching power supply has several advantages over a non-switching or linear power supply. The switching type is lighter in weight, smaller in size, and more efficient than linear power supplies. The block diagram in figure 5-1 illustrates the power supply used in the MTS-1.

The power supply normally operates in a switching frequency range of 20 to 25 Kilohertz. However, when the power supply output power is below 15 Watts, the power supply will go into a discontinuous current mode and the duty cycle and operating frequency will decrease quite rapidly as the output power is decreased further. With no load applied to the power supply, it will operate in the audio frequency range.

FULL-WAVE BRIDGE

The ac input voltage is applied to a full-wave bridge circuit to develop the primary dc voltage to be used in the supply. In series with this dc voltage are a fuse and a 50-ohm thermistor. The 16 millisecond holdover is provided by the filter capacitor for this full-wave bridge circuit.

LINEAR REGULATOR

The Linear Regulator provides the power to start and run the Pulse Width Modulator (PWM) oscillator circuit. The PWM oscillator circuit is shown as the modulator block of figure 5-1. This circuit provides the controlling waveform to turn the Drive and Switch circuit on and off. This waveform is controlled by current sensing and output voltage amplitude sensing.

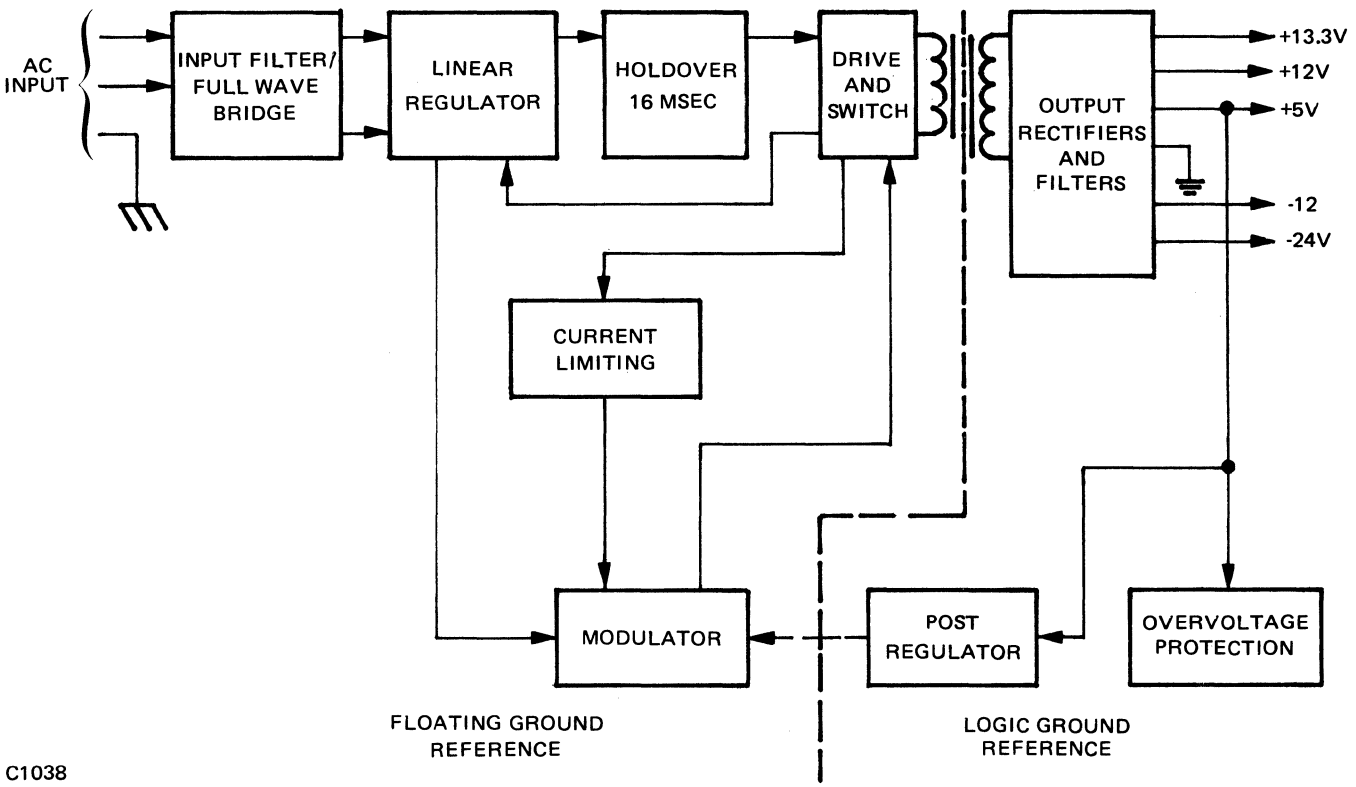


Figure 5-1. Switching Power Supply, Simplified Block Diagram

POWER SWITCH

The Power Switch applies the input voltage to the output transformer feeding the Output Rectifiers and Filters. The modulator waveform controls the length of time the Power Switch is on or off.

CURRENT LIMIT

The Current-Limiting circuit senses when the current through the Drive and Switch exceeds a fixed reference. When this reference is exceeded, a change signal is developed and applied to the modulator circuit. This change signal causes a reduction in the length of time the Drive and Switch is allowed to conduct.

POST REGULATOR

The Post Regulator circuit monitors the +5 Volt output. When it varies, an output signal is developed to cause the modulator to increase or decrease the length of time the Drive and Switch is on to correct the output voltage. The Post Regulator contains an optocoupler to couple the error amplitude from the logic ground reference portion of the Power Supply to the floating ground reference section of the power supply. The error amplitude causes the light-emitting diode in the optocoupler to increase or decrease the intensity of the light it releases. This light intensity change is detected and transformed back into a varying amplitude signal and applied to the modulator for subsequent control of the Drive and Switch.

OVERVOLTAGE

A circuit is provided for overvoltage detection on the +5 Volt supply output voltage. When the +5 Volt supply goes to approximately +5.1 Volts, a Silicon Control Rectifier (SCR) is triggered. The SCR tries to short, or "crowbar", +5 Volts and +13 Volts to ground.

VOLTAGE REFERENCE GROUNDS, VOLTAGE ISOLATION AND THE USE OF ELECTRICAL TEST EQUIPMENT

The Switching Power Supply has three "grounds" as shown in figure 5-2. The primary input power to the MTS-1 is provided by means of three wires, two of which provide the ac input voltage, and the third wire (green) is connected to the "Earth" ground at the building power distribution box. The green wire is attached to the MTS-1 chassis. After the ac voltage enters the power supply, it is rectified by a diode bridge and charges an electrolytic capacitor. The cathode of this capacitor defines the internal primary circuit ground.

CAUTION

The potential of the circuit ground is different from the green wire ground potential. Therefore, when using an oscilloscope or other measuring equipment to measure voltage potentials, electrical isolation must be provided between the power supply and the equipment, through use of measuring equipment capable of making a floating measurement, or by applying power to the supply through an isolation transformer.

WARNING

The primary ground is floating at line voltage; a shock hazard exists.

The third ground is defined by the secondary winding of the output transformer. This ground is electrically isolated from the other two grounds. This ground is referred to as "Logic" ground and is not normally connected to "Earth", or "Chassis," ground.

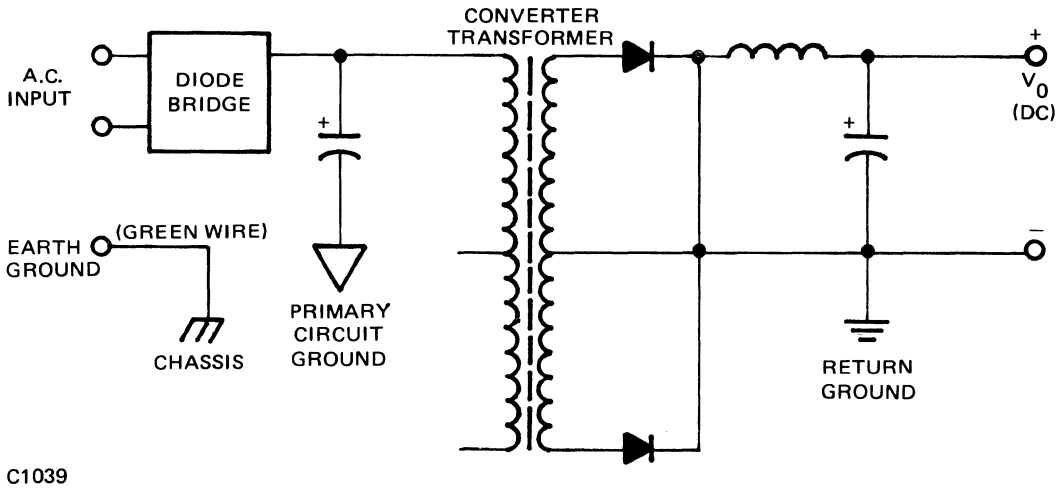


Figure 5-2. Grounding Illustration