Description

The μPD7220 Graphics Display Controller (GDC) is an intelligent microprocessor peripheral designed to be the heart of a high-performance raster-scan computer graphics and character display system. Positioned between the video display memory and the microprocessor bus, the GDC performs the tasks needed to generate the raster display and manage the display memory. Processor software overhead is minimized by the GDC's sophisticated instruction set, graphics figure drawing, and DMA transfer capabilities. The display memory supported by the GDC can be configured in any number of formats and sizes up to 256K 16-bit words. The display can be zoomed and panned, while partitioned screen areas can be independently scrolled. With its light pen input and multiple controller capability, the GDC is ideal for advanced computer graphics applications.

For a more detailed description of the GDC's operation, please refer to the GDC Design Manual.

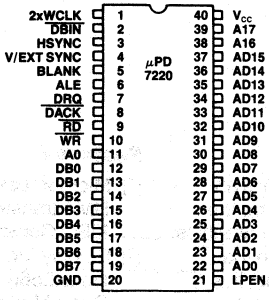
System Considerations

The GDC is designed to work with a general purpose microprocessor to implement a high-performance computer graphics system. Through the division of labor established by the GDC's design, each of the system components is used to the maximum extent through six-level hierarchy of simultaneous tasks. At the lowest level, the GDC generates the basic video raster timing, including sync and blanking signals. Partitioned areas on the screen and zooming are also accomplished at this level. At the next level, video display memory is modified during the figure drawing operations and data moves. Third, display memory addresses are calculated pixel by pixel as drawing progresses. Outside the GDC at the next level, preliminary calculations are done to prepare drawing parameters. At the fifth level, the picture must be represented as a list of graphics figures drawable by the GDC. Finally, this representation must be manipulated, stored, and communicated. By handling the first three levels, the GDC takes care of the high-speed and repetitive tasks required to implement a graphics system.

Features

- ☐ Microprocessor Interface
  - DMA transfers with 8257- or 8237-type controllers
  - FIFO Command Buffering
- ☐ Display Memory Interface
  - Up to 256K words of 16 bits
  - Read-Modify-Write (RMW) Display Memory cycles in under 800ns
  - Dynamic RAM refresh cycles for nonaccessed memory
- ☐ Light Pen Input
- ☐ External video synchronization mode
- ☐ Graphics Mode
  - Four megabit, bit-mapped display memory
- ☐ Character Mode
  - 8K character code and attributes display memory
- ☐ Mixed Graphics and Character Mode
  - 64K if all characters
  - 1 megapixel if all graphics
- ☐ Graphics Capabilities
  - Figure drawing of lines, arc/circles, rectangles, and graphics characters in 800ns per pixel
  - Display 1024-by-1024 pixels with 4 planes of color or grayscale
  - Two independently scrollable areas
- ☐ Character Capabilities
  - Auto cursor advance
  - Four independently scrollable areas
  - Programmable cursor height
  - Characters per row: up to 256
  - Character rows per screen: up to 100
- ☐ Video Display Format
  - Zoom magnification factors of 1 to 16
  - Panning
  - Command-settable video raster parameters
- ☐ Technology
  - Single +5 volt, NMOS, 40-pin DIP
- ☐ DMA Capability
  - Bytes or word transfers
  - 4 clock periods per byte transferred

Pin Configuration



Pin Identification

Pin No.	Symbol	Direction	Function
1	2xWCLK	IN	Clock Input
2	$\overline{DBIN}$	OUT	Display Memory Read Input Flag
3	HSYNC	OUT	Horizontal Video Sync Output
4	V/EXT SYNC	IN/OUT	Vertical Video Sync Output or External VSYNC Input
5	BLANK	OUT	CRT Blanking Output
6	ALE ($\overline{RAS}$)	OUT	Address Latch Enable Output
7	DRQ	OUT	DMA Request Output
8	$\overline{DACK}$	IN	DMA Acknowledge Input
9	$\overline{RD}$	IN	Read Strobe Input for Microprocessor Interface
10	WR	IN	Write Strobe Input for Microprocessor Interface
11	A0	IN	Address Select Input for Microprocessor Interface
12-19	DB0 to 7	IN/OUT	Bidirectional Data Bus to Host Microprocessor
20	GND	—	Ground
21	LPEN	IN	Light Pen Detect Input
22-34	AD0 to 12	IN/OUT	Address and Data Lines to Display Memory
35-37	AD13 to 15	IN/OUT	Utilization Varies with Mode of Operation
38	A16	OUT	Utilization Varies with Mode of Operation
39	A17	OUT	Utilization Varies with Mode of Operation
40	V_{CC}	—	+ 5V ± 10%

Character Mode Pin Utilization

Pin No.	Name	Direction	Function
35-37	AD13 to 15	OUT	Line Counter Bits 0 to 2 Outputs
38	A16	OUT	Line Counter Bit 3 Output
39	A17	OUT	Cursor Output and Line Counter Bit 4*

Mixed Mode Pin Utilization

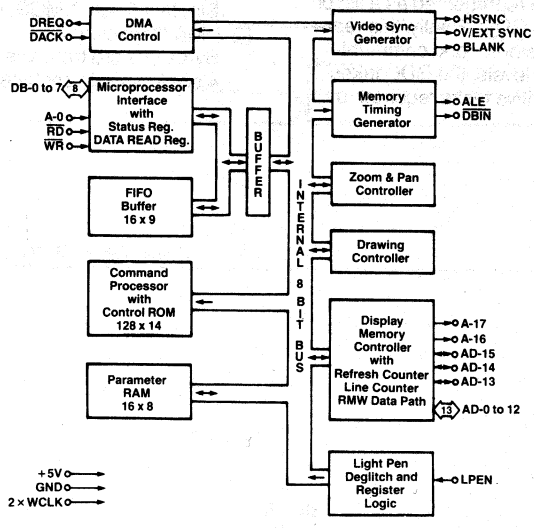
Pin No.	Name	Direction	Function
35-37	AD13 to 15	IN/OUT	Address and Data Bits 13 to 15
38	A16	OUT	Attribute Blink and Clear Line Counter* Output
39	A17	OUT	Cursor and Bit-Map Area* Flag Output

*Output 10 clock cycles after trailing edge of HSYNC. See figure for timing example.

Graphics Mode Pin Utilization

Pin No.	Name	Direction	Function
35-37	AD13 to 15	IN/OUT	Address and Data Bits 13 to 15
38	A16	OUT	Address Bit 16 Output
39	A17	OUT	Address Bit 17 Output

Block Diagram



GDC Components

Microprocessor Bus Interface

Control of the GDC by the system microprocessor is achieved through an 8-bit bidirectional interface. The status register is readable at any time. Access to the FIFO buffer is coordinated through flags in the status register and operates independently of the various internal GDC operations, due to the separate data bus connecting the interface and the FIFO buffer.

Command Processor

The contents of the FIFO are interpreted by the command processor. The command bytes are decoded, and the succeeding parameters are distributed to their proper destinations within the GDC. The command processor yields to the bus interface when both access the FIFO simultaneously.

DMA Control

The DMA control circuitry in the GDC coordinates transfers over the microprocessor interface when using an external DMA controller. The DMA Request and Acknowledge handshake lines directly interface with a μPD8257 or μPD8237 DMA controller, so that display data can be moved between the microprocessor memory and the display memory.

Parameter RAM

The 16-byte RAM stores parameters that are used repetitively during the display and drawing processes. In character mode, this RAM holds four sets of partitioned display area parameters; in graphics mode, the drawing pattern and graphics character take the place of two of the sets of parameters.

Video Sync Generator

Based on the clock input, the sync logic generates the raster timing signals for almost any interlaced, non-interlaced, or "repeat field" interlaced video format. The generator is programmed during the idle period following a reset. In video sync slave mode, it coordinates timing between multiple GDCs.

Memory Timing Generator

The memory timing circuitry provides two memory cycle types: a two-clock period refresh cycle and the read-modify-write (RMW) cycle which takes four clock periods. The memory control signals needed to drive the display memory devices are easily generated from the GDC's ALE and DBIN outputs.

Zoom & Pan Controller

Based on the programmable zoom display factor and the display area entries in the parameter RAM, the zoom and pan controller determines when to advance to the next memory address for display refresh and when to go on to the next display area. A horizontal zoom is produced by slowing down the display refresh rate while maintaining the video sync rates. Vertical zoom is accomplished by repeatedly accessing each line a number of times equal to the horizontal repeat. Once the line count for a display area is exhausted, the controller accesses the starting address and line count of the next display area from the parameter RAM. The system microprocessor, by modifying a display area starting address, can pan in any direction, independently of the other display areas.

Drawing Controller

The drawing processor contains the logic necessary to calculate the addresses and positions of the pixels of the various graphics figures. Given a starting point and the appropriate drawing parameters, the drawing controller needs no further assistance to complete the figure drawing.

Display Memory Controller

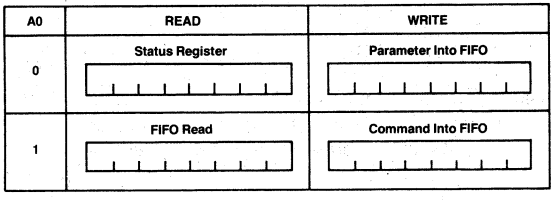
The display memory controller's tasks are numerous. Its primary purpose is to multiplex the address and data information in and out of the display memory. It also contains the 16-bit logic unit used to modify the display memory contents during RMW cycles, the character mode line counter, and the refresh counter for dynamic RAMs. The memory controller apportions the video field time between the various types of cycles.

Light Pen Deglitcher

Only if two rising edges on the light pen input occur at the same point during successive video fields are the pulses accepted as a valid light pen detection. A status bit indicates to the system microprocessor that the light pen register contains a valid address.

Programmer's View of GDC

The GDC occupies two addresses on the system microprocessor bus through which the GDC's status register and FIFO are accessed. Commands and parameters are written into the GDC's FIFO and are differentiated based on address bit A0. The status register or the FIFO can be read as selected by the address line.



GDC Microprocessor Bus Interface Registers

Commands to the GDC take the form of a command byte followed by a series of parameter bytes as needed for specifying the details of the command. The command processor decodes the commands, unpacks the parameters, loads them into the appropriate registers within the GDC, and initiates the required operations.

The commands available in the GDC can be organized into five categories as described in the following section.

GDC Command Summary

Video Control Commands

1. RESET — Resets the GDC to its idle state.
2. SYNC — Specifies the video display format.
3. VSYNC — Selects master or slave video synchronization mode.
4. CCHAR — Specifies the cursor and character row heights.

Display Control Commands

1. START — Ends Idle mode and unblanks the display.
2. BCTRL — Controls the blanking and unblanking of the display.
3. ZOOM — Specifies zoom factors for the display and graphics characters writing.
4. CURS — Sets the position of the cursor in display memory.
5. PRAM — Defines starting addresses and lengths of the display areas and specifies the eight bytes for the graphics character.
6. PITCH — Specifies the width of the X dimension of display memory.

Drawing Control Commands

1. WDAT — Writes data words or bytes into display memory.
2. MASK — Sets the mask register contents.
3. FIGS — Specifies the parameters for the drawing controller.
4. FIGD — Draws the figure as specified above.
5. GCHRD — Draws the graphics character into display memory.

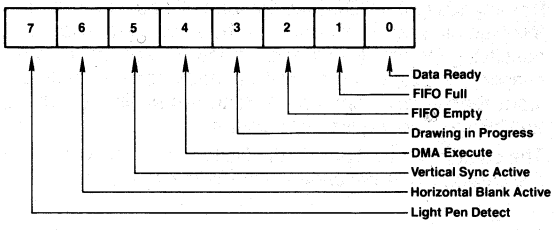
Data Read Commands

1. RDAT: — Reads data words or bytes from display memory.
2. CURD: — Reads the cursor position.
3. LPRD: — Reads the light pen address.

DMA Control Commands

1. DMAR — Requests a DMA read transfer.
2. DMAW — Requests a DMA write transfer.

Status Register Flags



Status Register (SR)

SR-7: Light Pen Detect

When this bit is set to 1, the light pen address (LAD) register contains a deglitched value that the system microprocessor may read. This flag is reset after the 3-byte LAD is moved into the FIFO in response to the light pen read command.

SR-6: Horizontal Blanking Active

A 1 value for this flag signifies that horizontal retrace blanking is currently underway.

SR-5: Vertical Sync

Vertical retrace sync occurs while this flag is a 1. The vertical sync flag coordinates display format modifying commands to the blanked interval surrounding vertical sync. This eliminates display disturbances.

SR-4: DMA Execute

This bit is a 1 during DMA data transfers.

SR-3: Drawing in Progress

While the GDC is drawing a graphics figure, this status bit is a 1.

SR-2: FIFO Empty

This bit and the FIFO Full flag coordinate system microprocessor accesses with the GDC FIFO. When it is 1, the Empty flag ensures that all the commands and parameters previously sent to the GDC have been interpreted.

SR-1: FIFO Full

A 1 at this flag indicates a full FIFO in the GDC. A 0 ensures that there is room for at least one byte. This flag needs to be checked before each write into the GDC.

SR-0: Data Ready

When this flag is a 1, it indicates that a byte is available to be read by the system microprocessor. This bit must be tested before each read operation. It drops to a 0 while the data is transferred from the FIFO into the microprocessor interface data register.

FIFO Operation & Command Protocol

The first-in, first-out buffer (FIFO) in the GDC handles the command dialogue with the system microprocessor. This flow of information uses a half-duplex technique, in which the single 16-location FIFO is used for both directions of data movement, one direction at a time. The FIFO's direction is controlled by the system microprocessor through the GDC's command set. The host microprocessor coordinates these transfers by checking the appropriate status register bits.

The command protocol used by the GDC requires differentiation of the first byte of a command sequence from the succeeding bytes. The first byte contains the operation code and the remaining bytes carry parameters. Writing into the GDC causes the FIFO to store a flag value alongside the data byte to signify whether the byte was written into the command or the parameter address. The command processor in the GDC tests this bit as it interprets the

entries in the FIFO.

The receipt of a command byte by the command processor marks the end of any previous operation. The number of parameter bytes supplied with a command is cut short by the receipt of the next command byte. A read operation from the GDC to the microprocessor can be terminated at any time by the next command.

The FIFO changes direction under the control of the system microprocessor. Commands written into the GDC always put the FIFO into write mode if it wasn't in it already. If it was in read mode, any read data in the FIFO at the time of the turnaround is lost. Commands which require a GDC response, such as RDAT, CURD and LPRD, put the FIFO into read mode after the command is interpreted by the GDC's command processor. Any commands and parameters behind the read-evoking command are discarded when the FIFO direction is reversed.

Read-Modify-Write Cycle

Data transfers between the GDC and the display memory are accomplished using a read-modify-write (RMW) memory cycle. The four clock period timing of the RMW cycle is used to: 1) output the address, 2) read data from the memory, 3) modify the data, and 4) write the modified data back into the initially selected memory address. This type of memory cycle is used for all interactions with display memory including DMA transfers, except for the two clock period display and RAM refresh cycles.

The operations performed during the modify portion of the RMW cycle merit additional explanation. The circuitry in the GDC uses three main elements: the Pattern register, the Mask register, and the 16-bit Logic Unit. The Pattern register holds the data pattern to be moved into memory. It is loaded by the WDAT parameters or, during drawing, from the parameter RAM. The Mask register contents determine which bits of the read data will be modified. Based on the contents of these registers, the Logic Unit performs the selected operations of REPLACE, COMPLEMENT, SET, or CLEAR on the data read from display memory.

The Pattern register contents are ANDed with the Mask register contents to enable the actual modification of the memory read data, on a bit-by-bit basis. For graphics drawing, one bit at a time from the Pattern register is combined with the Mask. When ANDed with the bit set to a 1 in the Mask register, the proper single pixel is modified by the Logic Unit. For the next pixel in the figure, the next bit in the Pattern register is selected and the Mask register bit is moved to identify the pixel's location within the word. The Execution word address pointer register, EAD, is also adjusted as required to address the word containing the next pixel.

In character mode, all of the bits in the Pattern register are used in parallel to form the respective bits of the modify data word. Since the bits of the character code word are used in parallel, unlike the one-bit-at-a-time graphics drawing process, this facility allows any or all of the bits in a memory word to be modified in one RMW memory cycle. The Mask register must be loaded with 1s in the positions where modification is to be permitted.

The Mask register can be loaded in either of two ways. In graphics mode, the CURS command contains a four-bit dAD field to specify the dot address. The command processor converts this parameter into the one-of-16 format used in the Mask register for figure drawing. A full 16 bits can be loaded into the Mask register using the MASK command. In addition to the character mode use mentioned above, the 16-bit MASK load is convenient in graphics mode when all of the pixels of a word are to be set to the same value.

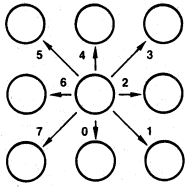
The Logic Unit combines the data read from display memory, the Pattern Register, and the Mask register to generate the data to be written back into display memory. Any one of four operations can be selected: REPLACE, COMPLEMENT, CLEAR or SET. In each case, if the respective Mask bit is 0, that particular bit of the read data is returned to memory unmodified. If the Mask bit is 1, the modification is enabled. With the REPLACE operation, the Pattern Register data simply takes the place of the read data for modification enabled bits. For the other three operations, a 0 in the modify data allows the read data bit to be returned to memory. A 1 value causes the specified operation to be performed in the bit positions with set Mask bits.

Figure Drawing

The GDC draws graphics figures at the rate of one pixel per read-modify-write (RMW) display memory cycle. These cycles take four clock periods to complete. At a clock frequency of 5MHz, this is equal to 800ns. During the RMW cycle the GDC simultaneously calculates the address and position of the next pixel to be drawn.

The graphics figure drawing process depends on the display memory addressing structure. Groups of 16 horizontally adjacent pixels form the 16-bit words which are handled by the GDC. Display memory is organized as a linearly addressed space of these words. Addressing of individual pixels is handled by the GDC's internal RMW logic.

During the drawing process, the GDC finds the next pixel of the figure which is one of the eight nearest neighbors of the last pixel drawn. The GDC assigns each of these eight directions a number from 0 to 7, starting with straight down and proceeding counterclockwise.



Drawing Directions

Figure drawing requires the proper manipulation of the address and the pixel bit position according to the drawing direction to determine the next pixel of the figure. To move to the word above or below the current one, it is necessary to subtract or add the number of words per line in display memory. This parameter is called the pitch. To move to the word to either side, the Execute word address cursor, EAD, must be incremented or decremented as the dot address pointer bit reaches the LSB or the MSB of the Mask register. To move to a pixel within the same word, it is necessary to rotate the dot address pointer register to the right or left.

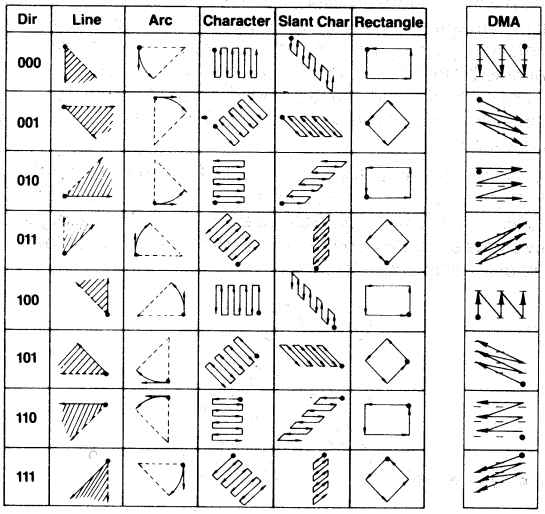
The table below summarizes these operations for each direction.

Dir	Operations to Address the Next Pixel
000	EAD + P → EAD
001	EAD + P → EAD dAD (MSB) = 1:EAD + 1 → EAD dAD → LR
010	dAD (MSB) = 1:EAD + 1 → EAD dAD → LR
011	EAD − P → EAD dAD (MSB) = 1:EAD + 1 → EAD dAD → LR
100	EAD − P → EAD
101	EAD − P → EAD dAD (LSB) = 1:EAD − 1 → EAD dAD → RR
110	dAD (LSB) = 1:EAD − 1 → EAD dAD → RR
111	EAD + P → EAD dAD (LSB) = 1:EAD − 1 → EAD dAD → RR

Where P = Pitch, LR = Left Rotate, RR = Right Rotate,
EAD = Execute Word Address, and
dAD = Dot Address stored in the Mask Register.

Whole word drawing is useful for filling areas in memory with a single value. By setting the Mask register to all 1s with the MASK command, both the LSB and MSB of the dAD will always be 1, so that the EAD value will be incremented or decremented for each cycle regardless of direction. One RMW cycle will be able to effect all 16 bits of the word for any drawing type. One bit in the Pattern register is used per RMW cycle to write all the bits of the word to the same value. The next Pattern bit is used for the word, etc.

For the various figures, the effect of the initial direction upon the resulting drawing is shown below:



Note that during line drawing, the angle of the line may be anywhere within the shaded octant defined by the DIR value. Arc drawing starts in the direction initially specified by the DIR value and veers into an arc as drawing proceeds. An arc may be up to 45 degrees in length. DMA transfers are done on word boundaries only, and follow the arrows indicated in the table to find successive word addresses. The slanted paths for DMA transfers indicate the GDC changing both the X and Y components of the word address when moving to the next word. It does not follow a 45 degree diagonal path by pixels.

Drawing Parameters

In preparation for graphics figure drawing, the GDC's Drawing Processor needs the figure type, direction and drawing parameters, the starting pixel address, and the pattern from the microprocessor. Once these are in place within the GDC, the Figure Draw command, FIGD, initiates the drawing operation. From that point on, the system microprocessor is not involved in the drawing process. The GDC Drawing Controller coordinates the RMW circuitry and address registers to draw the specifed figure pixel by pixel.

The algorithms used by the processor for figure drawing are designed to optimize its drawing speed. To this end, the specific details about the figure to be drawn are reduced by the microprocessor to a form conducive to high-speed address calculations within the GDC. In this way the repetitive, pixel-by-pixel calculations can be done quickly, thereby minimizing the overall figure drawing time. The table below summarizes the parameters.

Drawing Type	DC	D	D2	D1	DM
Initial Value*	0	8	8	−1	−1
Line	$\lvert\Delta I\rvert$	$2\lvert\Delta D\rvert - \lvert\Delta I\rvert$	$2(\lvert\Delta D\rvert - \lvert\Delta I\rvert)$	$2\lvert\Delta D\rvert$	–
Arc**	rsin ϕ	r−1	2(r−1)	−1	rsin $\theta\downarrow$
Rectangle	3	A−1	B−1	−1	A−1
Area Fill	B−1	A	A	–	–
Graphic Character***	B−1	A	A	–	–
Write Data	W−1	–	–	–	–
DMAW	D−1	C−1	–	–	–
DMAR	D−1	C−2	(C−2)/2†	–	–
Read Data	W	–	–	–	–

*Initial values for the various parameters remain as each drawing process ends.

**Circles are drawn with 8 arcs, each of which span 45°, so that $\sin\phi = 1/\sqrt{2}$ and $\sin\theta = 0$.

***Graphic characters are a special case of bit-map area filling in which B and A ≤ 8. If A = 8 there is no need to load D and D2.

Where:
−1 = all ONES value.

All numbers are shown in base 10 for convenience. The GDC accepts base 2 numbers (2s complement notation where appropriate).

– = No parameter bytes sent to GDC for this parameter.

ΔI = The larger at Δx or Δy.

ΔD = The smaller at Δx or Δy.

r = Radius of curvature, in pixels.

ϕ = Angle from major axis to end of the arc. $\phi \leq 45°$

θ = Angle from major axis to start of the arc. $\theta \leq 45°$

$\uparrow$ = Round up to the next higher integer.

$\downarrow$ = Round down to the next lower integer.

A = Number of pixels in the initially specified direction.

B = Number of pixels in the direction at right angles to the initially specified direction.

W = Number of words to be accessed.

C = Number of bytes to be transferred in the initially specified direction. (Two bytes per word if word transfer mode is selected.)

D = Number of words to be accessed in the direction at right angles to the initially specified direction.

DC = Drawing count parameter which is one less than the number of RMW cycles to be executed.

DM = Dots masked from drawing during arc drawing.

† = Needed only for word reads.

Graphics Character Drawing

Graphics characters can be drawn into display memory pixel-by-pixel. The up to 8-by-8 character display is loaded into the GDC's parameter RAM by the system microprocessor. Consequently, there are no limitations on the character set used. By varying the drawing parameters and drawing direction, numerous drawing options are available. In area fill applications, a character can be written into display memory as many times as desired without reloading the parameter RAM.

Once the parameter RAM has been loaded with up to eight graphics character bytes by the appropriate PRAM command, the GCHRD command can be used to draw the bytes into display memory starting at the cursor. The zoom magnification factor for writing, set by the zoom command, controls the size of the character written into the display memory in integer multiples of 1 through 16. The bit values in the PRAM are repeated horizontally and vertically the number of times specified by the zoom factor.

The movement of these PRAM bytes to the display memory is controlled by the parameters of the FIGS command. Based on the specified height and width of the area to be drawn, the parameter RAM is scanned to fill the required area.

For an 8-by-8 graphics character, the first pixel drawn uses the LSB of RA-15, the second pixel uses bit 1 of RA-15, and so on, until the MSB of RA-15 is reached.

The GDC jumps to the corresponding bit in RA-14 to continue the drawing. The progression then advances toward the LSB of RA-14. This snaking sequence is continued for the other 6 PRAM bytes. This progression matches the sequence of display memory addresses calculated by the drawing processor as shown above. If the area is narrower than 8 pixels wide, the snaking will advance to the next PRAM byte before the MSB is reached. If the area is less than 8 lines high, fewer bytes in the parameter RAM will be scanned. If the area is larger than 8 by 8, the GDC will repeat the contents of the parameter RAM in two dimensions, as required to fill the area with the 8-by-8 mozaic. (Fractions of the 8-by-8 pattern will be used to fill areas which are not multiples of 8 by 8.)

Parameter RAM Contents: RAM Address RA 0 to 15

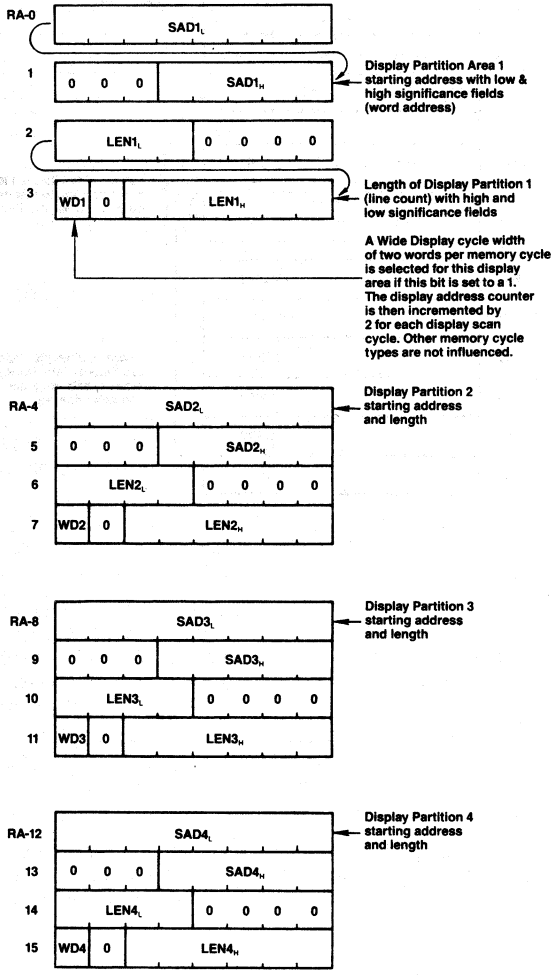
The parameters stored in the parameter RAM, PRAM, are available for the GDC to refer to repeatedly during figure drawing and raster-scanning. In each mode of operation the values in the PRAM are interpreted by the GDC in a predetermined fashion. The host microprocessor must load the appropriate parameters into the proper PRAM locations. PRAM loading command allows the host to write into any location of the PRAM and transfer as many bytes as desired. In this way any stored parameter byte or bytes may be changed without influencing the other bytes.

The PRAM stores two types of information. For specifying the details of the display area partitions, blocks of four bytes are used. The four parameters stored in each block include the starting address in display memory of each display area, and its length. In addition, there are two mode bits for each area which specify whether the area is a bit-mapped graphics area or a coded character area, and whether a 16-bit or a 32-bit wide display cycle is to be used for that area.

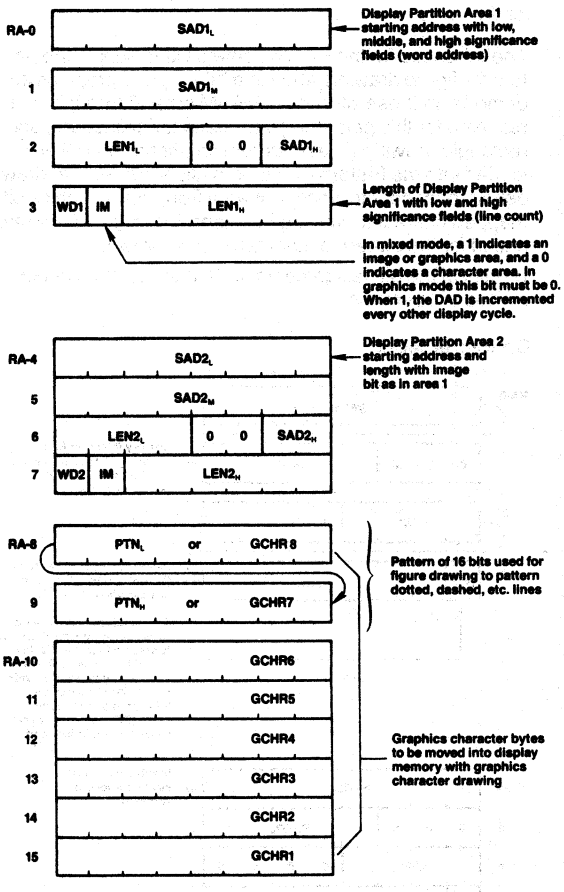
The other use for the PRAM contents is to supply the pattern for figure drawing when in a bit-mapped graphics area or mode. In these situations, PRAM bytes 8 through 16 are reserved for this patterning information. For line, arc, and rectangle drawing (linear figures) locations 8 and 9 are loaded into the Pattern Register to allow the GDC to draw dotted, dashed, etc. lines. For area filling and graphics bit-mapped character drawing locations 8 through 15 are referenced for the pattern or character to be drawn.

Details of the bit assignments are shown for the various modes of operation.

Character Mode



Graphics and Mixed Graphics and Character Modes

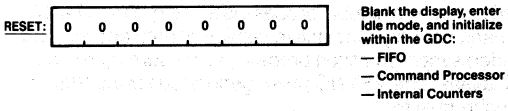


Command Bytes Summary

RESET	0	0	0	0	0	0	0	0
SYNC	0	0	0	0	1	1	1	DE
VSYNC	0	1	1	0	1	1	1	M
CCHAR	0	1	0	0	1	0	1	1
START	0	1	1	0	1	0	1	1
BCTRL	0	0	0	0	1	1	0	DE
ZOOM	0	1	0	0	0	1	1	0
CURS	0	1	0	0	1	0	0	1
PRAM	0	1	1	1	SA			
PITCH	0	1	0	0	0	1	1	1
WDAT	0	0	1	TYPE		0	MOD	
MASK	0	1	0	0	1	0	1	0
FIGS	0	1	0	0	1	1	0	0
FIGD	0	1	1	0	1	1	0	0
GCHRD	0	1	1	0	1	0	0	0
RDAT	1	0	1	TYPE		0	MOD	
CURD	1	1	1	0	0	0	0	0
LPRD	1	1	0	0	0	0	0	0
DMAR	1	0	1	TYPE		1	MOD	
DMAW	0	0	1	TYPE		1	MOD	

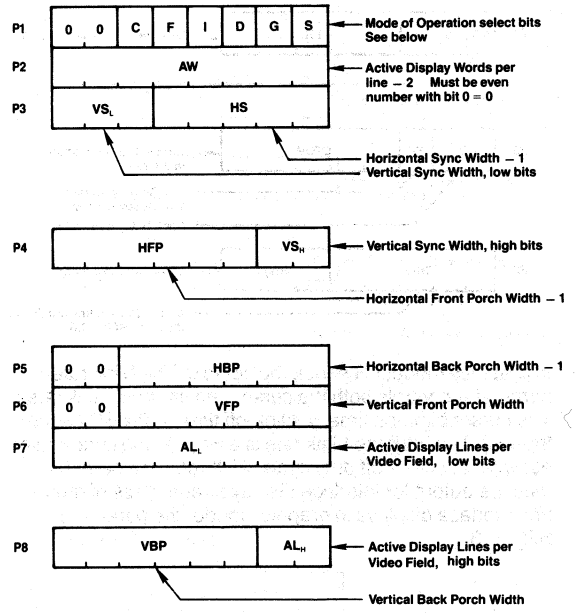
Video Control Commands

Reset



This command can be executed at any time and does not modify any of the parameters already loaded into the GDC.

If followed by parameter bytes, this command also sets the sync generator parameters as described below. Idle mode is exited with the START command.



In graphics mode, a word is a group of 16 pixels. In character mode, a word is one character code and its attributes, if any. The number of active words per line must be an even number from 2 to 256. An all-zero parameter value selects a count equal to 2^n where n = number of bits in the parameter field for vertical parameters. All horizontal widths are counted in display words. All vertical intervals are counted in lines.

Horizontal Back Porch Constraints

1. In general:
   HBP ≥ 3 Display Word Cycles (6 clock cycles).
2. If the IMAGE or WD modes change within one video field:
   HBP ≥ 5 Display Word Cycles (10 clock cycles).
3. If interlace or mixed mode is used:
   HBP ≥ 5 Display Word Cycles (10 clock cycles).

Horizontal Front Porch Constraints

1. If the display ZOOM function is used at other than 1X:
   HFP ≥ 2 Display Word Cycles (4 clock cycles).
2. If the GDC is used in the video sync Slave mode:
   HFP ≥ 4 Display Word Cycles (8 clock cycles).
3. If the Light Pen is used:
   HFP ≥ 6 Display Word Cycles (12 clock cycles).
4. If interlace mode is used:
   HFP ≥ 3 Display Word Cycles (6 clock cycles).

Horizontal SYNC Constraints

1. If Interlaced display mode is used:
   HS ≥ 5 Display Word Cycles (10 clock cycles).

Modes of Operation Bits

C	G	Display Mode
0	0	Mixed Graphics & Character
0	1	Graphics Mode
1	0	Character Mode
1	1	Invalid

I	S	Video Framing
0	0	Noninterlaced
0	1	Invalid
1	0	Interlaced Repeat Field for Character Displays
1	1	Interlaced

Repeat Field Framing: 2 Field Sequence with ½ line offset between otherwise identical fields.

Interlaced Framing: 2 Field Sequence with ½ line offset. Each field displays alternate lines.

Noninterlaced Framing: 1 field brings all of the information to the screen.

Total scanned lines in interlace mode is odd. The sum of VFP + VS + VBP + AL should equal one less than the desired odd number of lines.

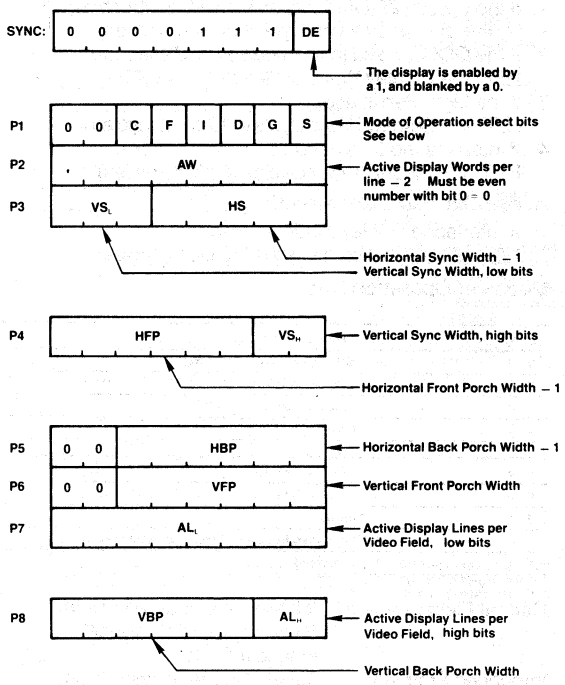
D	Dynamic RAM Refresh Cycles Enable
0	No Refresh — STATIC RAM
1	Refresh — Dynamic RAM

Dynamic RAM refresh is important when high display zoom factors or DMA are used in such a way that not all of the rows in the RAMs are regularly accessed during display raster generation and for otherwise inactive display memory.

F	Drawing Time Window
0	Drawing during active display time and retrace blanking
1	Drawing only during retrace blanking

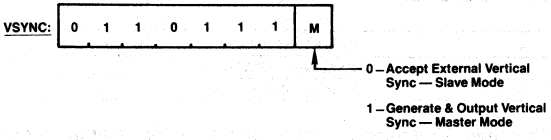
Access to display memory can be limited to retrace blanking intervals only, so that no disruptions of the image are seen on the screen.

SYNC Format Specify



This command also loads parameters into the sync generator. The various parameter fields and bits are identical to those at the RESET command. The GDC is not reset nor does it enter idle mode.

Vertical Sync Mode

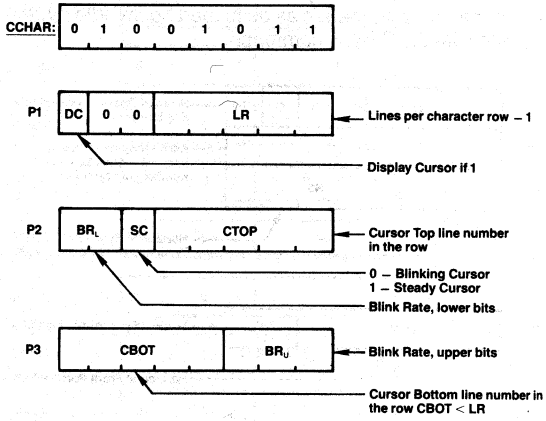


When using two or more GDCs to contribute to one image, one GDC is defined as the master sync generator, and the others operate as its slaves. The VSYNC pins of all GDCs are connected together.

A few considerations should be observed when synchronizing two or more GDCs to generate overlayed video via the VSYNC INPUT/OUTPUT pin. As mentioned above, the Horizontal Front Porch (HFP) must be 4 or more display cycles wide. This is equivalent to eight or more clock cycles. This gives the slave GDCs time to initialize their internal video sync generators to the proper point in the video field to match the incoming vertical sync pulse (VSYNC). This resetting of the generator occurs just after the end of the incoming VSYNC pulse, during the HFP interval. Enough time during HFP is required to allow the slave GDC to complete the operation before the start of the HSYNC interval.

Once the GDCs are initialized and set up as Master and Slaves, they must be given time to synchronize. It is a good idea to watch the VSYNC status bit of the Master GDC and wait until after one or more VSYNC pulses have been generated before the display process is started. The START command will begin the active display of data and will end the video synchronization process, so be sure there has been at least one VSYNC pulse generated for the Slaves to synchronize to.

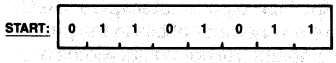
Cursor & Character Characteristics



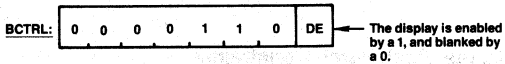
In graphics mode, LR should be set to 0. The blink rate parameter controls both the cursor and attribute blink rates. The cursor blink-on time = blink-off time = 2 x BR (video frames). The attribute blink rate is always ½ the cursor rate but with a ¾ on-¼ off duty cycle. All three parameter bytes must be output for interlace displays, regardless of mode. For interlace displays in graphics mode, the parameter $BR_L = 3$.

Display Control Commands

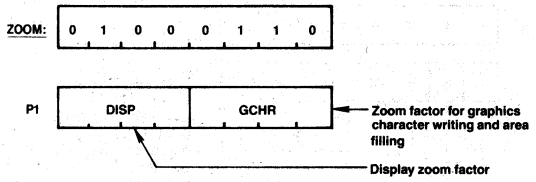
Start Display & End Idle Mode



Display Blanking Control

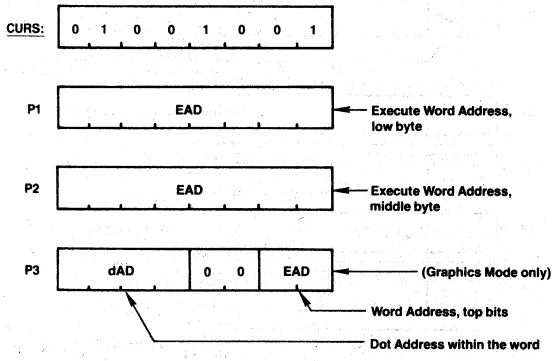


Zoom Factors Specify



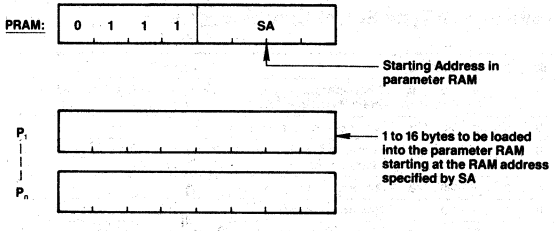
Zoom magnification factors of 1 through 16 are available using codes 0 through 15, respectively.

Cursor Position Specify



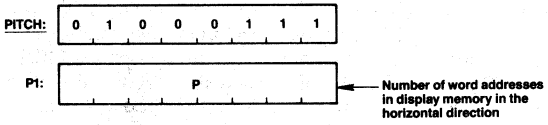
In character mode, the third parameter byte is not needed. The cursor is displayed for the word time in which the display scan address (DAD) equals the cursor address. In graphics mode, the cursor word address specifies the word containing the starting pixel of the drawing; the dot address value specifies the pixel within that word.

Parameter RAM Load



From the starting address, SA, any number of bytes may be loaded into the parameter RAM at incrementing addresses, up to location 15. The sequence of parameter bytes is terminated by the next command byte entered into the FIFO. The parameter RAM stores 16 bytes of information in predefined locations which differ for graphics and character modes. See the parameter RAM discussion for bit assignments.

Pitch Specification

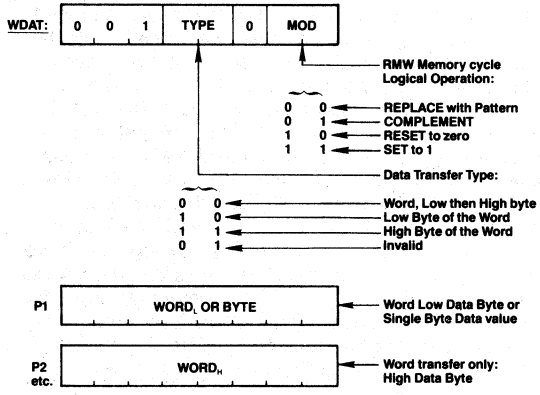


This value is used during drawing by the drawing processor to find the word directly above or below the current word, and during display to find the start of the next line.

The Pitch parameter (width of display memory) is set by two different commands. In addition to the PITCH command, the RESET (or SYNC) command also sets the pitch value. The "active words per line" parameter, which specifies the width of the raster-scan display, also sets the Pitch of the display memory. Note that the AW value is two less than the display window width. The PITCH command must be used to set the proper memory width larger than the window width.

Drawing Control Commands

Write Data into Display Memory



Upon receiving a set of parameters (two bytes for a word transfer, one for a byte transfer), one RMW cycle into Video Memory is done at the address pointed to by the cursor EAD. The EAD pointer is advanced to the next word, according to the previously specified direction. More parameters can then be accepted.

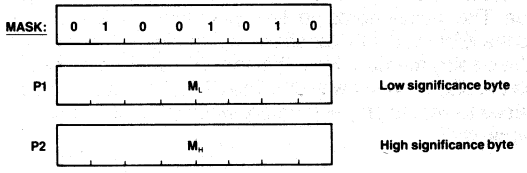
For byte writes, the unspecified byte is treated as all zeros during the RMW memory cycle.

In graphics bit-map situations, only the LSB of the WDAT parameter bytes is used as the pattern in the RMW operations. Therefore it is possible to have only an all ones or all zeros pattern. In coded character applications all the bits of the WDAT parameters are used to establish the drawing pattern.

The WDAT command operates differently from the other commands which initiate RMW cycle activity. It requires

parameters to set up the Pattern register while the other commands use the stored values in the parameter RAM. Like all of these commands, the WDAT command must be preceded by a FIGS command and its parameters. Only the first three parameters need be given following the FIGS opcode, to set up the type of drawing, the DIR direction, and the DC value. The DC parameter +1 will be the number of RMW cycles done by the GDC with the first set of WDAT parameters. Additional sets of WDAT parameters will see a DC value of 0 which will cause only one RMW cycle to be executed per set of parameters.

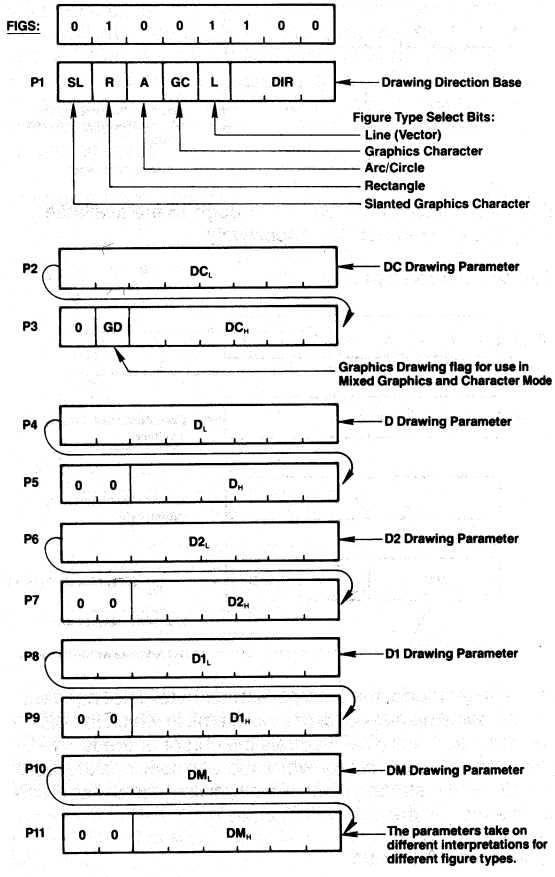
Mask Register Load



This command sets the value of the 16-bit Mask register of the figure drawing processor. The Mask register controls which bits can be modified in the display memory during a read-modify-write cycle.

The Mask register is loaded both by the MASK command and the third parameter byte of the CURS command. The MASK command accepts two parameter bytes to load a 16-bit value into the Mask register. All 16 bits can be individually one or zero, under program control. The CURS command on the other hand, puts a "1 of 16" pattern into the Mask register based on the value of the Dot Address value, dAD. If normal single-pixel-at-a-time graphics figure drawing is desired, there is no need to do a MASK command at all since the CURS command will set up the proper pattern to address the proper pixels as drawing progresses. For coded character DMA, and screen setting and clearing operations using the WDAT command, the MASK command should be used after the CURS command if its third parameter byte has been output. The Mask register should be set to all "ONES" for any "word-at-a-time" operation.

Figure Drawing Parameters Specify

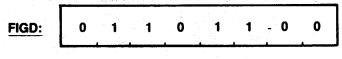


Valid Figure Type Select Combinations

SL	R	A	GC	L	Operation
0	0	0	0	0	Character Display Mode Drawing, Individual Dot Drawing, DMA, WDAT, and RDAT
0	0	0	0	1	Straight Line Drawing
0	0	0	1	0	Graphics Character Drawing and Area filling with graphics character pattern
0	0	1	0	0	Arc and Circle Drawing
0	1	0	0	0	Rectangle Drawing
1	0	0	1	0	Slanted Grapnics Character Drawing and Slanted Area Filling

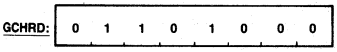
Only these bit combinations assure correct drawing operation.

Figure Draw Start



On execution of this instruction, the GDC loads the parameters from the parameter RAM into the drawing processor and starts the drawing process at the pixel pointed to by the cursor, EAD, and the dot address, dAD.

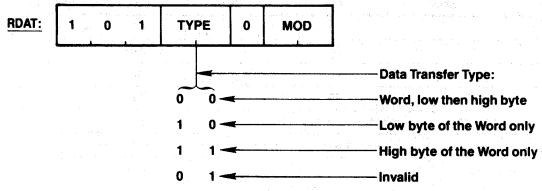
Graphics Character Draw and Area Filling Start



Based on parameters loaded with the FIGS command, this command initiates the drawing of the graphics character or area filling pattern stored in Parameter RAM. Drawing begins at the address in display memory pointed to by the EAD and dAD values.

Data Read Commands

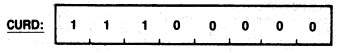
Read Data from Display Memory



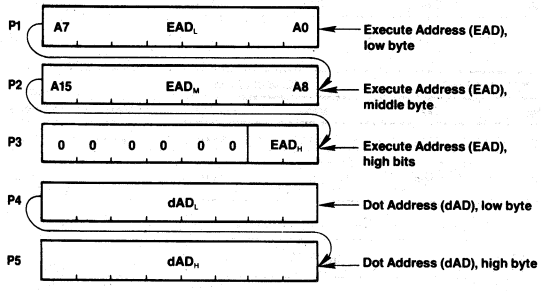
Using the DIR and DC parameters of the FIGS command to establish direction and transfer count, multiple RMW cycles can be executed without specification of the cursor address after the initial load (DC = number of words or bytes).

As this instruction begins to execute, the FIFO buffer direction is reversed so that the data read from display memory can pass to the microprocessor. Any commands or parameters in the FIFO at this time will be lost. A command byte sent to the GDC will immediately reverse the buffer direction back to write mode, and all RDAT information not yet read from the FIFO will be lost. MOD should be set to 00 if no modification to video buffer is desired.

Cursor Address Read



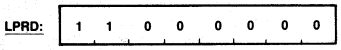
The following bytes are returned by the GDC through the FIFO:



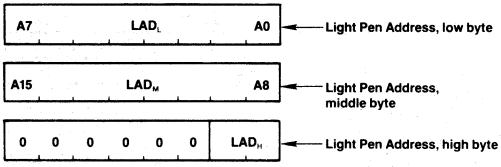
The Execute Address, EAD, points to the display memory word containing the pixel to be addressed.

The Dot Address, dAD, within the word is represented as a 1-of-16 code for graphics drawing operations.

Light Pen Address Read



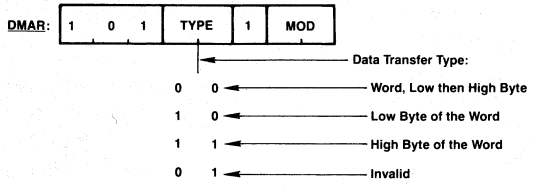
The following bytes are returned by the GDC through the FIFO:



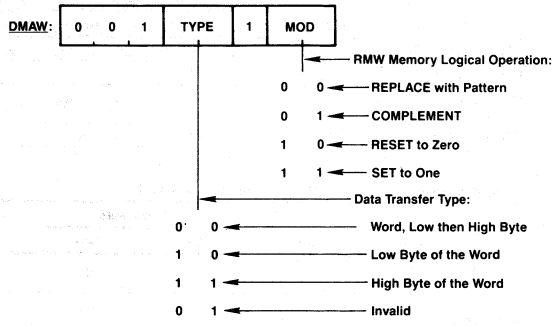
The light pen address, LAD, corresponds to the display word address, DAD, at which the light pen input signal is detected and deglitched.

The light pen may be used in graphics, character, or mixed modes but only indicates the word address of light pen position.

DMA Read Request



DMA Write Request



AC Characteristics, μPD7220D

T_a = 0°C to 70°C; V_{CC} = 5.0V ± 10%; GND = 0V

Read Cycle (GDC ↔ CPU)

Parameter	Symbol	7220D Limits Min	7220D Limits Max	7220D-1 Limits Min	7220D-1 Limits Max	7220D-2 Limits Min	7220D-2 Limits Max	Unit	Test Conditions
Address Setup to RD ↓	t_{AR}	0		0		0		ns	
Address Hold from RD ↑	t_{RA}	0		0		0		ns	
RD Pulse Width	t_{RR1}	t_{RD1} + 20	t_{RCY} − ½ t_{CLK}	t_{RD1} + 20	t_{RCY} − ½ t_{CLK}	t_{RD1} + 20	t_{RCY} − ½ t_{CLK}	ns	
Data Delay from RD ↓	t_{RD1}		120		80		70	ns	C_L = 50 pF
Data Floating from RD ↑	t_{DF}	0	120	0	100	0	90	ns	
RD Pulse Cycle	t_{RCY}	4 t_{CLK}		4 t_{CLK}		4 t_{CLK}		ns	

Write Cycle (GDC ↔ CPU)

Parameter	Symbol	7220D Limits Min	7220D Limits Max	7220D-1 Limits Min	7220D-1 Limits Max	7220D-2 Limits Min	7220D-2 Limits Max	Unit	Test Conditions
Address Setup to WR ↓	t_{AW}	0		0		0		ns	
Address Hold from WR ↑	t_{WA}	0		0		10		ns	
WR Pulse Width	t_{WW}	120		100		90		ns	
Data Setup to WR ↑	t_{DW}	100		80		70		ns	
Data Hold from WR ↑	t_{WD}	10		10		10		ns	
WR Pulse Cycle	t_{WCY}	4 t_{CLK}		4 t_{CLK}		4 t_{CLK}		ns	

DMA Read Cycle (GDC ↔ CPU)

Parameter	Symbol	7220D Limits Min	7220D Limits Max	7220D-1 Limits Min	7220D-1 Limits Max	7220D-2 Limits Min	7220D-2 Limits Max	Unit	Test Conditions
DACK Setup to RD ↓	t_{KR}	0		0		0		ns	
DACK Hold from RD ↑	t_{RK}	0		0		0		ns	
RD Pulse Width	t_{RR2}	t_{RD2} + 20		t_{RD2} + 20		t_{RD2} + 20		ns	
Data Delay from RD ↓	t_{RD2}		1.5 t_{CLK} + 120		1.5 t_{CLK} + 80		1.5 t_{CLK} + 70	ns	C_L = 50 pF
DREQ Delay from 2XWCLK ↑	t_{REQ}		150		120		110	ns	C_L = 50 pF
DREQ Setup to DACK ↓	t_{QK}	0		0		0		ns	
DACK High Level Width	t_{DK}	t_{CLK}		t_{CLK}		t_{CLK}		ns	
DACK Pulse Cycle	t_E	4 t_{CLK}*		4 t_{CLK}*		4 t_{CLK}*		ns	
DREQ ↓ Delay from DACK ↓	$t_{KQ(R)}$		t_{CLK} + 150		t_{CLK} + 120		t_{CLK} + 110	ns	C_L = 50 pF

* for high byte and low byte transfers: t_E = 5 t_{CLK}

DMA Write Cycle (GDC ↔ CPU)

Parameter	Symbol	7220D Limits Min	7220D Limits Max	7220D-1 Limits Min	7220D-1 Limits Max	7220D-2 Limits Min	7220D-2 Limits Max	Unit	Test Conditions
DACK Setup to WR ↓	t_{KW}	0		0		0		ns	
DACK Hold from WR ↑	t_{WK}	0		0		0		ns	
DREQ ↓ Delay from DACK ↓	$t_{KQ(W)}$		t_{CLK} + 150		t_{CLK} + 120		t_{CLK} + 100	ns	C_L = 50 pF
WR Pulse Width	t_{WW}	120	3 t_{CLK}	100	3 t_{CLK}	90	3 t_{CLK}	ns	

R/M/W Cycle (GDC ↔ Display Memory)

Parameter	Symbol	7220D Limits Min	7220D Limits Max	7220D-1 Limits Min	7220D-1 Limits Max	7220D-2 Limits Min	7220D-2 Limits Max	Unit	Test Conditions
Address/ Data Delay from 2XWCLK ↑	t_{AD}	30	160	30	130	30	115	ns	C_L = 50 pF
Address/Data Floating from 2XWCLK ↑	t_{OFF}	30	160	30	130	30	115	ns	C_L = 50 pF
Input Data Setup to 2XWCLK ↓	t_{DIS}	0		0		0		ns	
Input Data Hold from 2XWCLK ↓	t_{DIH}	t_{DE} − 20		t_{DE} − 20		t_{DE} − 20		ns	
DBIN Delay from 2XWCLK ↓	t_{DE}	30	120	30	90	30	80	ns	C_L = 50 pF
ALE ↑ Delay from 2XWCLK ↑	t_{RR}	30	125	30	100	30	90	ns	C_L = 50 pF

ALE ↓ Delay from 2XWCLK ↓	t_{RF}	30	100	30	80	30	70	ns	C_L = 50 pF
ALE Width	t_{RW}	1/3 t_{CLK}		1/3 t_{CLK}		1/3 t_{CLK}		ns	C_L = 50 pF
ALE Low Width	t_{RL}	t_{CLK} + 30		t_{CLK} + 30		t_{CLK} + 30		ns	

Display Cycle **(GDC ↔ Display Memory)**

Parameter	Symbol	7220D Limits Min	7220D Limits Max	7220D-1 Limits Min	7220D-1 Limits Max	7220D-2 Limits Min	7220D-2 Limits Max	Unit	Test Conditions
Video Signal Delay from 2XWCLK ↑	t_{VD}		150		120		100	ns	C_L = 50 pF

Input Cycle **(GDC ↔ Display Memory)**

Parameter	Symbol	7220D Limits Min	7220D Limits Max	7220D-1 Limits Min	7220D-1 Limits Max	7220D-2 Limits Min	7220D-2 Limits Max	Unit	Test Conditions
Input Signal Setup to 2XWCLK ↑	t_{PS}	30		20		15		ns	
Input Signal Width	t_{PW}	t_{CLK}		t_{CLK}		t_{CLK}		ns	

Clock **(2XWCLK)**

Parameter	Symbol	7220D Limits Min	7220D Limits Max	7220D-1 Limits Min	7220D-1 Limits Max	7220D-2 Limits Min	7220D-2 Limits Max	Unit	Test Conditions
Clock Rise Time	t_{CR}		20		20		20	ns	
Clock Fall Time	t_{CF}		20		20		20	ns	
Clock High Pulse Width	t_{CH}	105		80		70		ns	
Clock Low Pulse Width	t_{CL}	105		80		70		ns	
Clock Cycle	t_{CLK}	250	2000	200	2000	180	2000	ns	

DC Characteristics

T_a = 0°C to 70°C; V_{CC} = 5V ± 10%; GND = 0V

Parameter	Symbol	Limits Min	Typ	Max	Unit	Test Conditions
Input Low Voltage	V_{IL}	−0.5		0.8	V	①
Input High Voltage	V_{IH}	2.2		V_{CC} + 0.5	V	②
Output Low Voltage	V_{OL}			0.45	V	I_{OL} = 2.2 mA
Output High Voltage	V_{OH}	2.4			V	I_{OH} = −400 μA
Input Low Leak Current	I_{IL}			−10	μA	V_I = 0V
Input High Leak Current	I_{IH}			+10	μA	V_I = V_{CC}
Output Low Leak Current	I_{OL}			−10	μA	V_O = 0V
Output High Leak Current	I_{OH}			+10	μA	V_O = V_{CC}
Clock Input Low Voltage	V_{CL}	−0.5		0.6	V	
Clock Input High Voltage	V_{CH}	3.5		V_{CC} + 1.0	V	
V_{CC} Supply Current	I_{CC}			270	mA	

Capacitance

T_a = 25°C; V_{CC} = GND = 0V

Parameter	Symbol	Limits Min	Typ	Max	Unit	Test Conditions
Input Capacitance	C_{IN}			10	pF	fc = 1 MHz V_I (unmeasured) = 0V
I/O Capacitance	$C_{I/O}$			20	pF	
Output Capacitance	C_{OUT}			20	pF	
Clock Input Capacitance	C_ϕ			20	pF	

Notes:
① For 2XWCLK, V_{IL} = −0.5V to +0.6V.
② For 2XWCLK, V_{IH} = +3.9V to V_{CC} +1.0V.

Absolute Maximum Ratings* (Tentative)

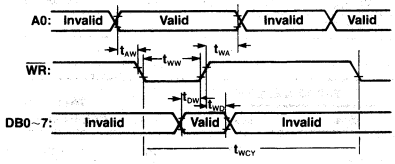
Ambient Temperature under Bias	0°C to 70°C
Storage Temperature	−65°C to 150°C
Voltage on Any Pin with Respect to Ground	−0.5V to +7V
Power Dissipation	1.5 W

*COMMENT: Exposing the device to stresses above those listed in Absolute Maximum Ratings could cause permanent damage. The device is not meant to be operated under conditions outside the limits described in the operational sections of this specification. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

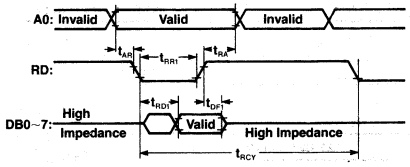
6

Timing Waveforms

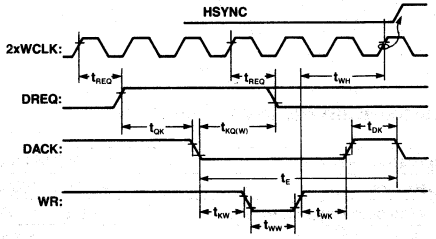
Microprocessor Interface Write Timing



Microprocessor Interface Read Timing



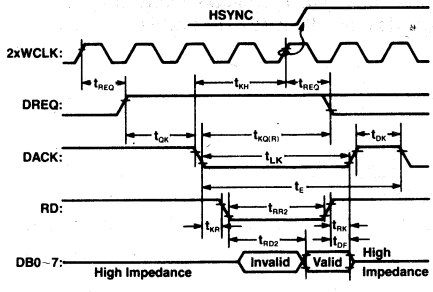
Microprocessor Interface DMA Write Timing



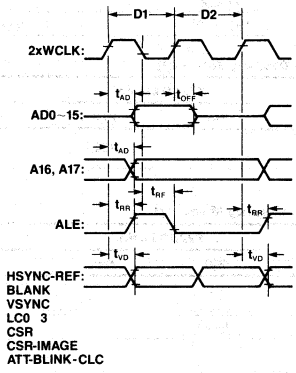
t_{WH} (WR↑ to HSYNC↑) ≥ t_{CLK}

t_{KH} (DACK↓ to HSYNC↑) ≥ t_{CLK}

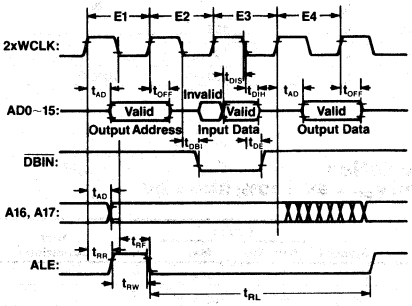
Microprocessor Interface DMA Read Timing



Display Memory Display Cycle Timing

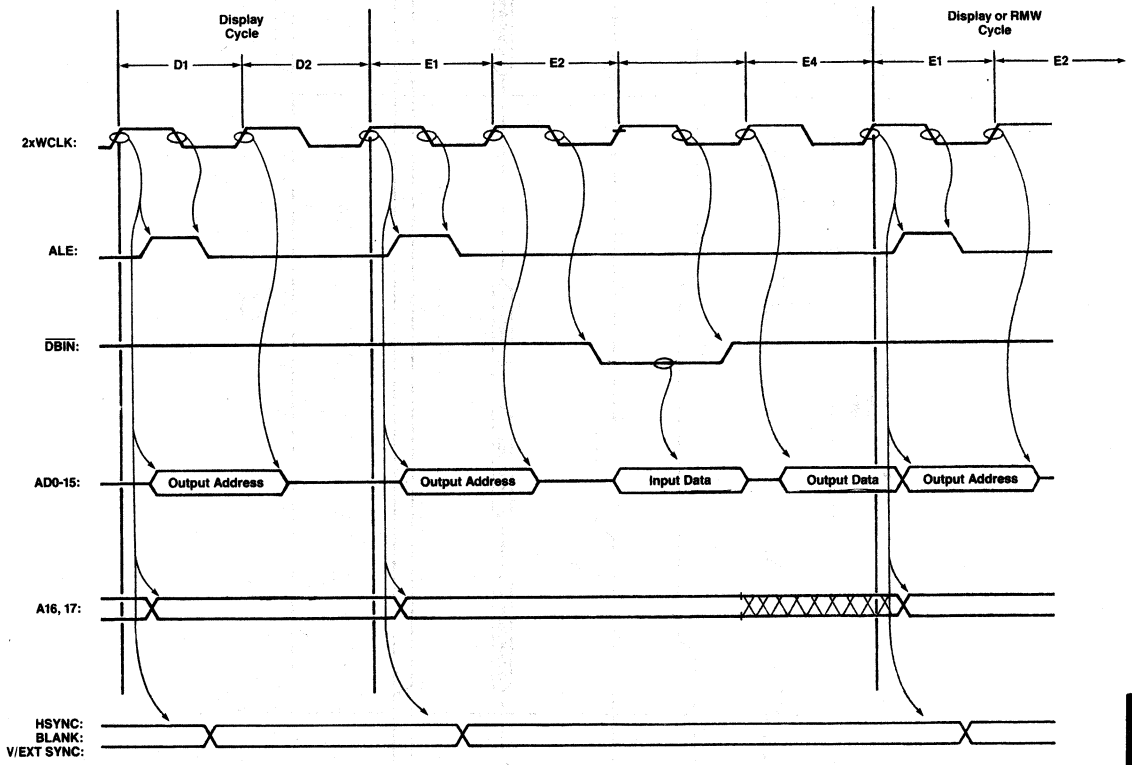


Display Memory RMW Timing



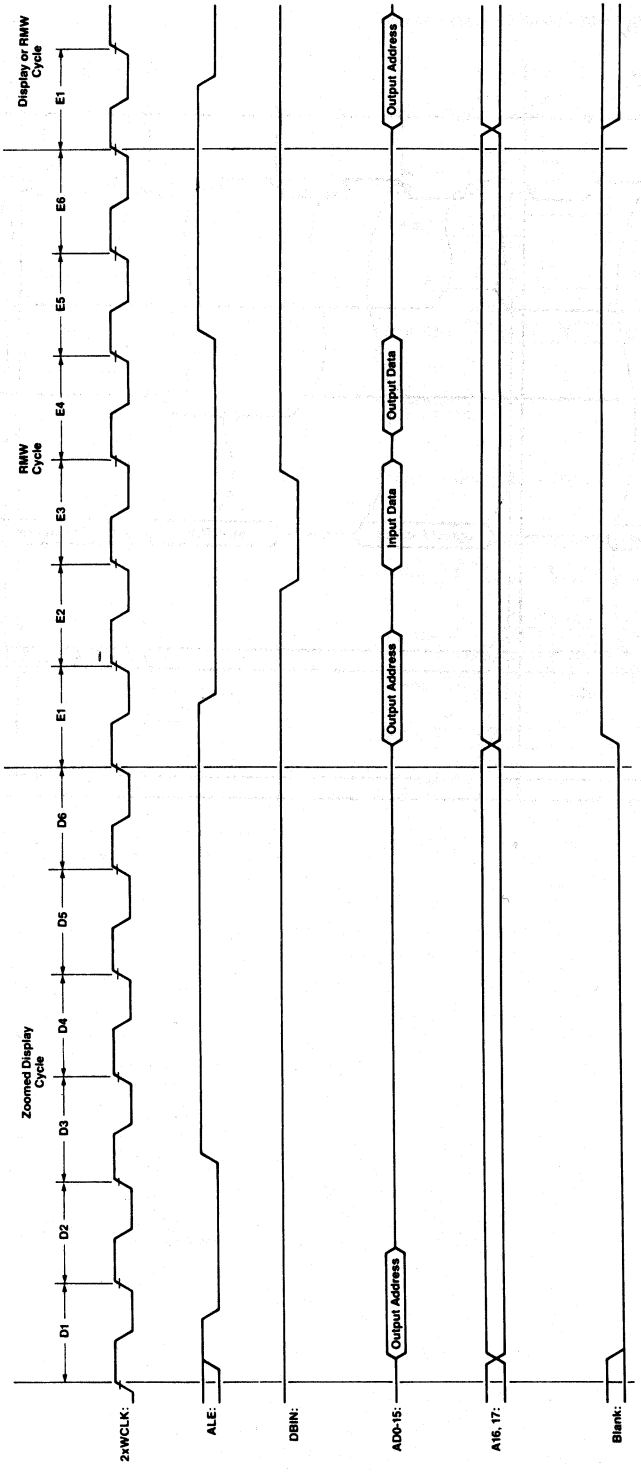
Timing Waveforms (Cont.)

Display and RMW Cycles (1x Zoom)

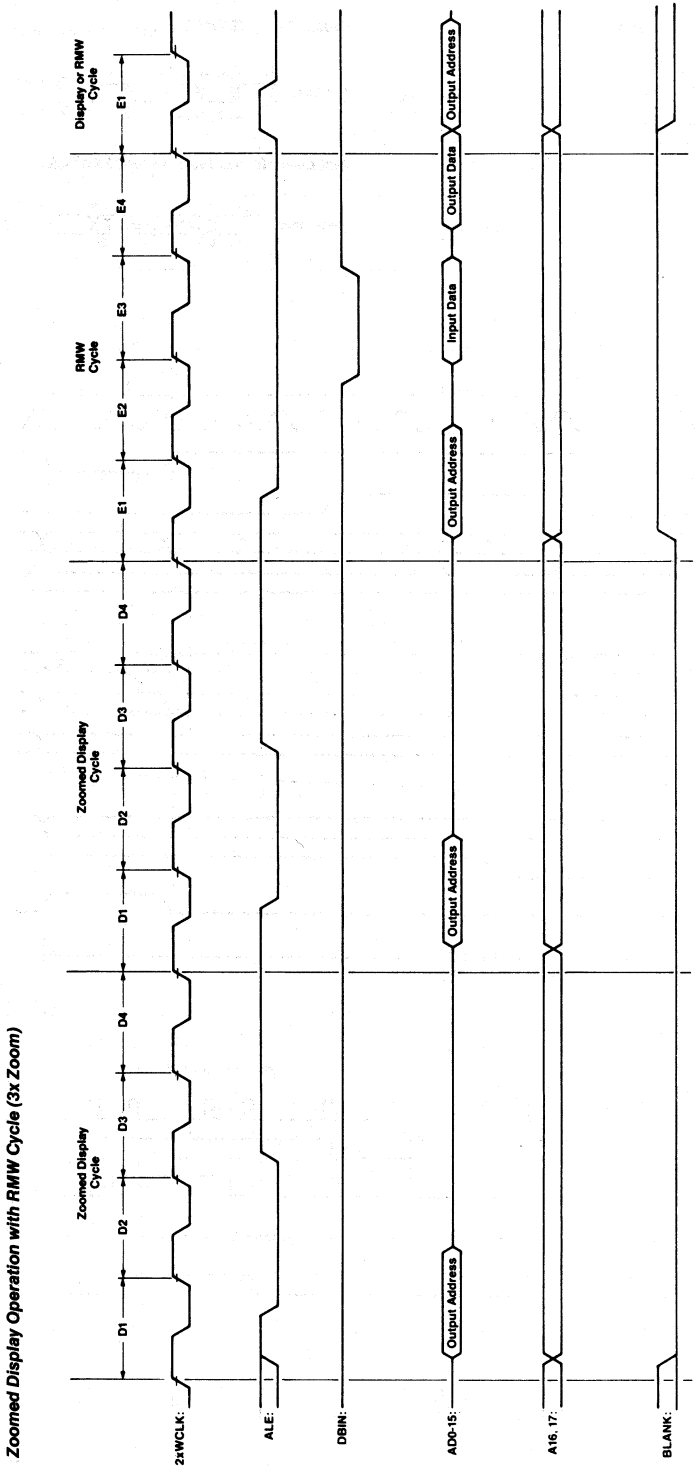


Timing Waveforms (Cont.)

Display and RMW Cycles (2x Zoom)



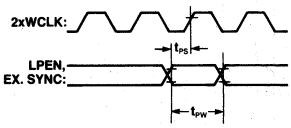
Timing Waveforms (Cont.)



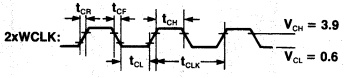
Zoomed Display Operation with RMW Cycle (3x Zoom)

Timing Waveforms (Cont.)

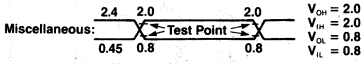
Light Pen and External Sync Input Timing



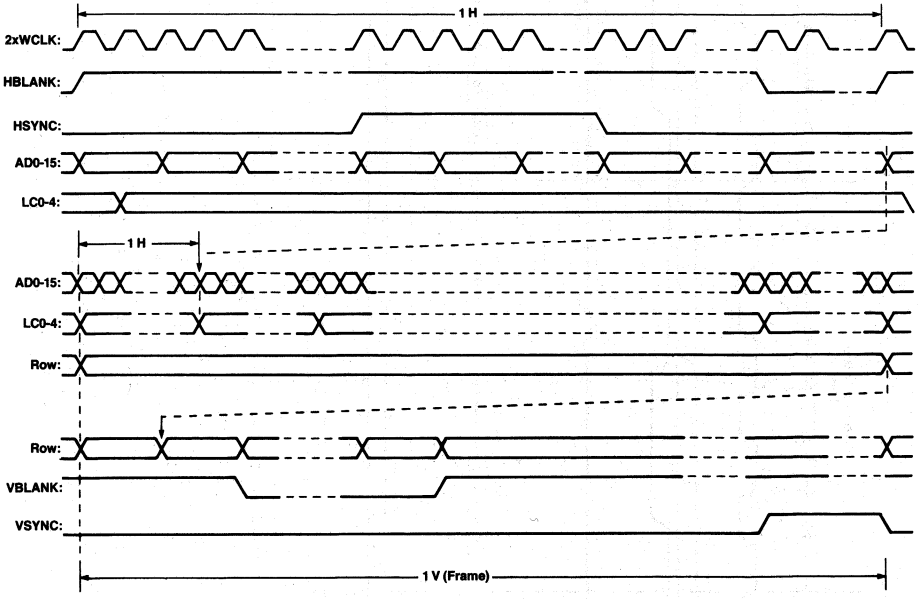
Clock Timing (2XWCLK)



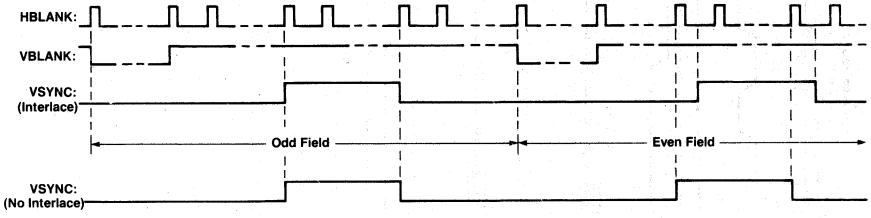
Test Level (for AC Tests, except 2XWCLK)



Video Sync Signals Timing

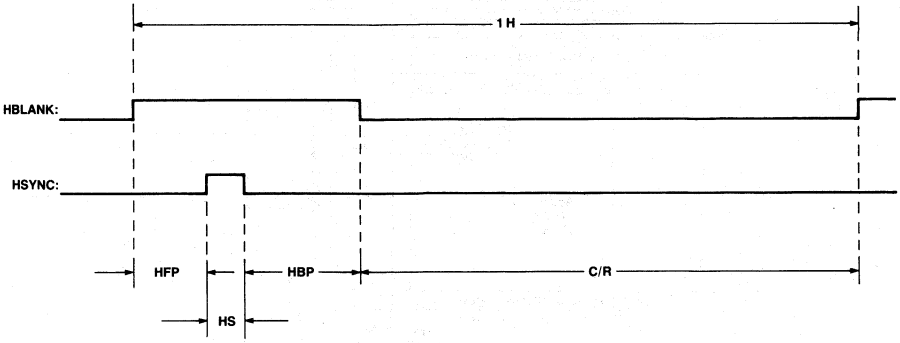


Interlaced Video Timing

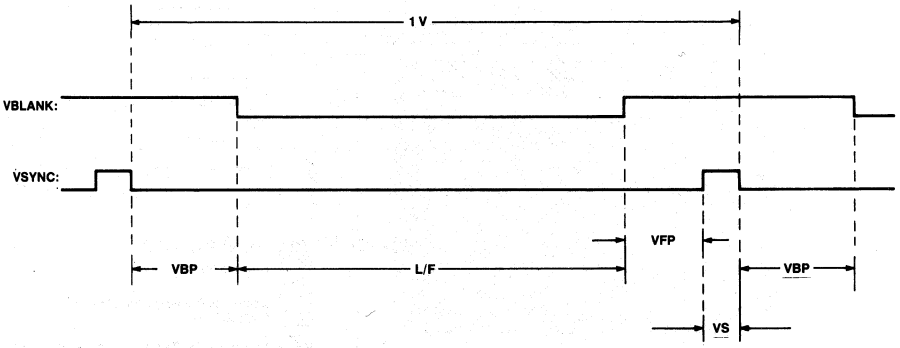


Timing Waveforms (Cont.)

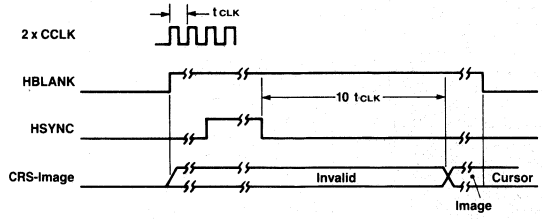
Video Horizontal Sync Generator Parameters



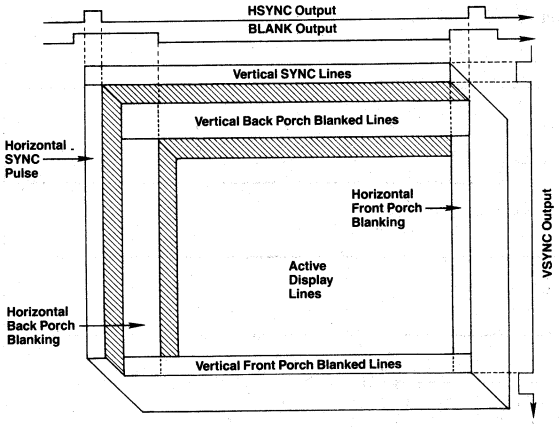
Video Vertical Sync Generator Parameters



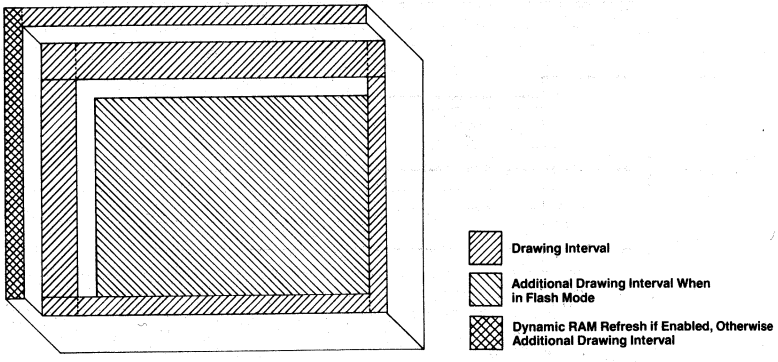
Cursor — Image Bit Flag



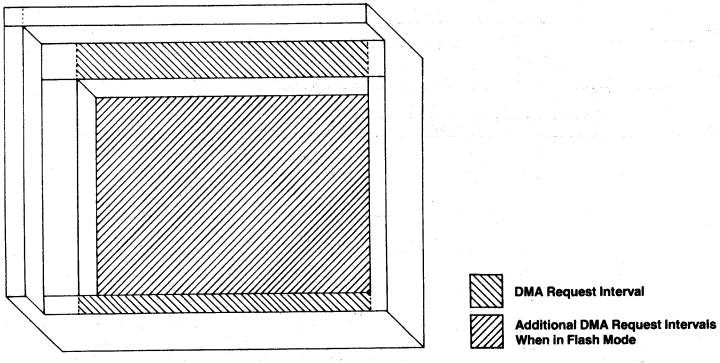
Video Field Timing



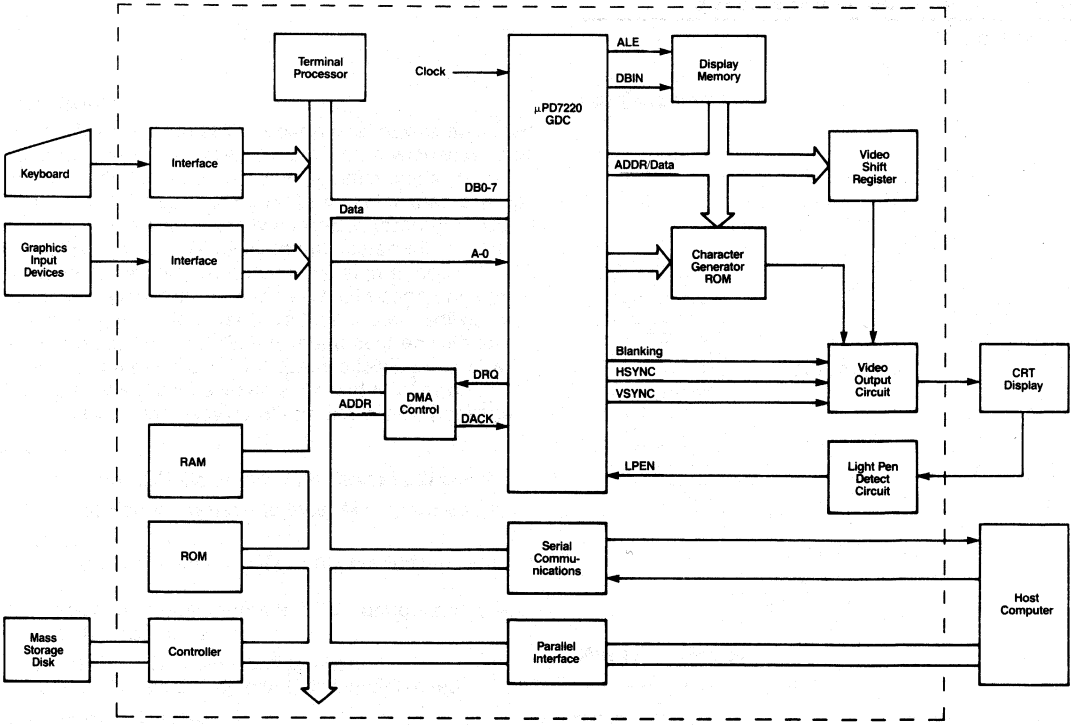
Drawing Intervals



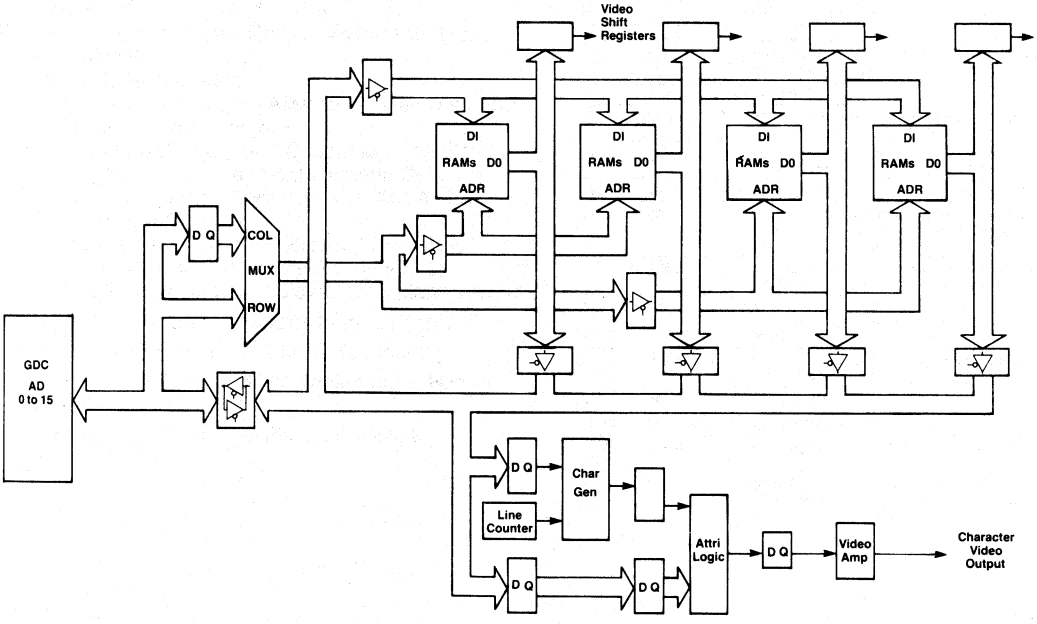
DMA Request Intervals



Block Diagram of a Graphics Terminal



Multiplane Display Memory Diagram



6

Package Outlines

For information, see Package Outline Section 7.

Ceramic, μPD7220D