

HD63484

Advanced CRT Controller (ACRTC)

The advanced CRT controller (ACRTC) CMOS VLSI microcomputer peripheral device can display both graphics and characters on raster-scan displays. It is a new generation CRT controller based on bitmapped technology. It executes high-level commands, like Line, Ellipse, Paint, Pattern, and Copy, issued by the MPU in screen X-Y coordinates, and performs the address translation necessary to draw into frame memory. It can draw in up to 64k colors, on three split screens and an independent window, and perform area clipping and hitting.

The ACRTC controls a CRT in one of three modes: character only, graphics only, and multiplexed character/graphics modes. Therefore, the ACRTC has many applications, from character-only displays to large full-graphics systems.

The ACRTC reduces CPU software overhead and enhances system throughput.

Features

- High-speed graphics
 - Drawing rate: 408 ns/pixel max (color drawing)
 - Commands: 38 commands including 23 graphic drawing commands
Dot, Line, Rectangle, Poly-line, Polygon, Circle, Ellipse, Paint, Copy, etc.
- Colors: 16 bits/word
 - 1, 2, 4, 8, 16 bits/pixel (5 types)
 - Monochrome to 64k colors max
- Pattern RAM: 32 bytes
- Converts logical X-Y coordinates to physical address
- Color operation and conditional drawing
- Drawing area control for hardware clipping and hitting
- Large frame-memory space
 - Maximum 2 Mbytes graphic memory and 128 kbytes character memory separate from the MPU memory
 - Maximum resolution: 4096×4096 pixels (1 bit/pixel mode)
- CRT display control
 - Split screens: three displays and one window
 - Zoom: 1 to 16 times
 - Scroll: vertical and horizontal
- Interleaved access mode for flashless display and superimposition
- External synchronization between ACRTCs or between ACRTC and external device (TV system or other controller)
- DMA interface
- Two programmable cursors
- Three scan modes
 - Non-interlaced
 - Interlace sync
 - Interlace sync and video
- Interrupt request to MPU
- 256 characters/line 32 raster/line, 4096 rasters/screen
- Maximum clock frequency: 9.8 MHz
- CMOS, single +5 V power supply

Ordering Information

| Part No. | Clock Frequency (2CLK) | Package |
|-------------|------------------------|-----------------------------|
| HD63484-4 | 4 MHz | DC-64 |
| HD63484-6 | 6 MHz | (64-pin ceramic DIP) |
| HD63484-8 | 8 MHz | |
| HD63484-98 | 9.8 MHz | |
| HD63484P4 | 4 MHz | DP-64 |
| HD63484P6 | 6 MHz | (64-pin plastic DIP) |
| HD63484P8 | 8 MHz | |
| HD63484P98 | 9.8 MHz | |
| HD63484CP4 | 4 MHz | CP-68 |
| HD63484CP6* | 6 MHz | (68-pin plastic PLCC) |
| HD63484CP8 | 8 MHz | |
| HD63484CP98 | 9.8 MHz | |
| HD63484Y4 | 4 MHz | PC-68 |
| HD63484Y6 | 6 MHz | (68-pin PGA) |
| HD63484Y8 | 8 MHz | |
| HD63484Y98 | 9.8 MHz | |
| HD63484PS4 | 4 MHz | DP-64S |
| HD63484PS6 | 6 MHz | (64-pin plastic shrink DIP) |
| HD63484PS8 | 8 MHz | |
| HD63484PS98 | 9.8 MHz | |

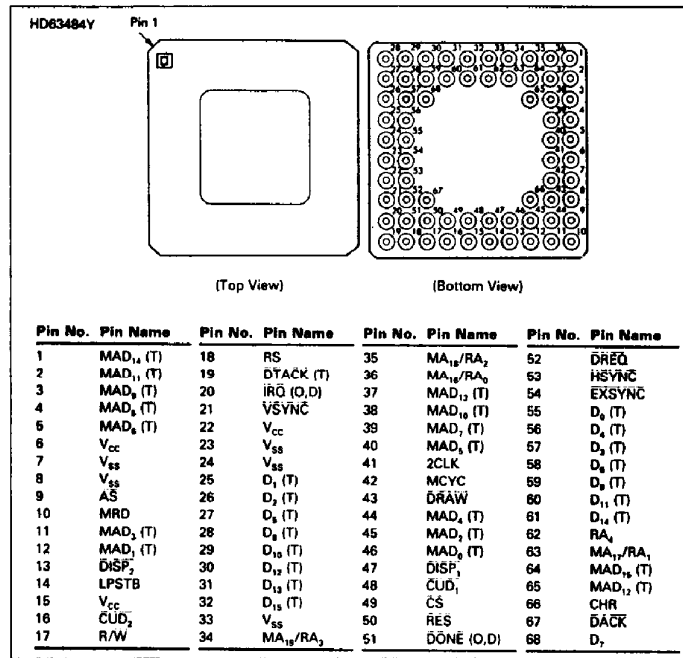
Note: Wide temperature range (−40C° to +80C°) version is available.



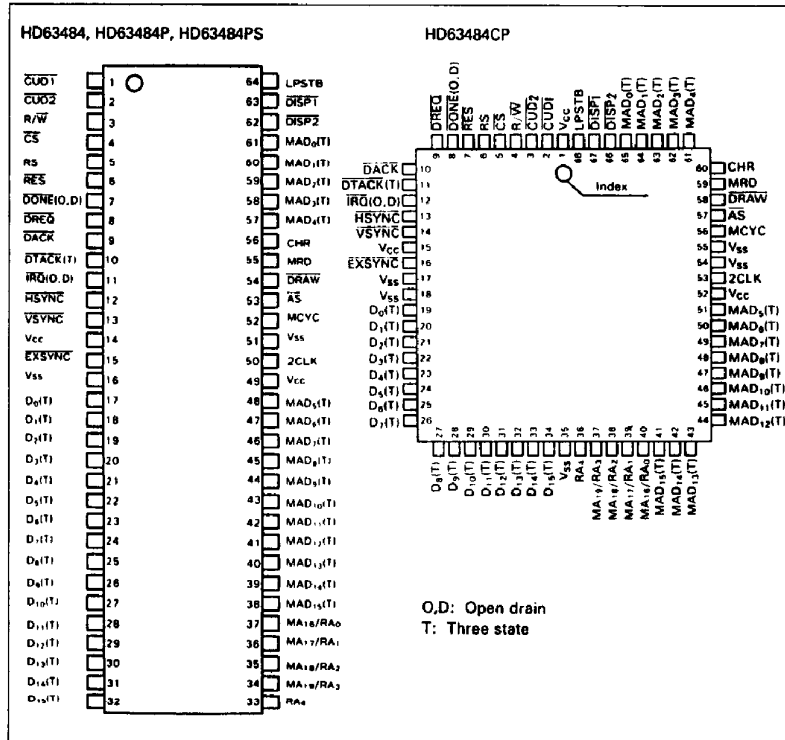
Hitachi America, Ltd. • Hitachi Plaza • 2000 Sierra Point Pkwy. • Brisbane, CA 94005-1819 • (415) 589-8300

213

2



Pin Arrangement



Pin Description

| Group | Mnemonic | Pin Number | | | I/O | Function |
|----------------|---|-----------------|--------------------------|---------------------------------------|-----------------|---|
| | | DIP | PLCC | PGA | | |
| MPU Interface | RES | 6 | 7 | 50 | I | ACRTC reset |
| | D ₀ -D ₁₅ * | 17-32 | 19-34 | 25-32 55-61 68 | I/O | Data bus (three state) |
| | R/W | 3 | 4 | 17 | I | Read/write strobe |
| | CS | 4 | 5 | 49 | I | Chip select |
| | RS | 5 | 6 | 18 | I | Register select |
| | DTACK | 10 | 11 | 19 | O | Data transfer acknowledge (three state) |
| | IRQ | 11 | 12 | 20 | O | Interrupt request (open drain) |
| DMAC Interface | DREQ | 8 | 9 | 52 | O | DMA request |
| | DACK | 9 | 10 | 67 | I | DMA acknowledge |
| | DONE | 7 | 8 | 51 | I/O | DMA done (open drain) |
| CRT Interface | 2CLK | 50 | 53 | 41 | I | ACRTC clock |
| | MAD ₀ -MAD ₁₅ * | 61-57, 48-38 | 65-61, 51-41 | 1-5, 11,12 37-40,44- 46, 64, 65 | I/O | Multiplexed frame buffer address/data bus |
| | AS | 53 | 57 | 9 | O | Adress strobe |
| | MA ₁₆ /RA ₀ -* MA ₁₉ /RA ₃ | 37-34 | 40-37 | 34-36 | O | Higher-order address bits/ character screen raster address |
| | RA ₄ | 33 | 36 | 62 | O | High-order character screen raster address bit |
| | CHR | 56 | 60 | 66 | O | Graphic or character screen access |
| | MCYC | 52 | 56 | 42 | O | Frame buffer memory access timing signal |
| | MRD | 55 | 59 | 10 | O | Frame buffer memory read |
| | DRAW | 54 | 58 | 43 | O | Draw/refresh signal |
| | DISP1, DISP2 | 63, 62 | 67, 66 | 47, 13 | O | Display enable |
| | CUD1, CUD2 | 1,2 | 2,3 | 48,16 | | Cursor display |
| | VSYNC | 13 | 14 | 21 | O | CRT vertical sync pulse |
| | HSYNC | 12 | 13 | 53 | | CRT horizontal sync pulse |
| | EXSYNC | 15 | 16 | 54 | I/O | External sync |
| LPSTB | 64 | 68 | 14 | I | Lightpen strobe | |
| Power Supply | V _{cc} | 14, 49 | 1, 15, 52 | 6, 15, 22 | | +5 V |
| | V _{ss} | 16, 51 | 17, 18, 35, 54, 55 | 7, 8, 23 24, 33 | | Ground |

*: PGA pin numbers don't correspond to D₀-D₁₅, MAD₀-MAD₁₅, MA₁₆/RA₀-MA₁₉/RA₃. Please refer to the pin arrangement.



MPU Interface

\overline{RES} (Reset): \overline{RES} is the MPU hardware reset.

D_0 - D_{15} (Data Bus): D_0 - D_{15} are the bidirectional data bus to/from the host MPU or DMAC. D_0 - D_7 are used in 8-bit data bus mode.

R/\overline{W} (Read/Write): R/\overline{W} input controls the direction of host/ACRTC transfers.

\overline{CS} (Chip Select): \overline{CS} input enables transfers between the host and the ACRTC.

RS(Register Select): RS input selects the ACRTC register to be accessed. It is usually connected to the least significant bit of the host address bus.

\overline{DTACK} (Data Transfer Acknowledge): \overline{DTACK} output provides asynchronous bus cycle timing. It is compatible with the HD68000 MPU \overline{DTACK} output.

\overline{IRQ} (Interrupt Request): \overline{IRQ} output generates interrupt service requests to the host MPU.

DMAC Interface

\overline{DREQ} (DMA Acknowledge): \overline{DACK} receives DMA acknowledge timing from the host DMAC.

\overline{DONE} (DMA Done): \overline{DONE} terminates DMA transfer. It is compatible with the HD68450 DMAC \overline{DONE} signal.

CRT Interface

2CLK (Dot Clock): 2CLK is the basic ACRTC operating clock, twice the frequency of the dot clock.

MAD_0 - MAD_{15} (Frame Memory Address/Data Bus): MAD_0 - MAD_{15} are the multiplexed frame buffer address/data bus.

\overline{AS} (Address Strobe): The \overline{AS} output demultiplexes the address/data bus (MAD_0 - MAD_{15}).

MA_{16}/RA_0 - MA_{19}/RA_3 (Memory Address/Raster Address): MA_{16}/RA_0 - MA_{19}/RA_3 are the upper bits of the graphics screen address multiplexed with the lower bits of the character screen raster address.

RA_4 (Raster Address): RA_4 is the high bit

of the character screen raster address (up to 32 rasters).

CHR (Character): CHR output indicates whether a graphic or character screen is being accessed.

MCYC (Memory Cycle): MCYC is the frame buffer memory access timing output, one-half the frequency of 2CLK.

MRD (Memory Read): MRD output controls the frame buffer data bus direction.

\overline{DRAW} (Draw): \overline{DRAW} output differentiates between drawing and CRT display refresh cycles.

$\overline{DISP1}$, $\overline{DISP2}$ (Display 1, 2): The $\overline{DISP1}$ and $\overline{DISP2}$ programmable display enable outputs can enable, disable, and blank logical screens.

$\overline{CUD1}$, $\overline{CUD2}$ (Cursor Display 1, 2): $\overline{CUD1}$ and $\overline{CUD2}$ outputs provides cursor timing programmed by ACRTC parameters such as cursor definition, cursor mode, cursor address, etc.

\overline{VSYNC} (Vertical Sync): \overline{VSYNC} outputs the CRT vertical synchronization pulse.

\overline{HSYNC} (Horizontal Sync): \overline{HSYNC} outputs the CRT horizontal synchronization pulse.

\overline{EXSYNC} (External Sync): \overline{EXSYNC} allows synchronization between multiple ACRTCs and other video signal generators.

LPSTB (Lightpen Strobe): LPSTB is the lightpen input.

Block Diagrams

ACRTC Functions

The ACRTC consists of 5 major functional blocks (figure 1). They operate in parallel to achieve maximum performance.

MPU Interface: The MPU interface interfaces asynchronously with the host MPU. Its functions include programmable interrupts handling, and DMA handshaking control.

CRT Interface: The CRT interface manages the frame buffer bus and the CRT timing input and output control signals. It also selects display refresh or drawing address outputs.



Drawing Processor: The drawing processor interprets commands and command parameters issued by the host bus (MPU and/or DMAC) and performs drawing operations on the frame buffer memory. It executes ACRTC drawing algorithms and converts logical X-Y addresses to physical frame buffer addresses.

It communicates with the host bus via separate 16-byte read and write FIFOs.

Display Processor: The display processor manages frame buffer refresh addressing based on the user-specified display screen organization. It combines and displays as many as 4 independent screen segments (3 horizontal split screens and 1 window) using an internal high-speed address calculation unit. It controls display refresh outputs in graphic (physical frame buffer address) or character (physical refresh memory address + row address) modes.

Timing Processor: The timing processor generates the CRT synchronization signals and signals used internally by the ACRTC.

Registers: The ACRTC registers that are visible to software are partitioned in the same way. They reside in the internal processor appropriate to their function. The registers in the display and timing processors are loaded with the basic display parameters during system initialization. During operation, the host communicates primarily with the ACRTC's drawing processor via the on-chip FIFOs.

High-Speed (= 9.8 MHz) Version of ACRTC

To keep up with the demand for improvements in the quality and resolution of CRT monitors, Hitachi has introduced a 9.8 MHz version of the HD63484 (ACRTC).

It can be used for:

1. High-resolution displays such as in office workstations, business personal computers, and CAD/CAM displays.
2. Applications requiring faster drawing than the current ACRTC with an 8-MHz 2CLK operation frequency.

High-Resolution Display

As shown in figure A, the 9.8-MHz allows the following configurations for a 4-bit/pixel system:

1. CRT monitor with 1024 x 808 dots, + 8 DA or + 4 SA
2. CRT monitor from 800 x 480 to 960 x 720 dots, + 4 DA

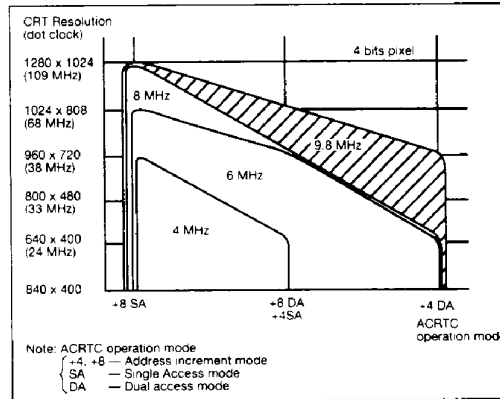


Figure A. ACRTC Operation Frequency and Supportable CRT Display Range

High-Speed Drawing Support

The ACRTC drawing speed depends on its operation frequency. Consequently it takes less time to draw with a 9.8-MHz clock than with an 8-MHz clock. Figure B compares drawing capabilities of the 9.8-MHz ACRTC and the 8-MHz ACRTC.

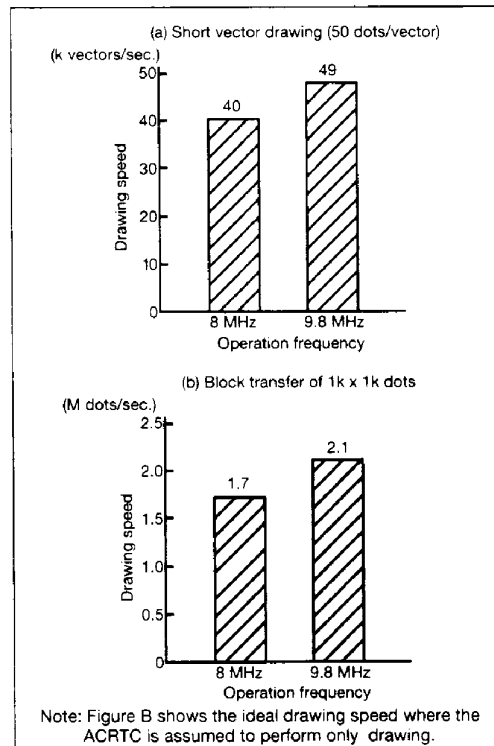


Figure B. Drawing Capability Comparison between 9.8-MHz and 8-MHz ACRTCs.



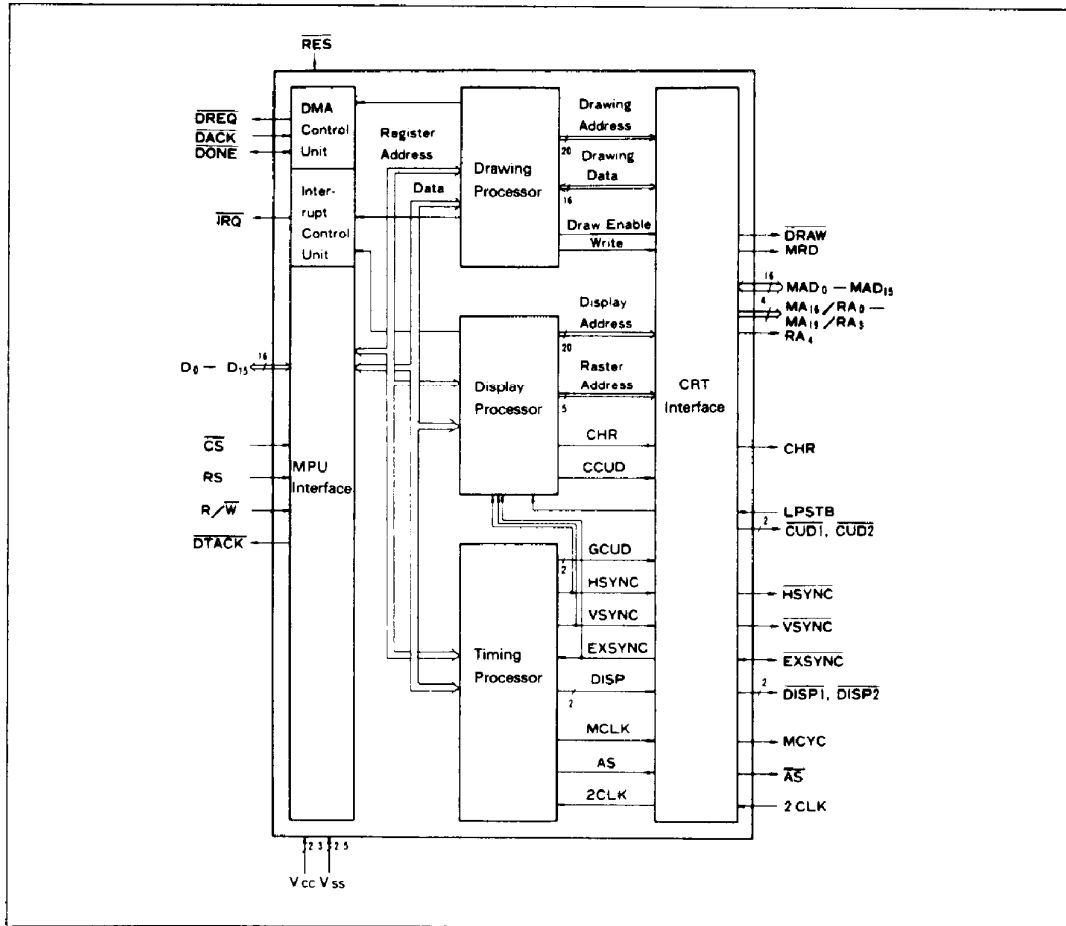


Figure 1 ACRTC Block Diagram

System Configuration

Current CRT controllers provide a single bus interface to the frame buffer that must be shared with the host MPU. However, refreshing large frame buffers, and accessing the frame buffer for drawing operations can quickly saturate the shared bus.

As shown in figure 2, the ACRTC uses separate host MPU and frame buffer interfaces. This allows the ACRTC full access to the frame buffer for display refresh and drawing operations and minimizes the ACRTC's use of the MPU system bus. A related benefit is that a large frame buffer (2 Mbyte for each ACRTC) can be used, even if the host MPU has a smaller address space or segment size restriction.

The ACRTC can use an external DMA controller. A DMA controller increases system throughput when many commands, parameters, and data must be transferred to the ACRTC. Advanced DMAC features, such as the HD68450 "chaining" modes can be used to develop powerful graphics system architectures.

However, more cost-sensitive or less performance-sensitive applications do not require a DMAC. The interface to the ACRTC can be handled under MPU software control.

While both ACRTC bus interfaces (host MPU and frame buffer) are 16 bits wide, the ACRTC also offers an 8-bit MPU mode for easy connection to popular 8-bit busses.



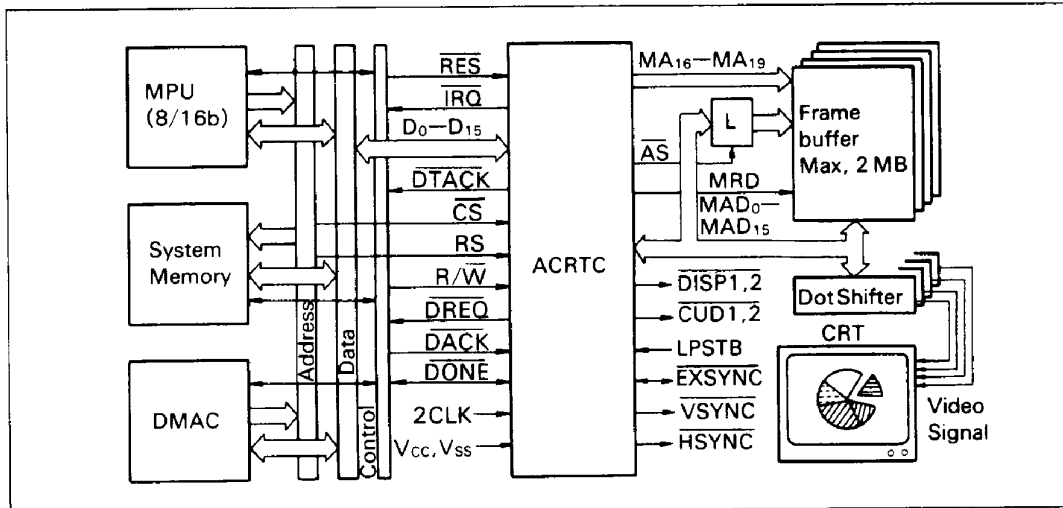


Figure 2 System Configuration

Programming Model

Address Space

The ACRTC allows the host to issue commands in logical X-Y coordinates. The ACRTC converts these physical linear word addresses with bit field offsets in the frame buffer. Figure 3 shows the relationship between the logical X-Y screen address and the frame buffer memory, which is organized as sequential 16-bit words. The host may specify logical pixels of 1, 2, 4, 8, or 16 physical bits in the frame buffer. The system in figure 3 uses 4-bit logical pixels, allowing 16 colors or tones to be selected.

Up to four logical screens (upper, base, lower, and window) are mapped onto the ACRTC physical address space. The host specifies a logical screen physical start address, logical screen physical memory width (memory words per raster), logical pixel physical memory width (bit per pixel), and the logical origin physical address. Then the ACRTC converts logical pixel X-Y addresses issued by the host MPU or the ACRTC drawing processor to physical frame buffer addresses. The ACRTC also performs bit extraction and masking to map logical pixel

operations (for example, 4 bits), to 16-bit word frame buffer addresses.

Registers

The ACRTC has over 200 bytes of accessible registers (figures 4, 5 and table 1). They are organized as hardware access, direct access, and FIFO access.

Hardware Access: The ACRTC is connected to the host MPU as a standard memory-mapped peripheral that occupies two word locations of the host's address space. The RS (register select) pin selects one of these two locations. When RS = 0 (low), read operations access the status register, and write operations access the address register.

The status register summarizes the ACRTC state. It monitors the overall state of the ACRTC for the host MPU.

When the MPU wants to access a direct access register, it puts the register's address

2



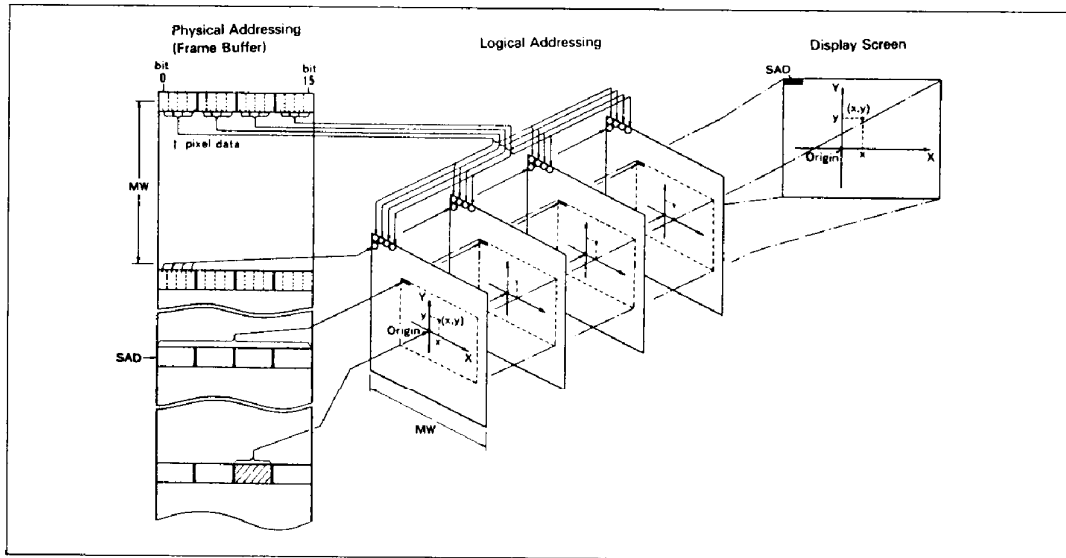


Figure 3. Logical/Physical Addressing

into the ACRTC address register.

Direct Access: The MPU accesses the direct access registers by first loading the register address into the address register. Then, when the MPU accesses the ACRTC with RS = 1 (high), the chosen register is accessed.

The FIFO entry register enables the MPU to access FIFO access registers using the ACRTC read and write FIFOs.

The command control register controls overall ACRTC operations, such as aborting or pausing commands, defining DMA protocols, and enabling/disabling interrupt sources.

The operation mode register defines basic parameters of ACRTC operation, such as frame buffer access mode, display or drawing priority, cursor and display timing skew factors, and raster scan mode.

The display control register independently enables and disables the four ACRTC logical address screens (upper, base, lower, and window). It also contains 8 user-defined video attribute bits.

The timing control RAM registers define ACRTC timing, including timing specifications for CRT control signals (HSYNC, VSYNC, etc), logical display screen size and

display period, and blink period.

The display control RAM contains registers which define logical screen display parameters, such as start address, raster address, and memory width. It also includes the cursor(s) definition, zoom factor, and lightpen registers.

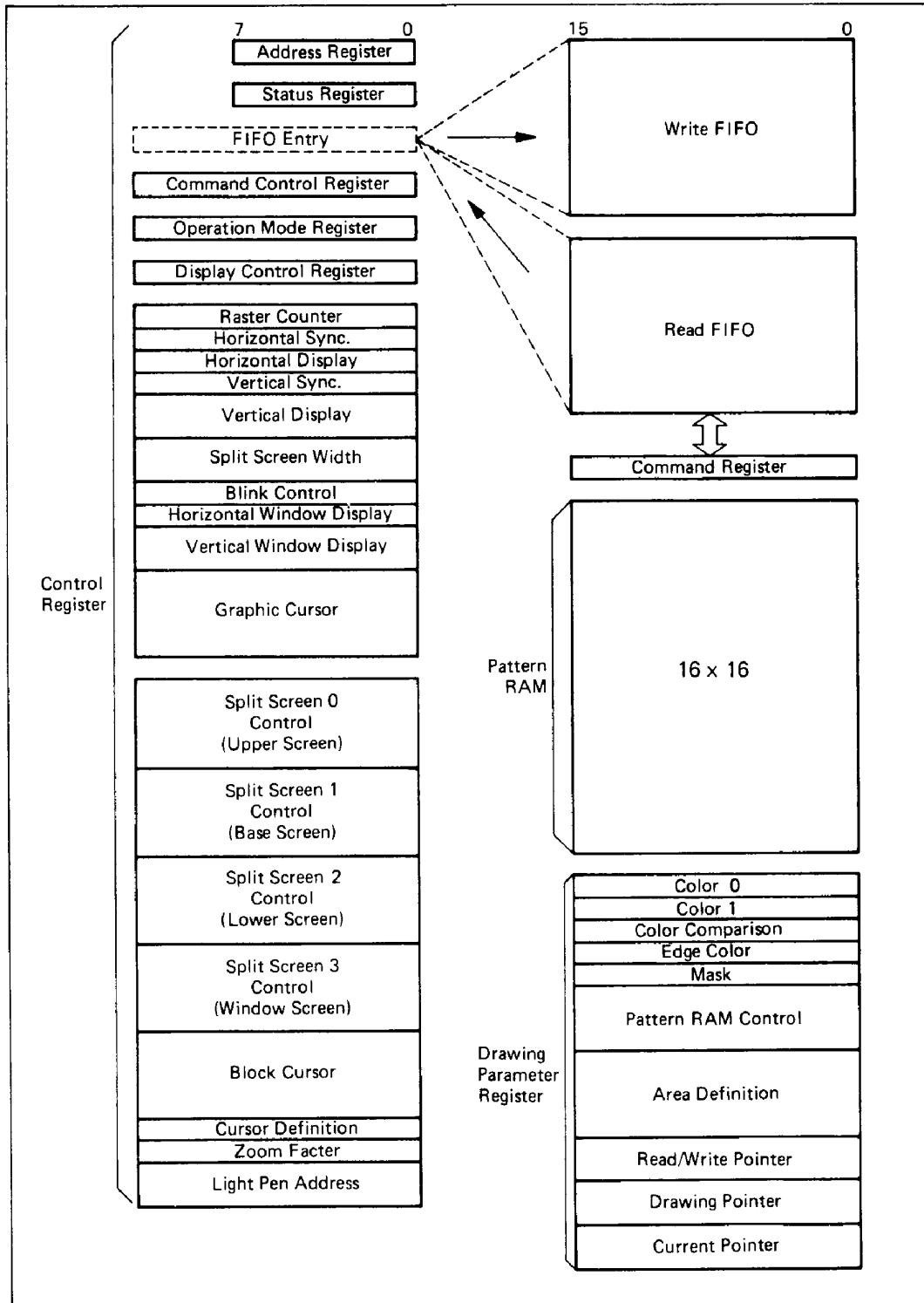
FIFO Access: For high-performance drawing, key drawing processor registers are coupled to the host MPU via the ACRTC's 16-byte read and write FIFOs.

ACRTC commands are sent from the MPU via the write FIFO to the command register. As the ACRTC completes a command, the next command is automatically fetched from the write FIFO and put into the command register.

The pattern RAM defines drawing and painting patterns. It is accessed with the ACRTC's Read Pattern RAM (RPTN) and Write Pattern RAM (WPTN) register access commands.

The drawing parameter registers define detailed parameters of the drawing process, such as color data, area control (hitting/clipping), and pattern RAM pointers. The drawing parameter registers are accessed using the ACRTC's Read Parameter Register (RPR) and Write Parameter Register (WPR) commands.





2

Figure 4. Programming Model



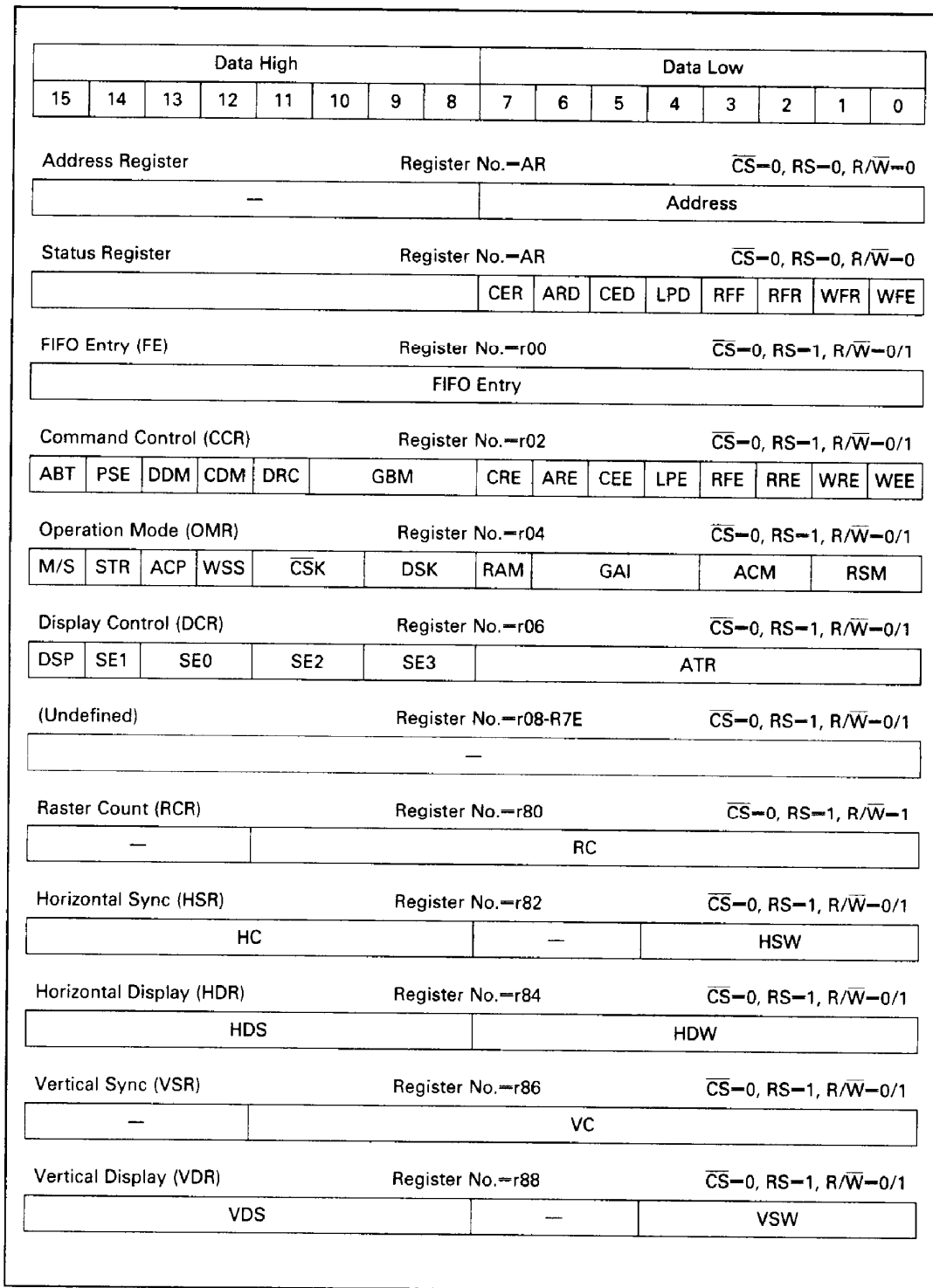


Figure 5. Hardware Access and Direct Access Registers



| | | | | | |
|---------------------------------|-------|--------------------------|-----------|---|--|
| Split Screen Width (SSW) | | Register No.—8A, 8C, 8E | | $\overline{CS}=0, RS=1, R/\overline{W}=0/1$ | |
| — | SP1 | | | | |
| — | SP0 | | | | |
| — | SP2 | | | | |
| Blink Control (BCR) | | Register No.—r90 | | $\overline{CS}=0, RS=1, R/\overline{W}=0/1$ | |
| BON1 | BOFF1 | BON2 | BOFF2 | | |
| Horizontal Window Display (HWR) | | Register No.—r92 | | $\overline{CS}=0, RS=1, R/\overline{W}=0/1$ | |
| HWS | | | HWW | | |
| Vertical Window Display (VDR) | | Register No.—r94, 96 | | $\overline{CS}=0, RS=1, R/\overline{W}=0/1$ | |
| — | VWS | | | | |
| — | VWW | | | | |
| Graphic Cursor (GCR) | | Register No.—r98, 9A, 9C | | $\overline{CS}=0, RS=1, R/\overline{W}=0/1$ | |
| CXE | | | CXS | | |
| — | CYS | | | | |
| — | CYE | | | | |
| (Underfined) | | Register No.—r9E-BE | | $\overline{CS}=0, RS=1, R/\overline{W}=0/1$ | |
| — | | | | | |
| Upper Screen | | | | | |
| Raster Address 0 (RAR0) | | Register No.—rC0 | | $\overline{CS}=0, RS=1, R/\overline{W}=0/1$ | |
| — | LRA0 | — | FRA0 | | |
| Memory Width 0 (MWR0) | | Register No.—C2 | | $\overline{CS}=0, RS=1, R/\overline{W}=0/1$ | |
| CHR | — | MW0 | | | |
| Start Address 0 (SAR0) | | Register No.—rC4, C6 | | $\overline{CS}=0, RS=1, R/\overline{W}=0/1$ | |
| — | SDA0 | — | SA0H/SRA0 | | |
| SA0L | | | | | |

2

Figure 5. Hardware Access and Direct Access Registers (cont)



| | | | |
|-------------------------|-------|-----------------------|---|
| Base Screen | | | |
| Raster Address 1 (RAR1) | | Register No.—rC8 | $\overline{CS}=0, RS=1, R/\overline{W}=0/1$ |
| — | LRA1 | — | FRA1 |
| Memory Width 1 (MWR1) | | Register No.—rCA | $\overline{CS}=0, RS=1, R/\overline{W}=0/1$ |
| CHR | — | MW1 | |
| Start Address 1 (SAR1) | | Register No.—rCC, rCE | $\overline{CS}=0, RS=1, R/\overline{W}=0/1$ |
| — | SDA1 | — | SA1H/SRA1 |
| SA1L | | | |
| Lower Screen | | | |
| Raster Address 2 (RAR2) | | Register No.—rD0 | $\overline{CS}=0, RS=1, R/\overline{W}=0/1$ |
| — | LRA2 | — | FRA2 |
| Memory Width 0 (MWR2) | | Register No.—rD2 | $\overline{CS}=0, RS=1, R/\overline{W}=0/1$ |
| CHR | — | MW2 | |
| Start Address 0 (SAR2) | | Register No.—rD4, D6 | $\overline{CS}=0, RS=1, R/\overline{W}=0/1$ |
| — | SDA2 | — | SA2H/SRA2 |
| SA2L | | | |
| Window Screen | | | |
| Raster Address 3 (RAR3) | | Register No.—rD8 | $\overline{CS}=0, RS=1, R/\overline{W}=0/1$ |
| — | LRA3 | — | FRA3 |
| Memory Width 0 (MWR3) | | Register No.—rDA | $\overline{CS}=0, RS=1, R/\overline{W}=0/1$ |
| CHR | — | MW3 | |
| Start Address 0 (SAR3) | | Register No.—rDC, rDE | $\overline{CS}=0, RS=1, R/\overline{W}=0/1$ |
| — | SDA3 | — | SA3H/SRA3 |
| SA3L | | | |
| Block Cursor 1 (BCUR1) | | Register No.—rE0, rE2 | $\overline{CS}=0, RS=1, R/\overline{W}=0/1$ |
| BCW1 | BCSR1 | — | BCER1 |
| BCA1 | | | |
| Block Cursor 2 (BCUR2) | | Register No.—rE4, rE6 | $\overline{CS}=0, RS=1, R/\overline{W}=0/1$ |
| BCW2 | BCSR2 | — | BCER2 |
| BCA2 | | | |

Figure 5. Hardware Access and Direct Access Registers (cont)



| | | | | | |
|-------------------------|------|-----------------------|---|---|-------|
| Cursor Definition (CDR) | | Register No.=rE8 | | $\overline{CS}=0, RS=1, R/\overline{W}=0/1$ | |
| CM | CON1 | COFF1 | — | CON2 | COFF2 |
| Zoom Factor (ZFR) | | Register No.=rEA | | $\overline{CS}=0, RS=1, R/\overline{W}=0/1$ | |
| HZF | VZF | — | | | |
| Lightpen Address (LPAR) | | Register No.=rEC, rEE | | $\overline{CS}=0, RS=1, R/\overline{W}=0/1$ | |
| — | | CHR | — | LPAH | |
| LPAL | | | | | |
| (Underfined) | | Register No.=rF0-rFE | | $\overline{CS}=0, RS=1, R/\overline{W}=0/1$ | |
| — | | | | | |

| | |
|---|--|
| ABT: Abort ACM: Access Mode ACP: Access Priority Address: Control Register number ARD: Area Detect ARE: Area Detect Interrupt Enable ATR: Attribute Control CDM: Command DMA Mode CED: Command End CEE: Command End Interrupt Enable CER: Command Error CRE: Command Error Interrupt Enable CSK: Cursor Display Skew DDM: Data DMA Mode DRC: DMA Request Control DSK: DISP Skew DSP: DISP Signal Control FE: FIFO Entry GAI: Graphic Address Increment Mode GBM: Graphic Bit Mode HC: Horizontal Cycle HDS: Horizontal Display Start HDW: Horizontal Display Width HSW: Horizontal Sync Width LPD: Light Pen Strobe Detect LPE: Light Pen Strobe Interrupt Enable M/S: Master/Slave PSE: Pause RAM: RAM Mode RC: Raster Count RFE: Read FIFO Full Interrupt Enable RFF: Read FIFO Full RFR: Read FIFO Ready RRE: Read FIFO Ready Interrupt Enable RSM: Raster Scan Mode SE0: Split Enable 0 SE1: Split Enable 1 SE2: Split Enable 2 SE3: Split Enable 3 STR: Start | VC: Vertical Cycle VDS: Vertical Display Start VSW: Vertical Sync Width WEE: Write FIFO Empty Interrupt Enable WFE: Write FIFO Empty WFR: Write FIFO Ready WRE: Write FIFO Ready Interrupt Enable WSS: Window Smooth Scroll SP0, SP1, SP2: Split Screen 0 Width, Split Screen 1 Width, Split Screen 2 Width BON1, BON2: Blink On 1, Blink On 2 BOFF1, BOFF2: Blink Off 1, Blink Off 2 HWS: Horizontal Window Start HWW: Horizontal Window Width VWS: Vertical Window Start VWW: Vertical Window Width CXS, CYS: Cursor X Start, Cursor Y Start CXE, CYE: Cursor X End, Cursor Y End FRA: First Raster Address LRA: Last Raster Address CHR: Character MW: Memory Width SDA: Start Dot Address SAH/SRA: Start Address High/Start Raster Address SAL: Start Address Low BCW1, BCW2: Block Cursor Width 1, Block Cursor Width 2 BCSR1, BCSR2: Block Cursor Start Raster 1, Block Cursor Start Raster 2 BCER1, BCER2: Block Cursor End Raster 1, Block Cursor End Raster 2 BCA1, BCA2: Block Cursor Address 1, Block Cursor Address 2 CM: Cursor Mode CON1, CON2: Cursor On 1, Cursor On 2 COFF1, COFF2: Cursor Off 1, Cursor Off 2 HZF, VZF: Horizontal Zoom Factor, Vertical Zoom Factor LPAH: Light Pen Address High LPAL: Light Pen Address Low |
|---|--|

2

Figure 5. Hardware Access and Direct Access Registers (cont)



Table 1. Drawing Parameter Registers

| Register No. | Read/Write | Name of Register | Abbr. | Data (H) | | | | | | | | Data (L) | | | | | | | |
|--------------|------------|---------------------|-------|----------|----|----|----|------|----|---|---|----------|---|---|---|------|---|---|---|
| | | | | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Pr00 | R/W | Color 0 | CL0 | | | | | | | | | CL0 | | | | | | | |
| Pr01 | R/W | Color 1 | CL1 | | | | | | | | | CL1 | | | | | | | |
| Pr02 | R/W | Color Comparison | CCMP | | | | | | | | | CCMP | | | | | | | |
| Pr03 | R/W | Edge Color | EDG | | | | | | | | | EDG | | | | | | | |
| Pr04 | R/W | Mask | MASK | | | | | | | | | MASK | | | | | | | |
| Pr05 | R/W | Pattern RAM Control | PRC | PPY | | | | PZCY | | | | PPX | | | | PZCX | | | |
| ↓ | | | | PSY | | | | — | | | | PSX | | | | — | | | |
| Pr07 | | | | PEY | | | | PZY | | | | PEX | | | | PZX | | | |
| Pr08 | R/W | Area Definition** | ADR | | | | | | | | | XMIN | | | | | | | |
| ↓ | | | | | | | | | | | | YMIN | | | | | | | |
| | | | | | | | | | | | | XMAX | | | | | | | |
| Pr08 | | | | | | | | | | | | YMAX | | | | | | | |
| Pr0C | R/W | Read Write Pointer | RWP | DN | | | | — | | | | RWPH | | | | | | | |
| Pr0D | | | | | | | | | | | | RWPL | | | | — | | | |
| Pr0E | — | Undefined | — | | | | | | | | | — | | | | | | | |
| Pr0F | | | | | | | | | | | | — | | | | | | | |
| Pr010 | R | Drawing Pointer | DP | DN | | | | — | | | | DPAH | | | | | | | |
| Pr11 | | | | | | | | | | | | DPAL | | | | DPD | | | |
| Pr12 | | | | | | | | | | | | X | | | | | | | |
| Pr13 | — | Undefined | — | | | | | | | | | Y | | | | | | | |
| Pr14 | | | | | | | | | | | | — | | | | | | | |
| Pr15 | | | | | | | | | | | | — | | | | | | | |

—: Always set to 0.
 **: Set two's complements for negative values of X and Y axis.

Drawing Parameter Register

- R: Register which can be read by Read Parameter Register Command (RPR)
- W: Register which can be written into by Write Parameter Register Command (WPR)
- : Access is not allowed
- CL0: Defines the color data used for the drawing when logical drawing data=0
- CL1: Defines the color data used for the drawing when logical drawing data=1
- CCMP: Defines the comparison color of the drawing operation
- PSX, PSY: Pattern Start Point
- PEX, PEY: Pattern End Point
- PPX, PPY: Pattern Scan Start Point
- PZX, PZY: Pattern Zoom
- PZCX, PZCY: Pattern Zoom Count
- XMIN, YMIN: Start point of Area definition
- XMAX, YMAX: End point of Area definition
- Dn: Screen Number
- RWPH: High-order 8 bits of Read Write Pointer Address
- RWPL: Low-order 12 bits of Read Write Pointer Address
- DPAH: High-order 8 bits of Drawing Pointer Address
- DPAL: Low-order 12 bits of Drawing Pointer Address
- DPD: Drawing Pointer Dot Address
- X, Y: Position indicated by Current Pointer on X-Y coordinate



Display Functions

Logical Display Screens

The ACRTC allows the frame buffer to be divided into four separate logical screens (table 2, figure 6).

In the simplest case, only the base screen parameters must be defined. Other screens may be selectively enabled, disabled, and blanked under software control.

The background screens (upper, base, and lower) split the screen into three horizontal partitions whose positions are fully programmable. A typical application might use the base screen for the bulk of user interac-

tion, using the lower screen for a "status line(s)" and the upper screen for "pulldown menus".

The window screen is unique, since the ACRTC usually gives it higher priority than the background screens. Thus, when the window, whose size and position is completely programmable, overlaps a background screen, the window is displayed. The exception is in the ACRTC superimposed mode, in which the window has the same priority as the background screens. In this mode, the window and background screens are "superimposed" on the display.



Table 2. Logical Screen

| Screen Number | Screen Name | Screen Group |
|---------------|---------------|--------------------|
| 0 | Upper screen | Background screens |
| 1 | Base screen | |
| 2 | Low screen | |
| 3 | Window screen | |

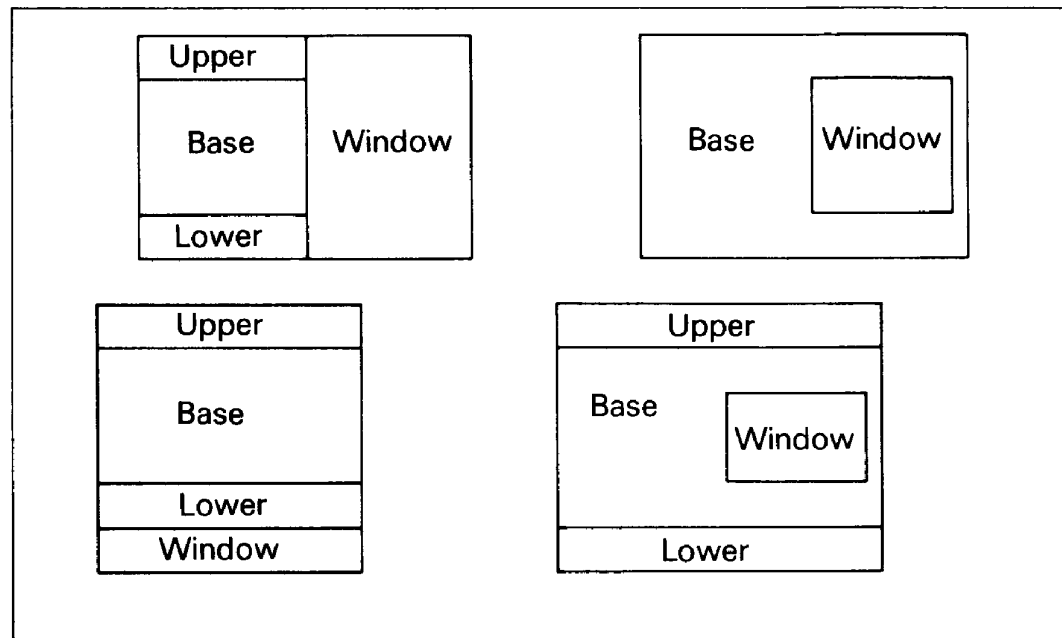


Figure 6. Screen Combination Examples



Frame Memory Setup

The ACRTC can have two independent frame memories, a 2-Mbyte frame buffer and a 128-kbyte refresh memory. The CHR output controls which memory is accessed.

Frame memory width is defined by setting up the memory width register (MWR). The horizontal width is independently defined by the horizontal display register (HDR). The memory area can therefore be specified bigger than the display area (figure 7).

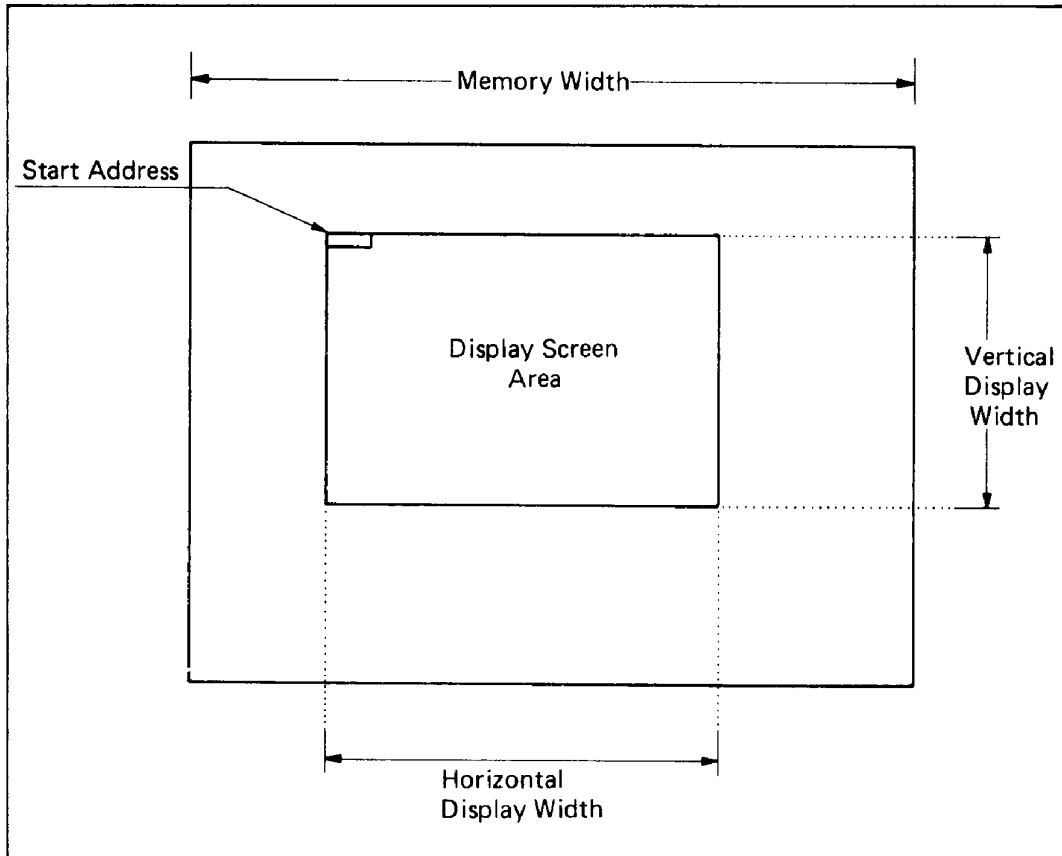


Figure 7. Frame Memory and Display Screen Area



Display Control

Figure 8 shows the relation between the frame memory and display screens. Each screen has its own memory width, vertical display width, and character/graphic attribution. These specifications are set by the control registers.

Horizontal display control registers are set in units of memory cycles. Vertical display control registers are set in units of rasters.

Note that display width of registers marked with an asterisk (*) in figure 9 is:

$$(\text{Display width}) = (\text{Register value}) + 1 \text{ memory cycles}$$

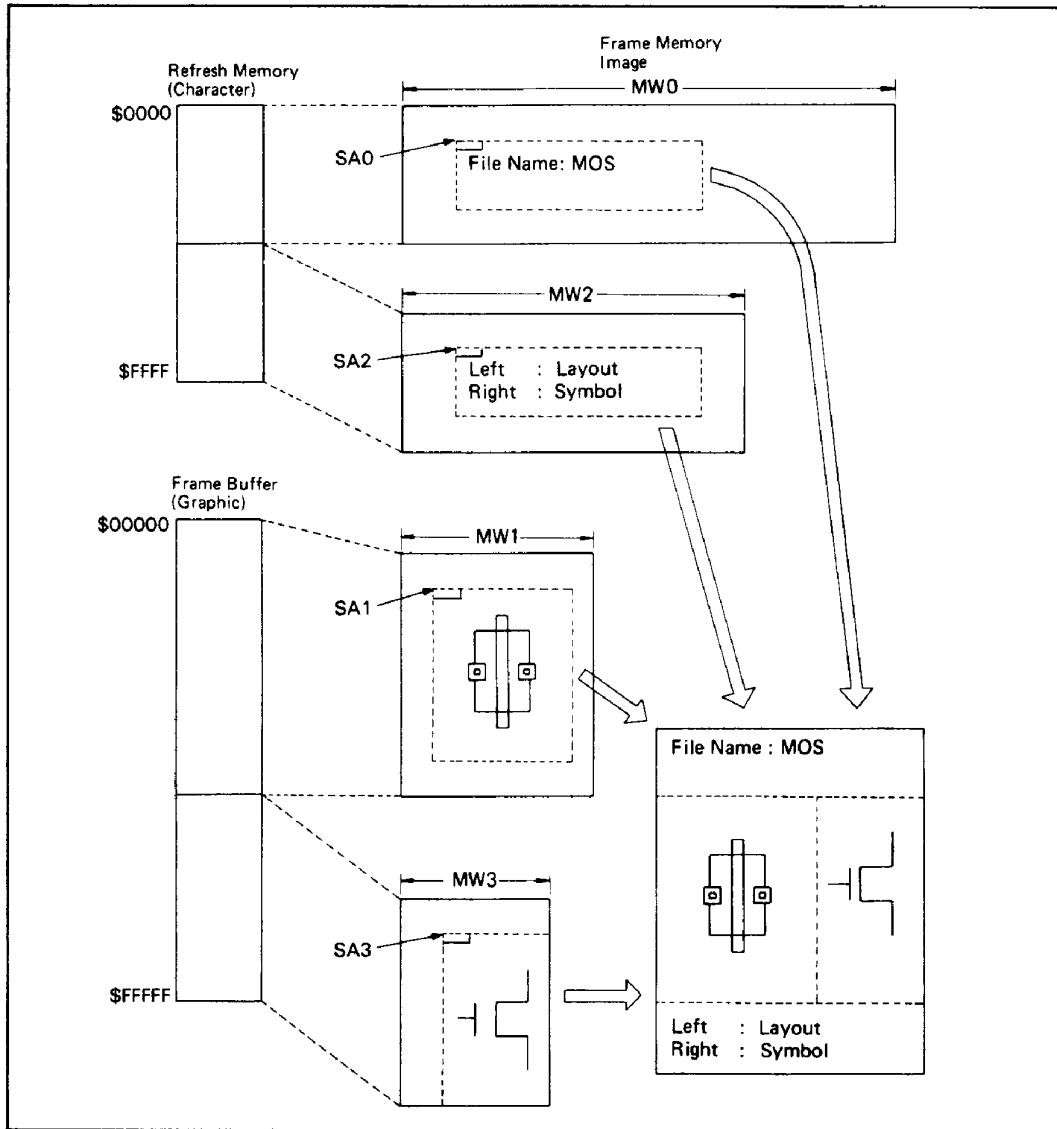


Figure 8. Frame Memory and Display Screens



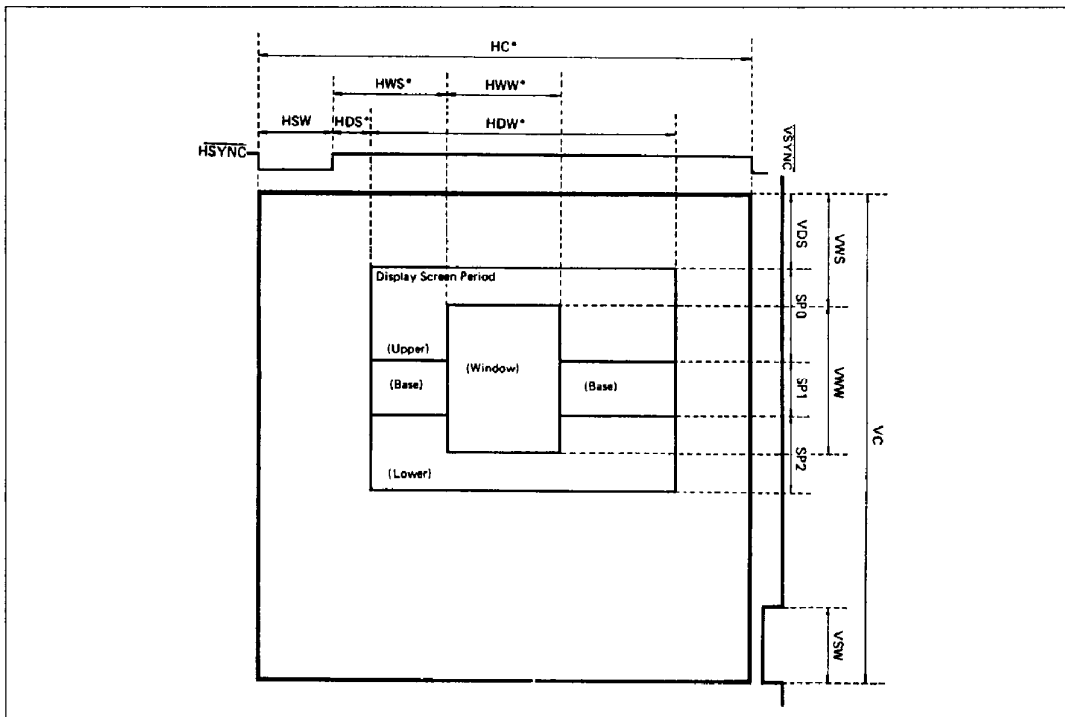


Figure 9. Display Screen Specification

Commands

The ACRTC has 38 commands classified into three groups (tables 3, 4):

- Register access
- Data transfer
- Graphic drawing

Five register access commands give the host MPU access to drawing processor drawing parameter registers and the pattern RAM.

Ten data transfer commands move data between the host system memory and the frame buffer, or within the frame buffer.

Twenty-three graphic drawing commands cause the ACRTC to draw. Parameters for these commands are specified using logical X-Y addressing.

All commands, parameters, and data are transferred via the ACRTC read and write FIFOs.

Assuming the ACRTC has been properly initialized, the MPU must perform two

steps to make the ACRTC draw:

1. First the MPU must specify drawing parameters that define the details associated with the drawing. For example, to draw a figure or paint an area, the MPU must specify the drawing or painting pattern by initializing the ACRTC pattern RAM and related pointers. If clipping or hitting control are desired, the MPU must specify the area to be monitored during drawing by initializing the area definition registers. Other drawing parameters include color, edge definition, etc.
2. After the drawing parameters have been specified, the MPU issues a drawing command and any required command parameters, such as the CRCL (circle) command with a radius parameter.

The ACRTC then performs the specified drawing operation by reading, modifying, and rewriting the contents of the frame buffer.



Downloaded from Elcodis.com electronic components distributor

Table 3. ACRTC Command Table

| Type | Mnemonic | Command Name | # (words) | Operation Cycles *1 |
|----------|----------|---------------------------|-----------|--|
| Register | ORG | Origin | 3 | 8 |
| Access | WPR | Write Parameter Register | 2 | 6 |
| Command | RPR | Read Parameter Register | 1 | 6 |
| | WPTN | Write Pattern RAM | n+2 | 4n+8 |
| | RPTN | Read Pattern RAM | 2 | 4n+10 |
| Data | DRD | DMA Read | 3 | $(4x+8)y+12[x \cdot y/8\uparrow] + (62 \sim 68)$ |
| Transfer | DWT | DMA Write | 3 | $(4x+8)y+16[x \cdot y/8\uparrow] + 34$ |
| Command | DMOD | DMA Modify | 3 | $(4x+8)y+16[x \cdot y/8\uparrow] + 34$ |
| | RD | Read | 1 | 12 |
| | WT | Write | 2 | 8 |
| | MOD | Modify | 2 | 8 |
| | CLR | Clear | 4 | $(2x+8)y+12$ |
| | SCLR | Selective Clear | 4 | $(4x+6)y+12$ |
| | CPY | Copy | 5 | $(6x+10)y+12$ |
| | SCPY | Selective Copy | 5 | $(6x+10)y+12$ |
| Graphic | AMOVE | Absolute Move | 3 | 56 |
| Drawing | RMOVE | Relative Move | 3 | 56 |
| Command | ALINE | Absolute Line | 3 | $P \cdot L + 18$ |
| | RLINE | Relative Line | 3 | $P \cdot L + 18$ |
| | ARCT | Absolute Rectangle | 3 | $2P(A+B)+54$ |
| | RRCT | Relative Rectangle | 3 | $2P(A+B)+54$ |
| | APLL | Absolute Polyline | 2n+2 | $\Sigma[P \cdot L + 16] + 8$ |
| | RPLL | Relative Polyline | 2n+2 | $\Sigma[P \cdot L + 16] + 8$ |
| | APLG | Absolute Polygon | 2n+2 | $\Sigma[P \cdot L + 16] + P \cdot Lo + 20$ |
| | RPLG | Relative Polygon | 2n+2 | $\Sigma[P \cdot L + 16] + P \cdot Lo + 20$ |
| | CRCL | Circle | 2 | $8d+66$ |
| | ELPS | Ellipse | 4 | $10d+90$ |
| | AARC | Absolute Arc | 5 | $8d+18$ |
| | RARC | Relative Arc | 5 | $8d+18$ |
| | AEARC | Absolute Ellipse Arc | 7 | $10d+96$ |
| | REARC | Relative Ellipse Arc | 7 | $10d+96$ |
| | AFRCT | Absolute Filled Rectangle | 3 | $(P \cdot A + 8)B + 18$ |
| | RFRCT | Relative Filled Rectangle | 3 | $(P \cdot A + 8)B + 18$ |
| | PAINT | Paint | 1 | $(18A+102)B-58$ *2 |
| | DOT | Dot | 1 | 8 |
| | PTN | Pattern | 2 | $(P \cdot A + 10)B + 20$ |
| | AGCPY | Absolute Graphic Copy | 5 | $((P+2)A+10)B+70$ |
| | RGCPY | Relative Graphic Copy | 5 | $((P+2)A+10)B+70$ |

- Notes: 1. 2CLK cycles.
 2. Applies to rectangular figures. Time varies for other shapes.
 3. Abbreviations
- n: Number of read/write data words
 - x: Number of words in X direction
 - y: Number of words in Y direction
 - †: Round up
 - P: Operation cycles—P=4 cycles when OPM=000-011;
P=6 cycles when OPM=100-111
 - L, Lo: Number of dots in a straight line
 - d: Total number of dots
 - A: Number of dots in main scan direction
 - B: Number of dots in sub scan direction



Table 4. Command Operation Codes and Parameters

| Type | Mnemonic | Operation Code | Parameter | |
|----------|-------------------------|--|---|--|
| Register | ORG | 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 DPH DPL | | |
| Access | WPR | 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 D | | |
| Command | RPR | 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 RN | | |
| | WPTN | 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 PRA n D ₁ , ..., D _n | | |
| | RPTN | 0 0 0 1 1 1 0 0 0 0 0 0 0 0 0 0 0 PRA n | | |
| Data | DRD | 0 0 1 0 0 1 0 0 0 0 0 0 0 0 0 0 0 AX AY | | |
| Transfer | DWT | 0 0 1 0 1 0 0 0 0 0 0 0 0 0 0 0 0 AX AY | | |
| Command | DMOD | 0 0 1 0 1 1 0 0 0 0 0 0 0 0 0 0 0 MM AX AY | | |
| | RD | 0 1 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 | | |
| | WT | 0 1 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 D | | |
| | MOD | 0 1 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 MM D | | |
| | CLR | 0 1 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 D AX AY | | |
| | SCLR | 0 1 0 1 1 1 0 0 0 0 0 0 0 0 0 0 0 MM D AX AY | | |
| | CPY | 0 1 1 0 S DSD 0 0 0 0 0 0 0 0 0 0 0 SAH SAL AX AY | | |
| | SCPY | 0 1 1 1 S DSD 0 0 0 0 0 0 0 0 0 0 0 MM SAH SAL AX AY | | |
| | Graphic Drawing Command | AMOVE | 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 X Y | |
| | | RMOVE | 1 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 dX dY | |
| | | ALINE | 1 0 0 0 1 0 0 0 AREA COL OPM X Y | |
| | | RLINE | 1 0 0 0 1 1 0 0 AREA COL OPM dX dY | |
| | | ARCT | 1 0 0 1 0 0 0 0 AREA COL OPM X Y | |
| RRCT | | 1 0 0 1 0 1 0 0 AREA COL OPM dX dY | | |
| APLL | | 1 0 0 1 1 0 0 0 AREA COL OPM n X ₁ , Y ₁ , ... X _n , Y _n | | |
| RPLL | | 1 0 0 1 1 1 0 0 AREA COL OPM n dX ₁ , dY ₁ , ... dX _n , dY _n | | |
| APLG | | 1 0 1 0 0 0 0 0 AREA COL OPM n X ₁ , Y ₁ , ... X _n , Y _n | | |
| RPLG | | 1 0 1 0 0 1 0 0 AREA COL OPM n dX ₁ , dY ₁ , ... dX _n , dY _n | | |
| CRCL | | 1 0 1 0 1 0 0 C AREA COL OPM r | | |
| ELPS | | 1 0 1 0 1 1 0 C AREA COL OPM a b DX | | |
| AARC | | 1 0 1 1 0 0 0 C AREA COL OPM X _c Y _c X _e Y _e | | |
| RARC | | 1 0 1 1 0 1 0 C AREA COL OPM dX _c dY _c dX _e dY _e | | |
| AEARC | | 1 0 1 1 1 0 0 C AREA COL OPM a b X _c Y _c X _e Y _e | | |
| REARC | | 1 0 1 1 1 1 0 C AREA COL OPM a b dX _c dY _c dX _e dY _e | | |
| AFRCT | | 1 1 0 0 0 0 0 0 AREA COL OPM X Y | | |
| RFRCT | | 1 1 0 0 0 1 0 0 AREA COL OPM dX dY | | |
| PAINT | | 1 1 0 0 1 0 0 E AREA 0 0 0 0 | | |
| DOT | | 1 1 0 0 1 1 0 0 AREA COL OPM | | |
| PTN | | 1 1 0 1 SL SD AREA COL OPM SZ | | |
| AGCPY | | 1 1 1 0 S DSD AREA 0 0 OPM X _s Y _s DX DY | | |
| RGCPY | | 1 1 1 1 S DSD AREA 0 0 OPM dX _s dY _s DX DY | | |

- Notes: 1. Register access commands abbreviations
 RN: Drawing parameter register number (\$0-\$13)
 PRA: Pattern RAM read/write operation starting address (\$0-\$F)
 DPH: Drawing pointer register high word (figure 29)
 DPL: Drawing pointer register low word (figure 29)
 DPAH: Higher 8 bits of drawing pointer address
 DPAL: Lower 12 bits of drawing pointer address
 DPD: Dot position in memory address
2. Data transfer commands abbreviations
 MM: Modify mode
 S: Source scan direction (figure 30)
 DSD: Destination scan direction (figure 31)
 AX: Number of words in X direction—1
 AY: Number of words in Y direction—1
 D: Write data
 SAH: Source start address high word (figure 32)
 SAL: Source start address low word
3. Graphic drawing commands abbreviations
 AREA: Area mode
 COL: Color mode
 OPM: Operation mode
 C: Circle drawing direction—C = 0 for counterclockwise;
 C = 1 for clockwise



E: Edge color definition—E = 0, edge color is data in edge color register;
E = 1, edge color is any color except data in color register

SL: Slant (figure 33)
SD: Scan direction (figure 33)
S: Source scan direction (figure 30)
DSD: Destination scan direction (figure 31)

4. Parameter abbreviations

X, X1, ..., Xn: Absolute X address from origin point
Y, Y1, ..., Yn: Absolute Y address from origin point
dX: Relative X address from current pointer
dY: Relative Y address from current pointer
n: Number of nodes
dX1, ..., dXn: Relative X address from each node
dY1, ..., dYn: Relative Y address from each node
r: Number of dots on radius
a, b: ratio of dX squared to dY squared in ellipse— $a:b = dX^2:dY^2$
DX: X-direction dot number
DY: Y-direction dot number
Xc: Absolute X address of center point of arc/ellipse
Yc: Absolute Y address of center point of arc/ellipse
dXc: Relative X address from current point to center point of arc/ellipse
dYc: Relative Y address from current point to center point of arc/ellipse
Xe: Absolute X address of end point of arc/ellipse
Ye: Absolute Y address of end point of arc/ellipse
dXe: Relative X address from current point to end point of arc/ellipse
dYe: Relative Y address from current point to end point of arc/ellipse
Xs: Absolute X address of start point of arc/ellipse
Ys: Absolute Y address of start point of arc/ellipse
dXs: Relative X address from current point to start point of arc/ellipse
dYs: Relative Y address from current point to start point of arc/ellipse

2

Program Transfer

For program transfer, the MPU specifies the FIFO entry address and then writes commands/parameters to the write FIFO under program control (RS = high, R/W, CS = low). The MPU writes are normally synchronized with FIFO status by software polling or interrupt.

Software Polling: WFR, WFE interrupts are disabled.

1. MPU program checks the SR (status register) for write FIFO ready flag (WFR) = 1, then writes 1 command/parameter word.
2. MPU program checks the SR (status register) for write FIFO empty flag (WFE) = 1, then writes 1 to 8 command/parameter words.

Interrupt Driven: WFR, WFE interrupts are enabled.

1. MPU WFR interrupt service routine writes 1 command/parameter word.
2. MPU WFE interrupt service routine writes 1 to 8 command/parameter words.

Register Access Commands: When writing register access commands to an in-

itally empty write FIFO, the MPU does not have to synchronize to write FIFO status. The ACRTC can fetch and execute these commands faster than the MPU can issue them.

Command DMA Transfer

Commands and parameters can be transferred from the MPU system memory by an external DMAC. The MPU initiates and terminates command DMA transfer under software control (CDM bit of CCR). Command DMA transfer can also be terminated by asserting the ACRTC DONE signal. DONE is an input in command DMA transfer mode.

In command DMA transfer mode, the ACRTC issues cycle stealing DMA requests to the DMAC when the write FIFO is ready. The DMA data is automatically sent from system memory to the ACRTC write FIFO regardless of the contents of the address register.

Make sure that the write FIFO is empty and all previous commands are terminated before starting the command DMA transfer.

Data DMA transfer cannot be executed in command DMA transfer mode.



Table 7. Graphic Drawing Commands

| Command | Function |
|---------|---|
| AMOVE | Move the current pointer (CP) to an absolute logical pixel X-Y address |
| RMOVE | Move the current pointer (CP) to a relative logical pixel X-Y address |
| ALINE | Draw a straight line from the current pointer (CP) to a command-specified end point in absolute coordinates |
| RLINE | Draw a straight line from the current pointer (CP) to a command-specified end point in relative coordinates |
| ARCT | Draw a rectangle defined by the current pointer (CP) and a command-specified diagonal point in absolute coordinates |
| RRCT | Draw a rectangle defined by the current pointer (CP) and a command-specified diagonal point in relative coordinates |
| APLL | Draw a polyline (multiple contiguous segments) from the current pointer (CP) through command-specified points in absolute coordinates |
| RPLL | Draw a polyline (multiple contiguous segments) from the current pointer (CP) through command-specified points in relative coordinates |
| APLG | Draw a polygon which connects the current pointer (CP) and command-specified points in absolute coordinates |
| RPLG | Draw a polygon which connects the current pointer (CP) and command-specified points in relative coordinates |
| CRCL | Draw a circle of radius R placing the current pointer (CP) at the center |
| ELPS | Draw an ellipse whose shape is specified by command parameters, placing the current pointer (CP) at the center |
| AARC | Draw an arc by using the current pointer (CP) as a start point with an end point and a center point in absolute coordinates |
| RARC | Draw an arc by using the current pointer (CP) as a start point with an end point and a center point in relative coordinates |
| AEARC | Draw an ellipse arc by using the current pointer (CP) as a start point with an end point and a center point in absolute coordinates |
| REARC | Draw an ellipse arc by using the current pointer (CP) as a start point with an end point and a center point in relative coordinates |
| AFRCT | Paint a rectangular area specified by the current pointer (CP) and command parameters (absolute coordinates) according to a figure pattern stored in the pattern RAM (tiling) |
| RFRCT | Paint a rectangular area specified by the current point (CP) and command parameters (relative coordinates) according to a figure pattern stored in the pattern RAM (tiling) |
| PAINT | Paint a closed area surrounded by edge color using a figure pattern stored in the pattern RAM (tiling) |
| DOT | Mark a dot on the coordinates indicated by the current pointer (CP) |
| PTN | Draw a graphic pattern defined in the pattern RAM onto a rectangular area specified by the current point (CP) and by the pattern size (rotation angle: 45°) |
| AGCPY | Copy a rectangular area specified by the absolute coordinates to the address specified by the current pointer (CP) (rotation angle: 90°/mirror reflection) |
| RGCPY | Copy a rectangular area specified by the relative coordinates to the address specified by the current pointer (CP) (rotation angle: 90°/mirror reflection) |

Notes on System Design

Power-On Sequence

The conditions in figure 10 must be satisfied at power-on.

Output Waveform

If excessive ringing (figure 11) occurs on CRT data buses, (MAD_0 - MAD_{15} , MA_{16}/RA_0 - MA_{19}/RA_4), damping resistors may be required as shown in figure 12.

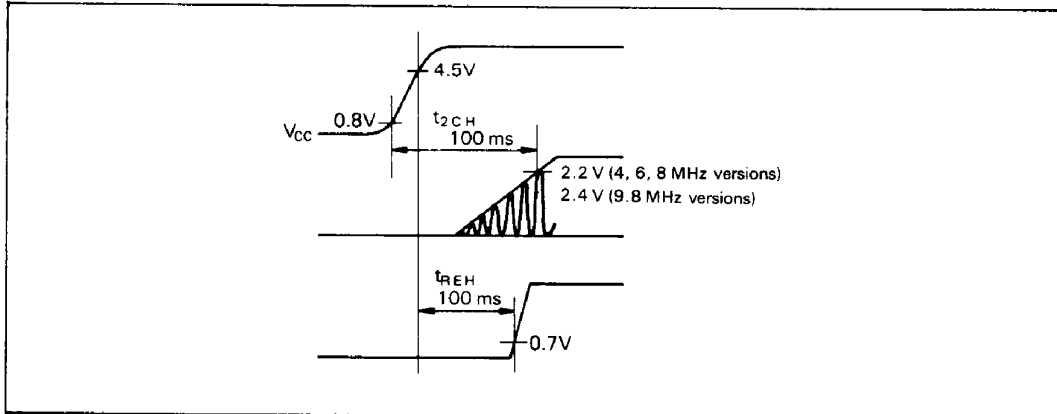
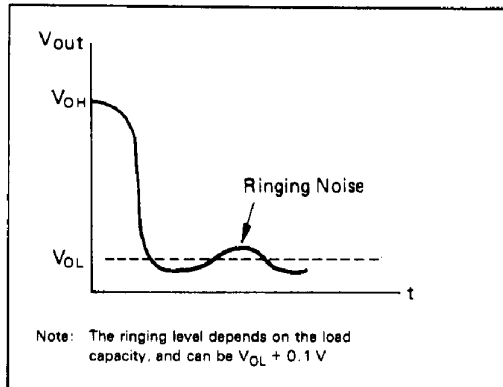


Figure 10. Power-On Sequence



Note: The ringing level depends on the load capacity, and can be $V_{OL} + 0.1V$

Figure 11. Ringing Noise

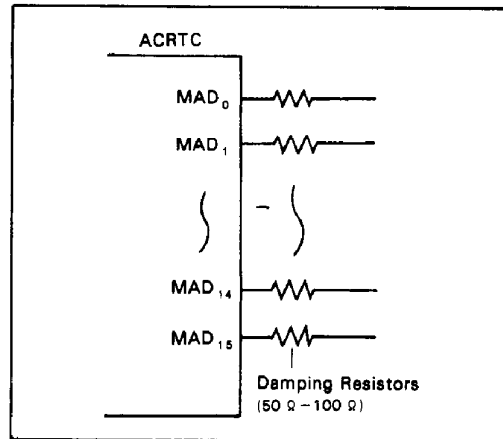


Figure 12. Damping Resistors



Power Supply Circuit

When laying out the V_{CC} and V_{SS} traces on

the circuit board, locate capacitors as close as possible to each power supply pin (figures 13, 14, 15).

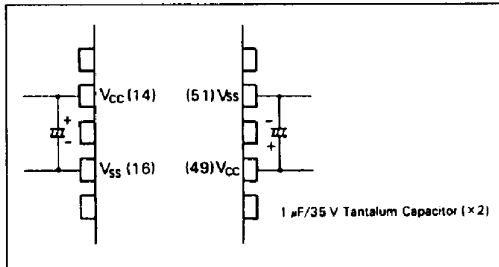


Figure 13. Power Supply Circuit Example, 64-Pin DIP

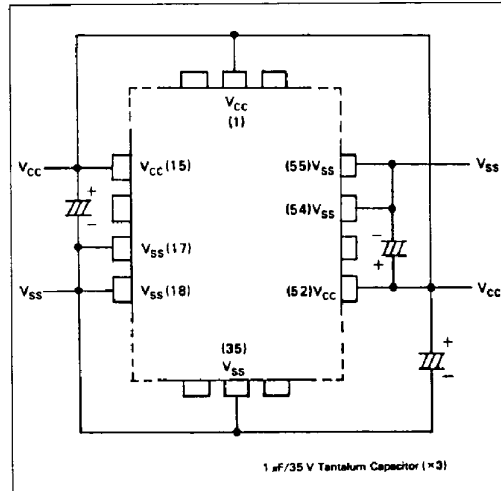


Figure 14. Power Supply Circuit Example, 68-Pin PLCC

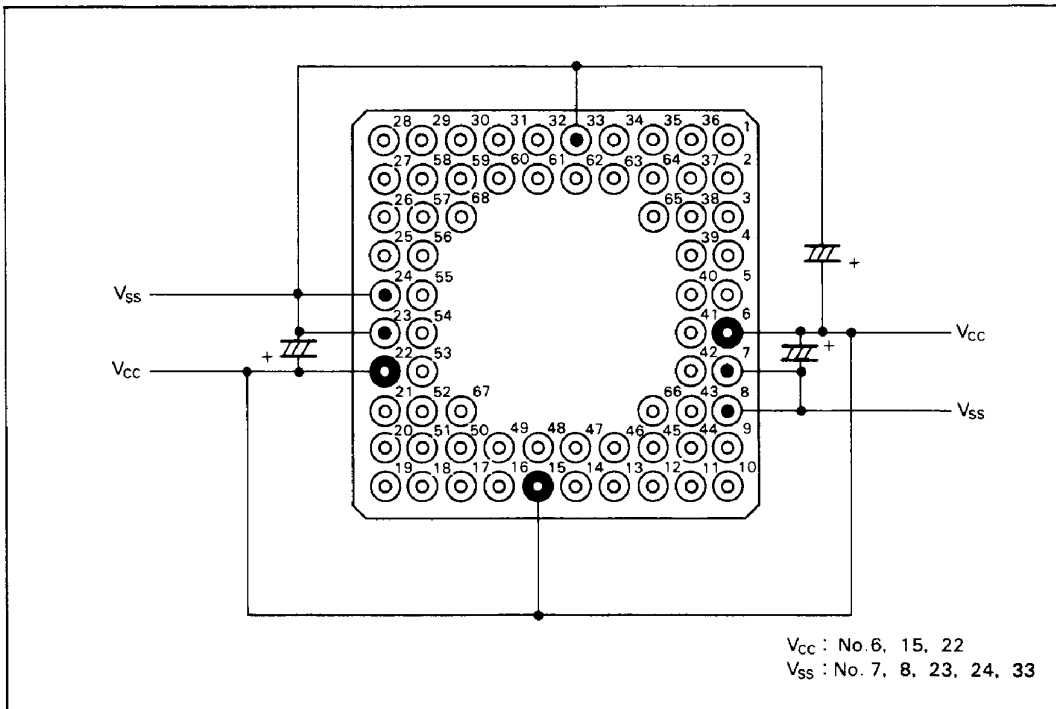


Figure 15. Power Supply Circuit Example 68-Pin PGA

2



HD63484

Absolute Maximum Ratings

| Item | Symbol | Rating | Unit |
|--------------------------------|-------------------|------------------------|------|
| Supply Voltage | V_{CC} (Note 1) | -0.3 to +7.0 | V |
| Input Voltage | V_{in} (Note 1) | -0.3 to $V_{CC} + 0.3$ | V |
| Allowable Output Current | $ I_O $ (Note 2) | 5 | mA |
| Total Allowable Output Current | $ ΣI_O $ (Note 3) | 120 | mA |
| Operating Temperature | T_{opr} | 0 to +70 | °C |
| Storage Temperature | T_{stg} | -55 to +150 | °C |

- Notes: 1. Referenced to $V_{SS} = 0$ V.
2. The maximum current that may be drawn from, or flow out of, one output, or one common input/output terminal.
3. The total sum of currents that may be drawn from, or flow out of, all output or common input/output terminals.
4. Using an LSI beyond its maximum rating may result in its permanent destruction. LSIs should usually be used under recommended operating conditions. Exceeding any of these conditions may adversely affect reliability.

Recommended Operating Conditions

| Item | Symbol | Min | Typ | Max | Unit |
|---|-----------------|------|-----|----------|------|
| Supply Voltage | V_{CC} (Note) | 4.75 | 5.0 | 5.25 | V |
| Input Low Voltage | V_{IL} (Note) | 0 | | 0.7 | V |
| Input High Voltage (4, 6, 8 MHz versions) | V_{IH} (Note) | 2.2 | | V_{CC} | V |
| (9.8 MHz version) | V_{IH} (Note) | 2.4 | | V_{CC} | V |
| Operating Temperature | T_{opr} | 0 | 25 | 75 | °C |

Note: Referenced to $V_{SS} = 0$ V



Electrical Characteristics**DC Characteristics**(V_{CC}=5.0V ± 5%, V_{SS}=0V, Ta=0°C to +70°C unless otherwise noted)

| Item | | Symbol | Min | Max | Unit | Test Condition |
|----------------------------|---|------------------|----------------------|-----------------|------|--|
| Input High Level Voltage | All inputs | V _{IH} | 2.2 | V _{CC} | V | 4, 6, 8 MHz versions |
| | | | 2.4 | V _{CC} | V | 9.8 MHz ★ version |
| Input Low Level Voltage | All inputs | V _{IL} | -0.3 | 0.7 | V | |
| Input Leak Current | R/W, CS, RS, RES, DACK, 2CLK, LPSTB | I _{in} | -2.5 | 2.5 | μA | V _{in} =-0.4 to V _{CC} |
| Hi-Z Input Current | D ₀ -D ₁₅ , MAD ₀ -MAD ₁₅ , EXSYNC | I _{TSI} | -10 | 10 | μA | V _{in} =-0.4 to V _{CC} |
| Output High Level Voltage | D ₀ -D ₁₅ , MAD ₀ -MAD ₁₅ , CUD1, CUD2, DREQ, DTACK, HSYNC, VSYNC, EXSYNC, MRD, DRAW, AS, DISP1, DISP2, CHR, MCYC, RA ₄ , MA ₁₆ /RA ₀ -MA ₁₅ /RA ₃ | V _{OH} | V _{CC} -1.0 | | μA | I _{OH} =-400μA |
| Output Low Level Voltage | D ₀ -D ₁₅ , MAD ₀ -MAD ₁₅ , CUD1, CUD2, DREQ, DTACK, HSYNC, VSYNC, EXSYNC, MRD, DRAW, AS, DISP1, DISP2, CHR, MCYC, RA ₄ , MA ₁₆ /RA ₀ -MA ₁₅ /RA ₃ | V _{OL} | | 0.5 | V | I _{OL} =-2.2mA |
| | | | | | 0.5 | V |
| Output Leak Current (Hi-Z) | IRQ, DONE | I _{LOD} | | 10 | μA | V _{OH} =V _{CC} |
| Input Capacitance | D ₀ -D ₁₅ , MAD ₀ -MAD ₁₅ , EXSYNC, R/W, CS, RS, RES, DACK, 2CLK, LPSTB | C _{in} | | 17 | pF | V _{in} =0 V, Ta=25°C, f=1.0 MHz |
| Output Capacitance | IRQ, DONE | C _{out} | | 15 | pF | V _{in} =0 V, Ta=25°C, f=1.0 MHz |
| Current Consumption | | I _{CC} | | 60 | mA | 4 MHz version |
| | | | | 80 | mA | 6 MHz version |
| | | | | 100 | mA | 8 MHz version |
| | | | | 120 | mA | 9.8 MHz version |

2



HD63484

AC Characteristics ($V_{CC}=5.0V \pm 5\%$, $V_{SS}=0V$, $T_a=0$ to $+70^\circ C$ unless otherwise noted)

Clock Timing

| No. | Item | Symbol | 4 MHz Version | | 6 MHz Version | | 8 MHz Version | | 9.8 MHz Version | | Unit | Reference Figure Number |
|-----|------------------------------|-------------------|---------------|------|---------------|------|---------------|------|-----------------|------|------|-------------------------|
| | | | Min | Max | Min | Max | Min | Max | Min | Max | | |
| | Operation Frequency of 2CLK | f | 1 | 4 | 1 | 6 | 1 | 8 | 1 | 9.8 | MHz | 17 |
| 1 | Clock Cycle Time | t _{cy} | 250 | 1000 | 167 | 1000 | 125 | 1000 | 102 | 1000 | ns | |
| 2 | Clock High Level Pulse Width | t _{PWCH} | 115 | 500 | 75 | 500 | 55 | 500 | 46 | 500 | ns | |
| 3 | Clock Low Level Pulse Width | t _{PWCL} | 115 | 500 | 75 | 500 | 55 | 500 | 46 | 500 | ns | |
| 4 | Clock Rise Time | t _{cr} | — | 10 | — | 10 | — | 10 | — | 5 | ns | |
| 5 | Clock Fall Time | t _{cf} | — | 10 | — | 10 | — | 10 | — | 5 | ns | |

MPU Read/Write Cycle Timing

| No. | Item | Symbol | 4 MHz Version | | 6 MHz Version | | 8 MHz Version | | 9.8 MHz Version | | Unit | Reference Figure Number |
|-----|---|--------------------|---------------|-----|---------------|-----|---------------|-----|-----------------|-----|------|-------------------------|
| | | | Min | Max | Min | Max | Min | Max | Min | Max | | |
| 6 | R/W Setup Time | t _{RWS} | 70 | — | 60 | — | 50 | — | 50 | — | ns | 18—20 |
| 7 | R/W Hold Time | t _{RWH} | 0 | — | 0 | — | 0 | — | 0 | — | ns | |
| 8 | RS Setup Time | t _{RSS} | 70 | — | 60 | — | 50 | — | 50 | — | ns | |
| 9 | RS Hold Time | t _{RSH} | 0 | — | 0 | — | 0 | — | 0 | — | ns | |
| 10 | \overline{CS} Setup Time | t _{CSS} | 50 | — | 40 | — | 40 | — | 40 | — | ns | 18, 19 |
| 11 | \overline{CS} High Level Width | t _{WCSSH} | 80 | — | 70 | — | 60 | — | 60 | — | ns | |
| 12 | | | | | | | | | | | | |
| 13 | Read Wait Time | t _{RWAI} | 0 | — | 0 | — | 0 | — | 0 | — | ns | 18, 20 |
| 14 | Read Data Access Time | t _{RDAC} | — | 120 | — | 100 | — | 80 | — | 80 | ns | |
| 15 | Read Data Hold Time | t _{RDH} | 10 | — | 10 | — | 10 | — | 10 | — | ns | |
| 16 | Read Data Turn Off Time | t _{RDZ} | — | 60 | — | 60 | — | 60 | — | 60 | ns | |
| 17 | \overline{DTACK} Delay Time (Z to L) | t _{DTKZL} | — | 90 | — | 80 | — | 70 | — | 70 | ns | 18—20 |
| 18 | \overline{DTACK} Delay Time (D to L) | t _{DTKDL} | 0 | — | 0 | — | 0 | — | 0 | — | ns | 18, 20 |
| 19 | \overline{DTACK} Release Time (L to H) | t _{DTKLH} | — | 100 | — | 90 | — | 80 | — | 80 | ns | 18—20 |
| 20 | \overline{DTACK} Turn Off Time (H to Z) | t _{DTKZ} | — | 100 | — | 100 | — | 100 | — | 100 | ns | |
| 21 | Data Bus 3-State Recovery Time 1 | t _{DBRT1} | 0 | — | 0 | — | 0 | — | 0 | — | ns | 18, 20 |
| 22 | Write Wait Time | t _{WWAI} | 0 | — | 0 | — | 0 | — | 0 | — | ns | 19,20 |
| 23 | Write Data Setup Time | t _{WDS} | 80 | — | 60 | — | 40 | — | 40 | — | ns | |
| 24 | Write Data Hold Time | t _{WDH} | 10 | — | 10 | — | 10 | — | 10 | — | ns | |



DMA Read/Write Cycle Timing

| No. | Item | Symbol | 4 MHz Version | | 6 MHz Version | | 8 MHz Version | | 9.8 MHz Version | | Unit | Reference Figure Number |
|-----|------------------------------------|--------|---------------|-----|---------------|-----|---------------|-----|-----------------|-----|------|-------------------------|
| | | | Min | Max | Min | Max | Min | Max | Min | Max | | |
| 25 | DREQ Delay Time 1 | tDRQD1 | — | 150 | — | 130 | — | 110 | — | 110 | ns | 21—24 |
| 26 | DREQ Delay Time 2 | tDRQD2 | — | 90 | — | 80 | — | 70 | — | 70 | ns | |
| 27 | DMA R/W Setup Time | tDMRWS | 70 | — | 60 | — | 50 | — | 50 | — | ns | |
| 28 | DMA R/W Hold Time | tDMRWH | 0 | — | 0 | — | 0 | — | 0 | — | ns | |
| 29 | DACK Setup Time | tDAKS | 50 | — | 40 | — | 40 | — | 40 | — | ns | |
| 30 | DACK High Level Width | tWDAKH | 80 | — | 70 | — | 60 | — | 60 | — | ns | |
| 31 | | | | | | | | | | | | |
| 32 | DMA Read Wait Time | tDRW | 0 | — | 0 | — | 0 | — | 0 | — | ns | 21, 22 |
| 33 | DMA Read Data Access Time | tDRDAC | — | 120 | — | 100 | — | 80 | — | 80 | ns | |
| 34 | DMA Read Data Hold Time | tDRDH | 10 | — | 10 | — | 10 | — | 10 | — | ns | |
| 35 | DMA Read Data Turn Off Time | tDRDZ | — | 60 | — | 60 | — | 60 | — | 60 | ns | |
| 36 | DMA DTACK Delay Time (Z to L) | tDDTZL | — | 90 | — | 80 | — | 70 | — | 70 | ns | 21—24 |
| 37 | DMA DTACK Delay Time (D to L) | tDDTDL | 0 | — | 0 | — | 0 | — | 0 | — | ns | 21, 22 |
| 38 | DMA DTACK Release Time (L to H) | tDDTLH | — | 100 | — | 90 | — | 80 | — | 80 | ns | 21—24 |
| 39 | DMA DTACK Turn Off Time (H to Z) | tDDTHZ | — | 100 | — | 100 | — | 100 | — | 100 | ns | |
| 40 | DONE Output Delay Time | tDND | — | 90 | — | 80 | — | 70 | — | 70 | ns | |
| 41 | DONE Output Turn Off Time (L to Z) | tDNLZ | — | 100 | — | 90 | — | 80 | — | 80 | ns | |
| 42 | Data Bus 3-State Recovery Time 2 | tDBRT2 | 0 | — | 0 | — | 0 | — | 0 | — | ns | 21, 22 |
| 43 | DONE Input Pulse Width | tDNPW | 2 | — | 2 | — | 2 | — | 2 | — | tcyc | 23, 24 |
| 44 | DMA Write Wait Time | tDWW | 0 | — | 0 | — | 0 | — | 0 | — | ns | |
| 45 | DMA Write Data Setup Time | tDWDS | 80 | — | 60 | — | 40 | — | 40 | — | ns | |
| 46 | DMA Write Data Hold Time | tDWDH | 10 | — | 10 | — | 10 | — | 10 | — | ns | |
| 47 | | | | | | | | | | | | |



Frame Memory Read/Write Cycle Timing

| No. | Item | Symbol | 4 MHz Version | | 6 MHz Version | | 8 MHz Version | | 9.8 MHz Version | | Unit | Reference Figure Number |
|-----|---------------------------------------|--------|---------------|-----|---------------|-----|---------------|-----|-----------------|-----|------|-------------------------|
| | | | Min | Max | Min | Max | Min | Max | Min | Max | | |
| 48 | AS "Low" Level Pulse Width | tPWASL | 80 | — | 40 | — | 25 | — | 20 | — | ns | 25—28 |
| 49 | Memory Address Hold Time 2 | tMAH2 | 10 | — | 10 | — | 10 | — | 5 | — | ns | |
| 50 | AS Delay Time 1 | tASD1 | — | 90 | — | 75 | — | 60 | — | 50 | ns | |
| 51 | AS Delay Time 2 | tASD2 | 5 | 90 | 5 | 75 | 5 | 65 | 5 | 40 | ns | |
| 52 | Memory Address Delay Time | tMAD | 15 | 95 | 15 | 80 | 15 | 70 | 10 | 50 | ns | |
| 53 | Memory Address Hold Time 1 | tMAH1 | 25 | — | 25 | — | 25 | — | 15 | — | ns | |
| 54 | Memory Address Turn Off Time (A to Z) | tMAAZ | — | 50 | — | 50 | — | 50 | — | 35 | ns | |
| 55 | Memory Read Data Setup Time | tMRDS | 60 | — | 50 | — | 35 | — | 30 | — | ns | 26 |
| 56 | Memory Read Data Hold Time | tMRDH | 10 | — | 10 | — | 10 | — | 0 | — | ns | 25—28 |
| 57 | MA/RA Delay Time | tMARAD | — | 100 | — | 90 | — | 80 | — | 60 | ns | |
| 58 | MA/RA Hold Time | tMARAH | 10 | — | 10 | — | 10 | — | 5 | — | ns | 25—27 |
| 59 | MCYC Delay Time | tMCYCD | 5 | 60 | 5 | 50 | 5 | 50 | 5 | 40 | ns | 25—29 |
| 60 | MRD Delay Time | tMRDD | — | 90 | — | 80 | — | 70 | — | 50 | ns | 24—28 |
| 61 | MRD Hold Time | tMRH | 10 | — | 10 | — | 10 | — | 5 | — | ns | |
| 62 | DRAW Delay Time | tDRWD | — | 90 | — | 80 | — | 70 | — | 50 | ns | |
| 63 | DRAW Hold Time | tDRWH | 10 | — | 10 | — | 10 | — | 5 | — | ns | 27 |
| 64 | Memory Write Data Delay Time | tMWDD | — | 90 | — | 80 | — | 70 | — | 50 | ns | |
| 65 | Memory Write Data Hold Time | tMWDH | 10 | — | 10 | — | 10 | — | 5 | — | ns | 25—28 |
| 110 | Memory Address Setup Time 1 | tMAS1 | 10 | — | 10 | — | 10 | — | 10 | — | ns | |
| 112 | Memory Address Setup Time 2 | tMAS2 | 10 | — | 10 | — | 10 | — | 10 | — | ns | |

Notes: 1. Characteristic No.52 is independent of the 2CLK operation frequency (f) and timing of No.51 and No.110.
 2. New characteristics No.50 and No.52 shown above are applicable only to lot numbers 5M*, 6*, 7**, and greater (* means don't care).

For the other lot numbers, applicable characteristics are as follows:

| No. | Symbol | 4 MHz Version | | 6 MHz Version | | 8 MHz Version | | Unit |
|-----|--------|---------------|-----|---------------|-----|---------------|-----|------|
| | | Min | Max | Min | Max | Min | Max | |
| 50 | tASD1 | — | 90 | — | 75 | — | 65 | ns |
| 52 | tMAD | — | 95 | — | 80 | — | 70 | ns |



Display Control Signal Output Timing

| No. | Item | Symbol | 4 MHz Version | | 6 MHz Version | | 8 MHz Version | | 9.8 MHz Version | | Unit | Reference Figure Number |
|-----|--------------------------|--------|---------------|-----|---------------|-----|---------------|-----|-----------------|-----|------|-------------------------|
| | | | Min | Max | Min | Max | Min | Max | Min | Max | | |
| 67 | HSYNC Delay Time | tHSD | — | 90 | — | 80 | — | 70 | — | 50 | ns | 29 |
| 68 | VSYNC Delay Time | tVSD | — | 90 | — | 80 | — | 70 | — | 50 | ns | |
| 69 | DISP1, DISP2 Delay Time | tDSPD | — | 90 | — | 80 | — | 70 | — | 50 | ns | |
| 70 | CUD1, CUD2 Delay Time | tCUDD | — | 90 | — | 80 | — | 70 | — | 50 | ns | |
| 71 | EXSYNC Output Delay Time | tEXD | 20 | 90 | 20 | 80 | 20 | 70 | 15 | 50 | ns | |
| 72 | CHR Delay Time | tCHD | — | 90 | — | 80 | — | 70 | — | 50 | ns | |
| 73 | | | | | | | | | | | | |
| 74 | | | | | | | | | | | | |



EXSYNC Input Timing

| No. | Item | Symbol | 4 MHz Version | | 6 MHz Version | | 8 MHz Version | | 9.8 MHz Version | | Unit | Reference Figure Number |
|-----|--------------------------|--------|---------------|-----|---------------|-----|---------------|-----|-----------------|-----|------|-------------------------|
| | | | Min | Max | Min | Max | Min | Max | Min | Max | | |
| 75 | EXSYNC Input Pulse Width | tEXSW | 3 | — | 3 | — | 3 | — | 3 | — | tcyc | 30 |
| 76 | EXSYNC Input Setup Time | tEXS | 60 | — | 60 | — | 50 | — | 30 | — | ns | |
| 77 | EXSYNC Input Hold Time | tEXH | 15 | — | 15 | — | 15 | — | 10 | — | ns | |

LPSTB Input Timing

| No. | Item | Symbol | 4 MHz Version | | 6 MHz Version | | 8 MHz Version | | 9.8 MHz Version | | Unit | Reference Figure Number |
|-----|--------------------------|--------|---------------|-----|---------------|-----|---------------|-----|-----------------|-----|------|-------------------------|
| | | | Min | Max | Min | Max | Min | Max | Min | Max | | |
| 78 | LPSTB Uncertain Time 1 | tLPD1 | 70 | — | 70 | — | 70 | — | 45 | — | ns | 31, 32 |
| 79 | LPSTB Uncertain Time 2 | tLPD2 | 10 | — | 10 | — | 10 | — | 10 | — | ns | |
| 80 | LPSTB Input Hold Time | tLPH | 10 | — | 10 | — | 10 | — | 10 | — | ns | |
| 81 | LPSTB Input Inhibit time | tLPI | 4 | — | 4 | — | 4 | — | 4 | — | tcyc | |

RES and DACK Input Timing

| No. | Item | Symbol | 4 MHz Version | | 6 MHz Version | | 8 MHz Version | | 9.8 MHz Version | | Unit | Reference Figure Number |
|-----|-------------------------|--------|---------------|-----|---------------|-----|---------------|-----|-----------------|-----|------|-------------------------|
| | | | Min | Max | Min | Max | Min | Max | Min | Max | | |
| 82 | DACK Setup Time for RES | tDAKSR | 100 | — | 100 | — | 100 | — | 100 | — | ns | 33 |
| 83 | DACK Hold Time for RES | tDAKHR | 0 | — | 0 | — | 0 | — | 0 | — | ns | |
| 84 | RES Input Pulse Width | tRES | 10 | — | 10 | — | 10 | — | 10 | — | tcyc | |



IRQ and Attributes Output Cycle Timing

| No. | Item | Symbol | 4 MHz Version | | 6 MHz Version | | 8 MHz Version | | 9.8 MHz Version | | Unit | Reference Figure Number |
|-----|------------------|--------|---------------|-----|---------------|-----|---------------|-----|-----------------|-----|------|-------------------------|
| | | | Min | Max | Min | Max | Min | Max | Min | Max | | |
| 85 | IRQ Delay Time 1 | tIRQ1 | — | 250 | — | 200 | — | 150 | — | 150 | ns | 34, 35 |
| 86 | IRQ Delay Time 2 | tIRQ2 | — | 500 | — | 500 | — | 500 | — | 500 | ns | |
| 87 | ATR Delay Time 1 | tATR1 | — | 100 | — | 90 | — | 80 | — | 60 | ns | 28 |
| 88 | ATR Hold Time 1 | tATRH1 | 10 | — | 10 | — | 10 | — | 5 | — | ns | |
| 89 | | | | | | | | | | | | |
| 90 | ATR Delay Time 2 | tATR2 | — | 100 | — | 90 | — | 80 | — | 60 | ns | 28 |
| 91 | ATR Hold Time 2 | tATRH2 | 10 | — | 10 | — | 10 | — | 5 | — | ns | |

Synchronous Bus Timing, MPU or DMA Read/Write Cycle

| No. | Item | Symbol | 4 MHz Version | | 6 MHz Version | | 8 MHz Version | | 9.8 MHz Version | | Unit | Reference Figure Number |
|-----|-----------------------|---------|---------------|-----|---------------|-----|---------------|-----|-----------------|-----|------|-------------------------|
| | | | Min | Max | Min | Max | Min | Max | Min | Max | | |
| 100 | CS Cycle Time | tCSC | 4 | — | 4 | — | 4 | — | 4 | — | tcyc | 18, 19 |
| 101 | CS Low Level Width | tWCSSL | 2 | — | 2 | — | 2 | — | 2 | — | tcyc | |
| 102 | CS High Level Width | tWCSSH | 2 | — | 2 | — | 2 | — | 2 | — | tcyc | |
| 104 | DACK Cycle Time | tDACKC | 4 | — | 4 | — | 4 | — | 4 | — | tcyc | 22, 24 |
| 105 | DACK Low Level Width | tWDACKL | 2 | — | 2 | — | 2 | — | 2 | — | tcyc | |
| 106 | DACK High Level Width | tWDACKH | 2 | — | 2 | — | 2 | — | 2 | — | tcyc | |

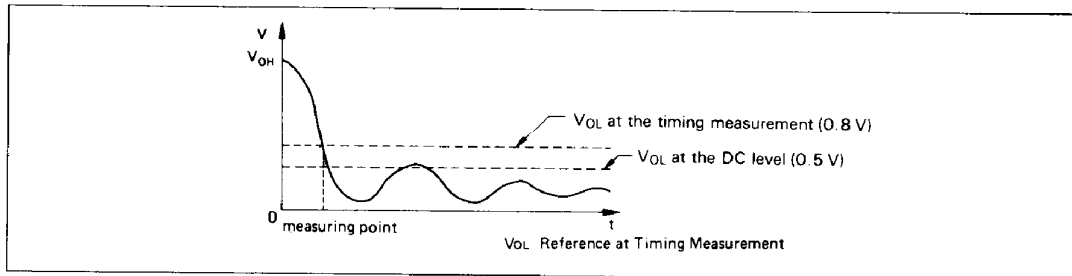


Figure 16. Test Points

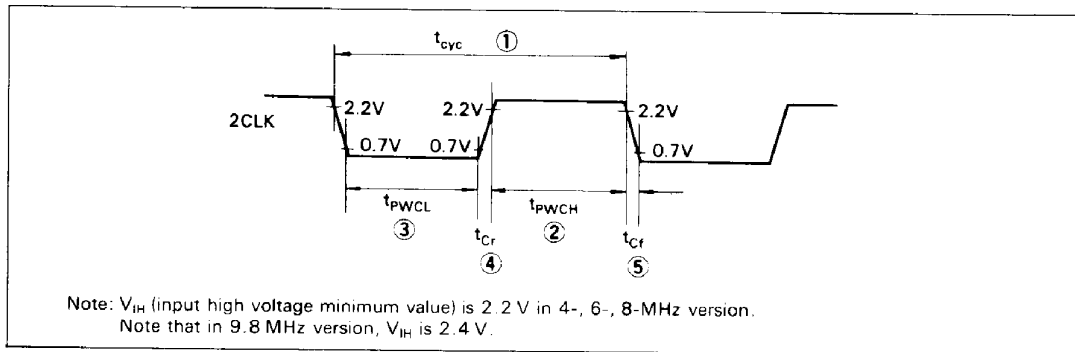
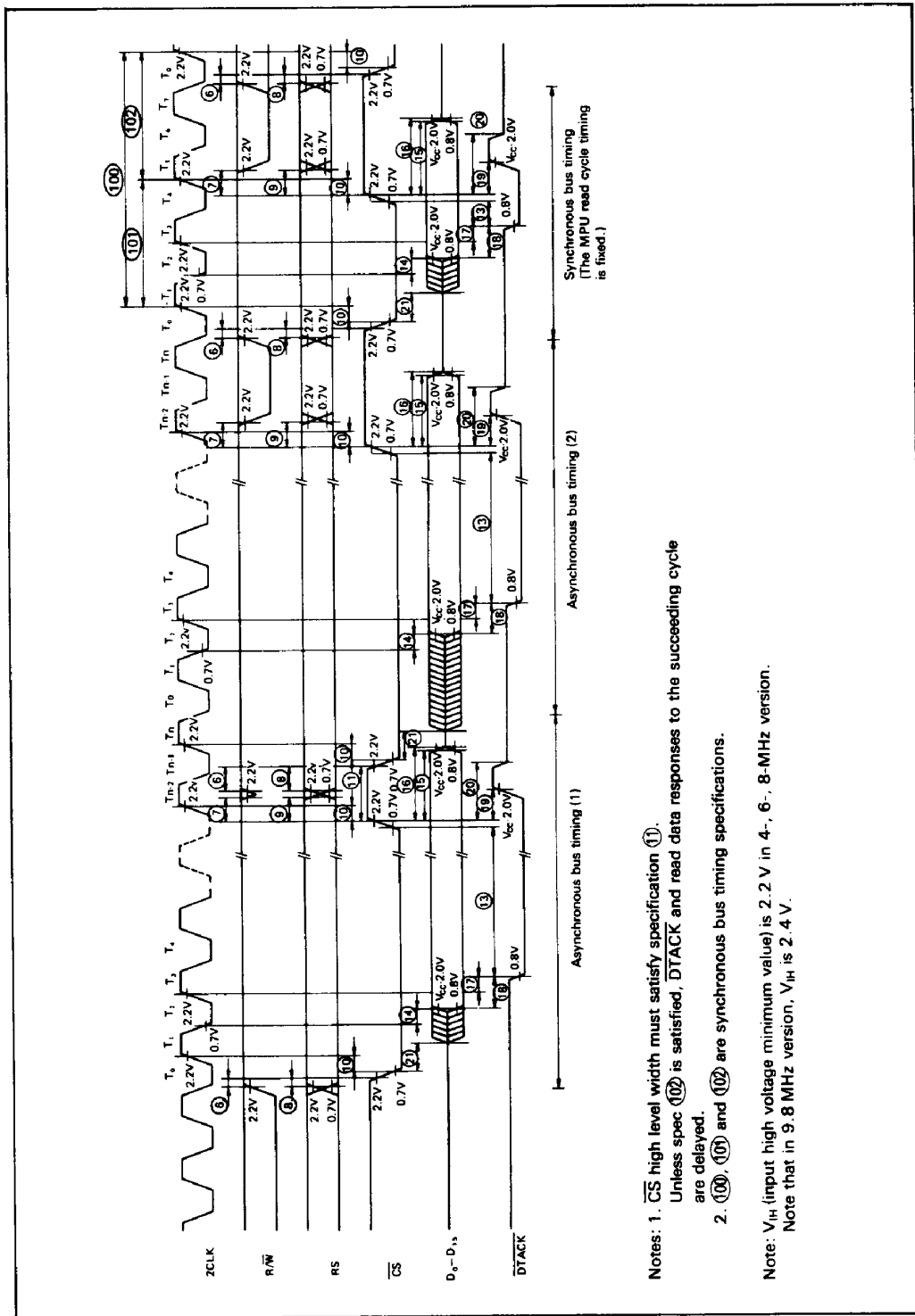


Figure 17. 2CLK Waveform





- Notes: 1. CS high level width must satisfy specification (1).
 Unless spec (10) is satisfied, DTACK and read data responses to the succeeding cycle are delayed.
2. (10), (11) and (12) are asynchronous bus timing specifications.
- Note: V_{IH} (input high voltage minimum value) is 2.2 V in 4-, 6-, 8-MHz version.
 Note that in 9.8 MHz version, V_{IH} is 2.4 V.

Figure 18. MPU Read Cycle Timing (MPU ← ACRTC)

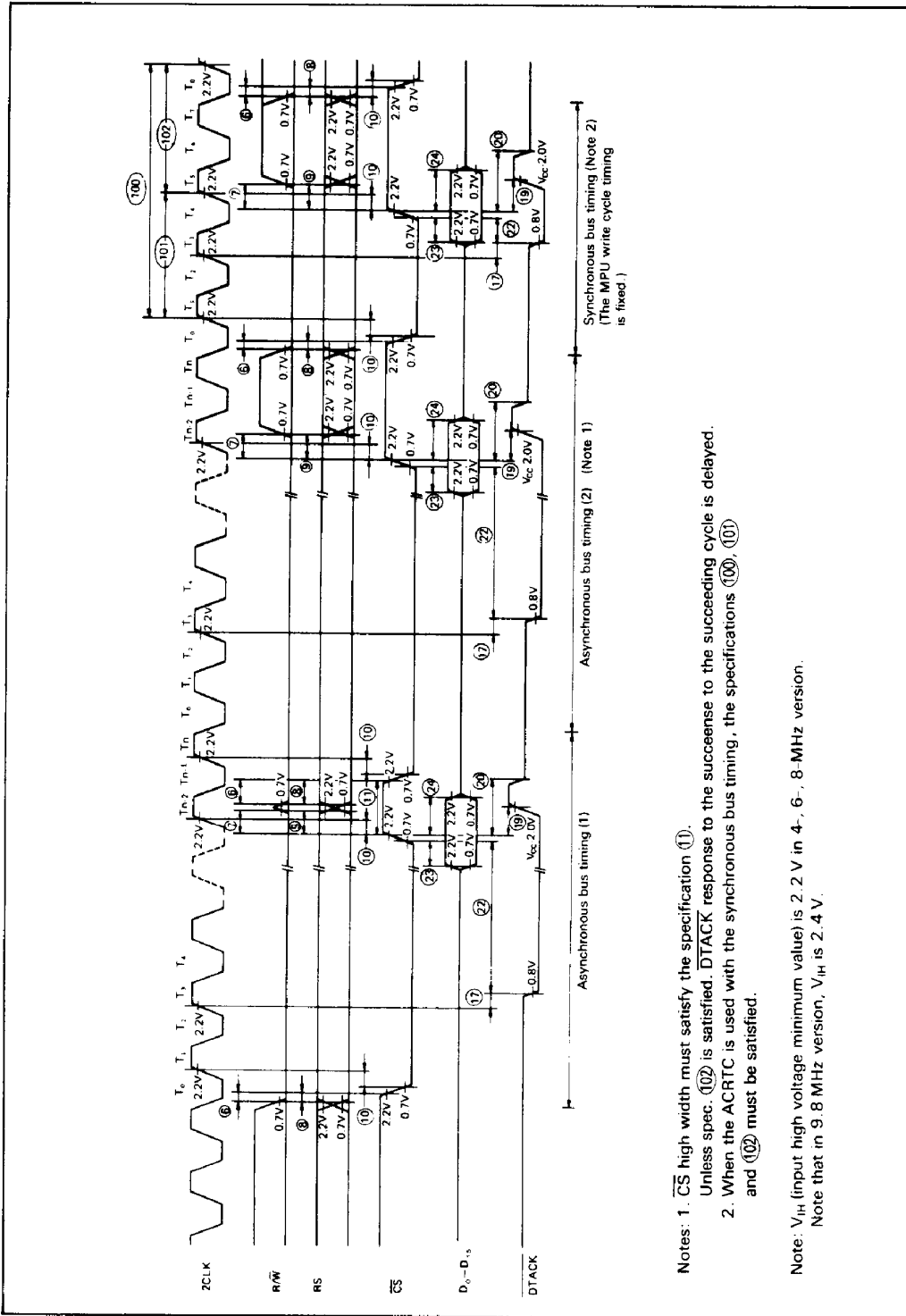
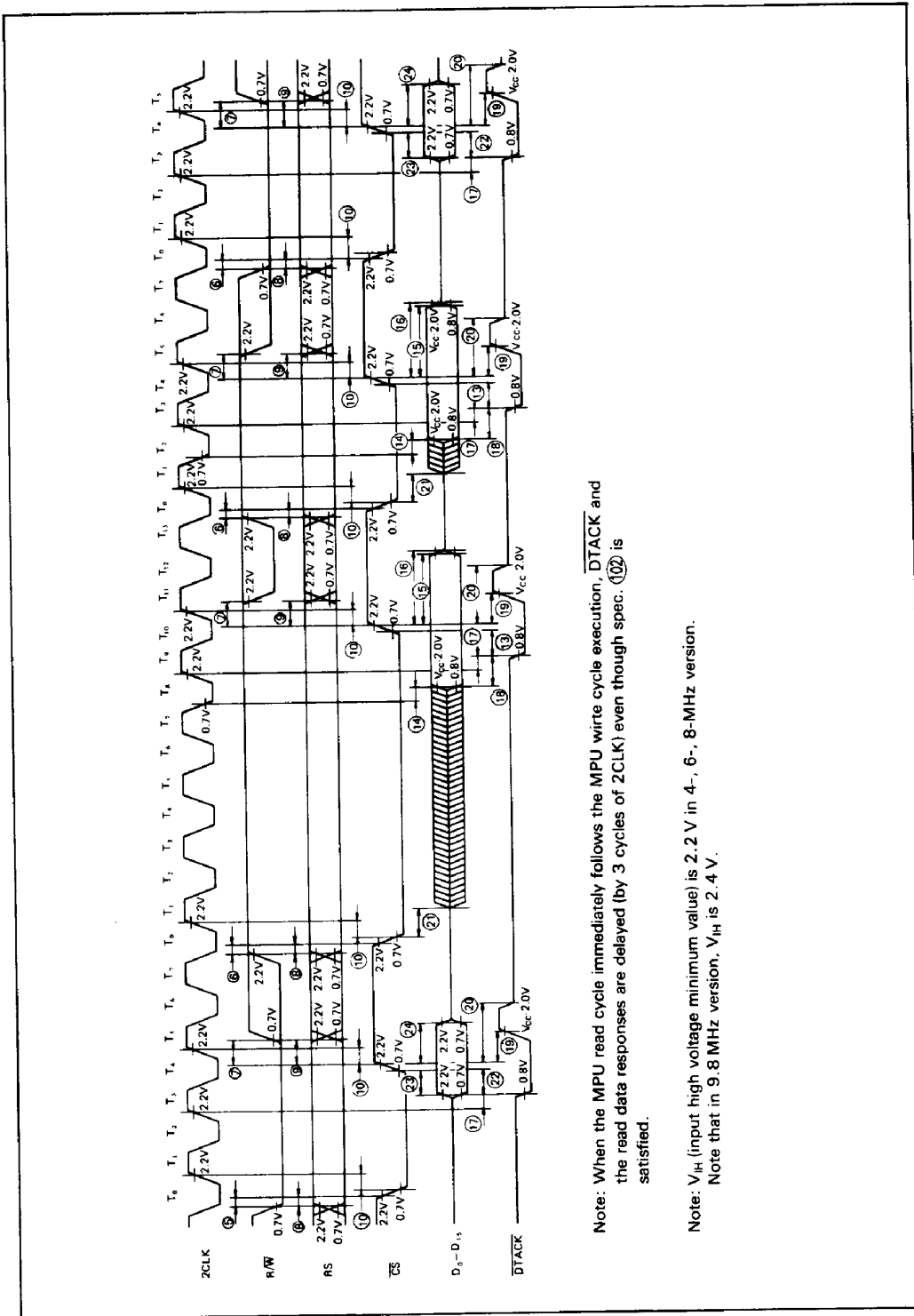


Figure 19. MPU Write Cycle Timing (MPU → ACRTC)



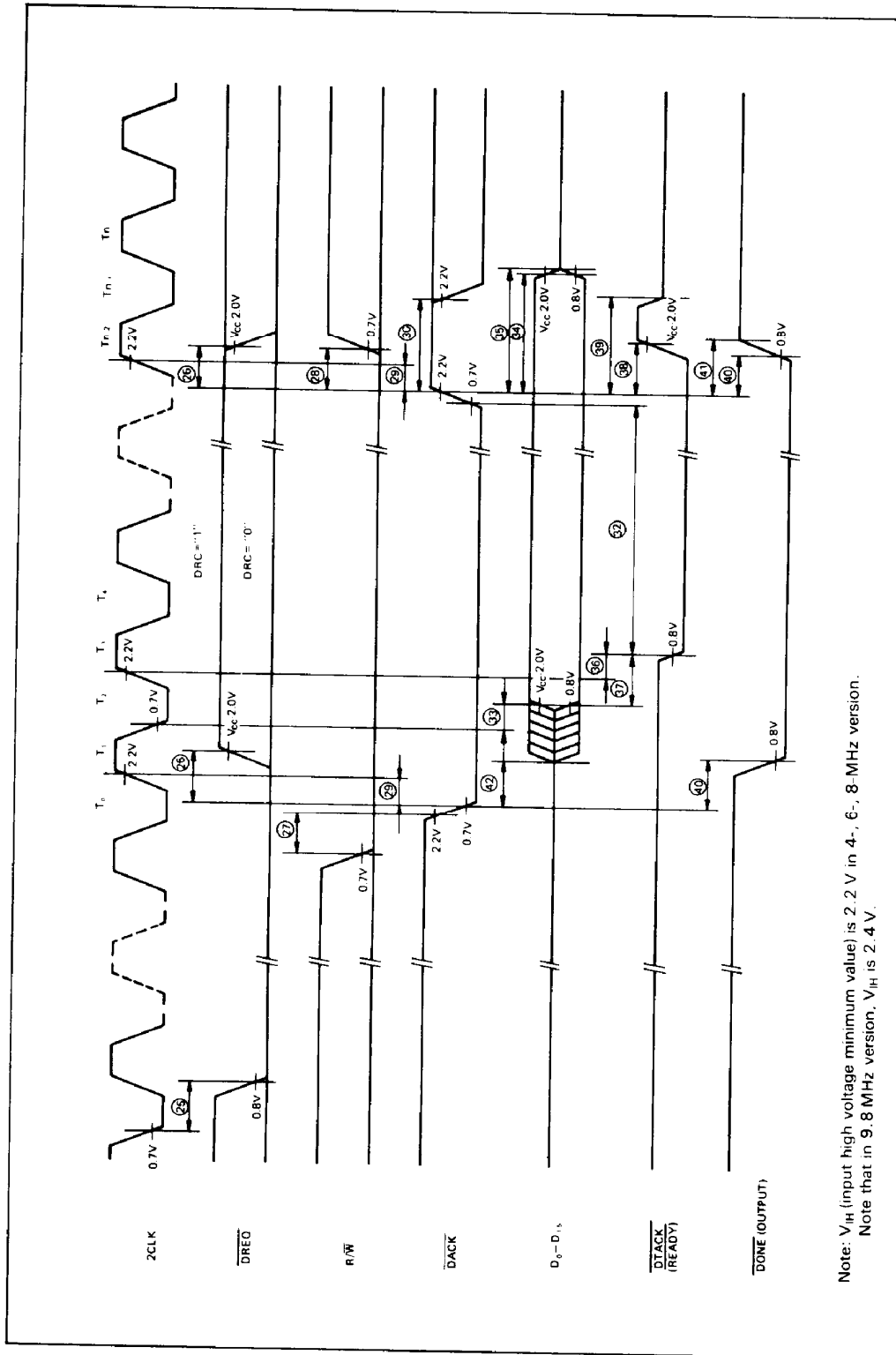


Note: When the MPU read cycle immediately follows the MPU write cycle execution, DTACK and the read data responses are delayed (by 3 cycles of 2CLK) even though spec. (10) is satisfied.

Note: V_{IH} (input high voltage minimum value) is 2.2 V in 4-, 6-, 8-MHz version. Note that in 9.8 MHz version, V_{IH} is 2.4 V.

Figure 20. MPU Read/Write Cycle Timing (MPU -> ACRTC)

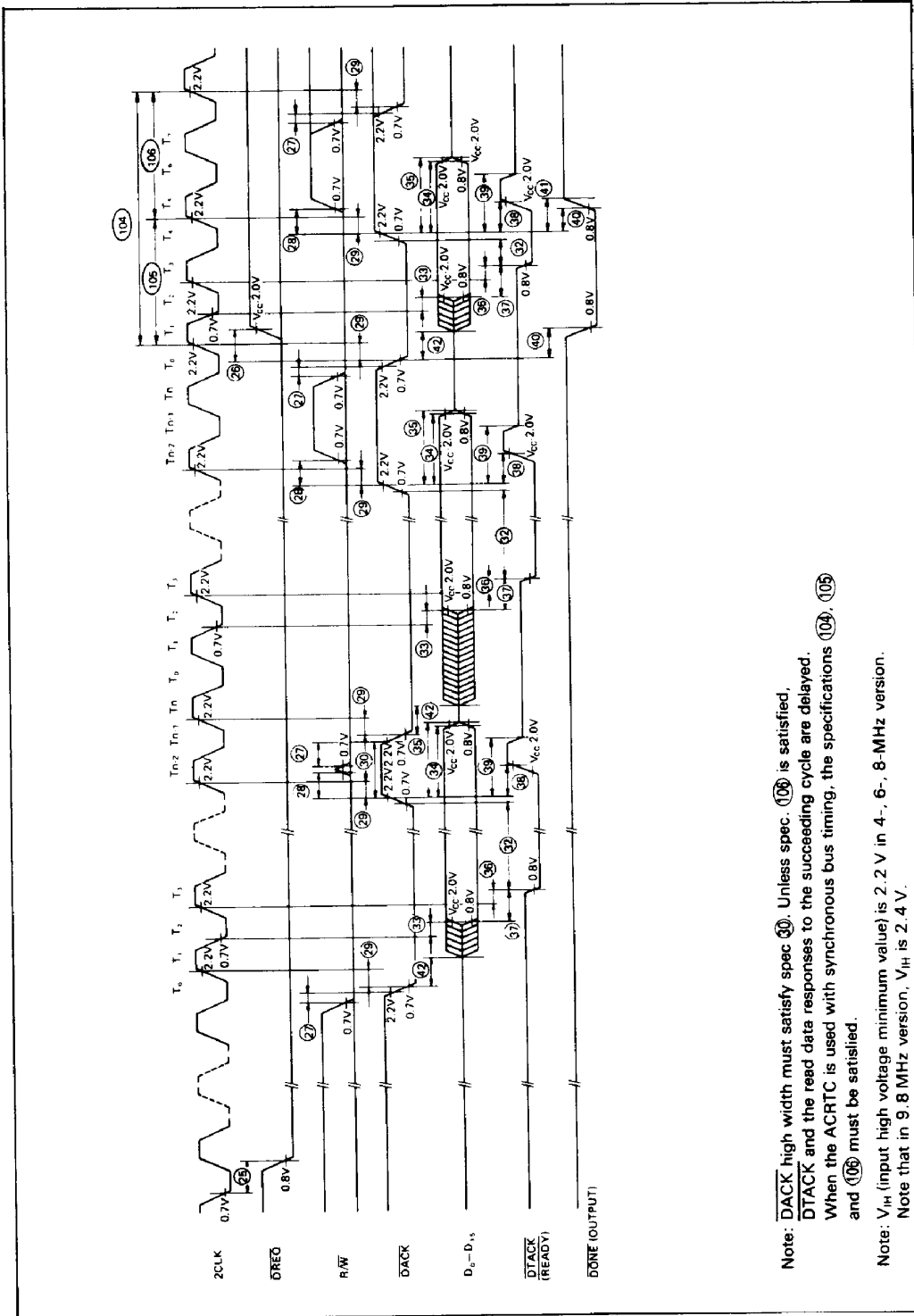




Note: V_{IH} (input high voltage minimum value) is 2.2 V in 4-, 6-, 8-MHz version.
 Note that in 9.8 MHz version, V_{IH} is 2.4 V.

Figure 21. DMA Read Cycle Timing (Memory - ACRTC)





Note: \overline{DACK} high width must satisfy spec 30. Unless spec. 106 is satisfied, \overline{DTACK} and the read data responses to the succeeding cycle are delayed. When the ACRTC is used with synchronous bus timing, the specifications 104, 105 and 106 must be satisfied.

Note: V_{IH} (input high voltage minimum value) is 2.2 V in 4-, 6-, 8-MHz version. Note that in 9.8 MHz version, V_{IH} is 2.4 V.

Figure 22. DMA Ready Cycle Timing (Memory - ACRTC): Burst Mode



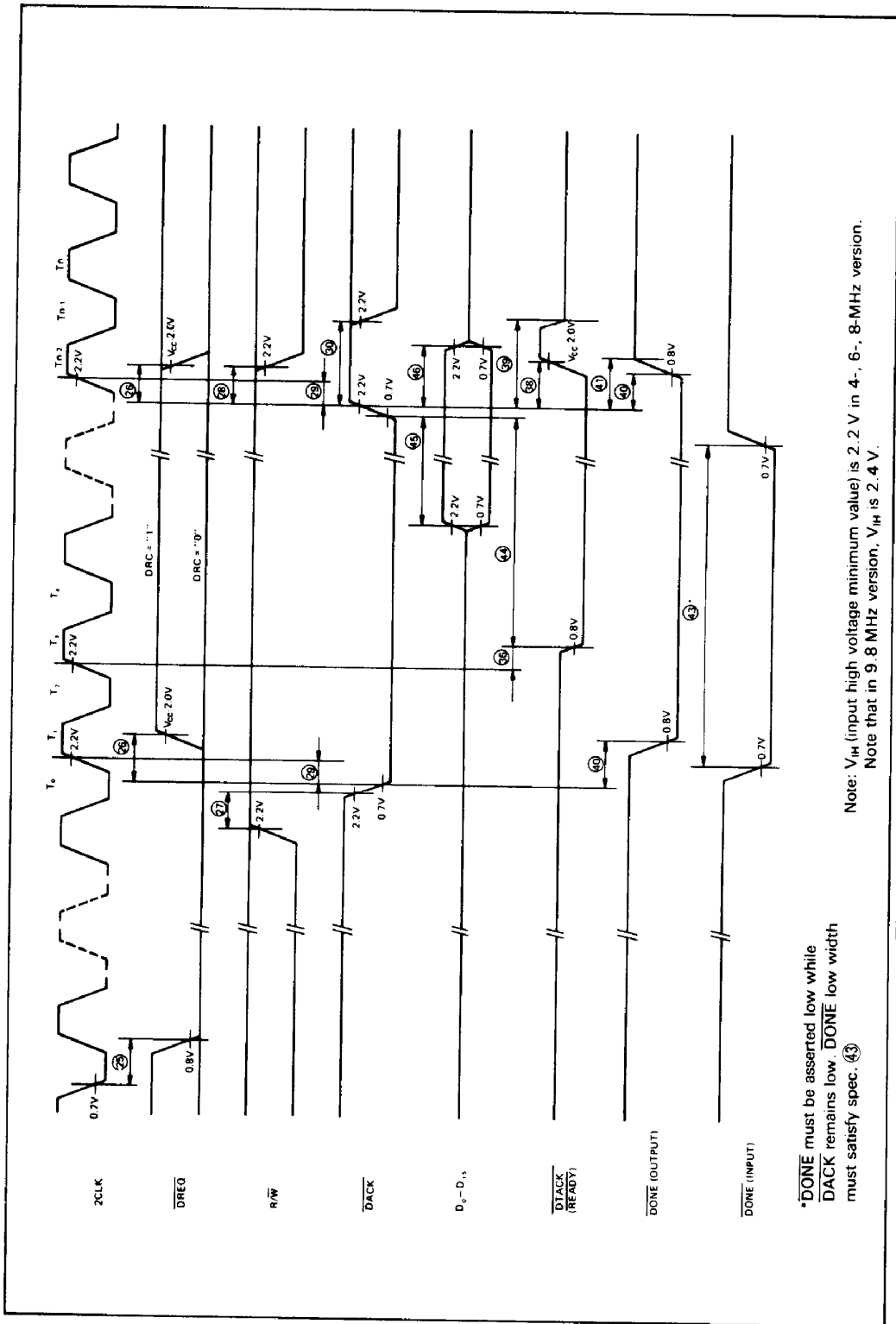
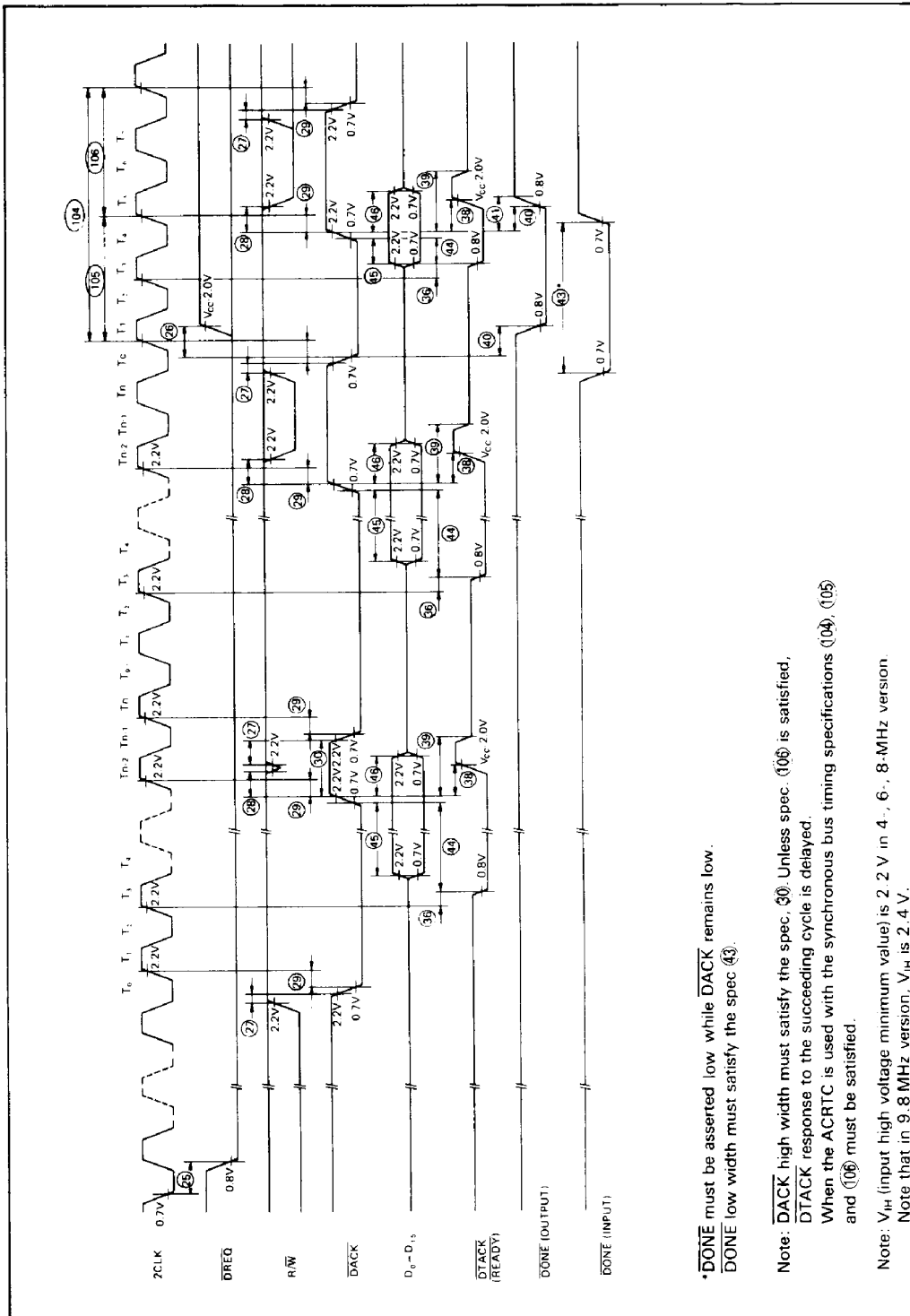


Figure 23. DMA Write Cycle Timing (Memory → ACRTC)





***DONE** must be asserted low while **DACK** remains low.
DONE low width must satisfy the spec (43).

Note: **DACK** high width must satisfy the spec. (30). Unless spec. (106) is satisfied, **DTACK** response to the succeeding cycle is delayed.
 When the ACRTC is used with the synchronous bus timing specifications (104), (105) and (106) must be satisfied.

Note: V_{IH} (input high voltage minimum value) is 2.2V in 4-, 6-, 8-MHz version.
 Note that in 9.8 MHz version, V_{IH} is 2.4V.

Figure 24. DMA Write Cycle Timing (Memory - ACRTC): Burst Mode

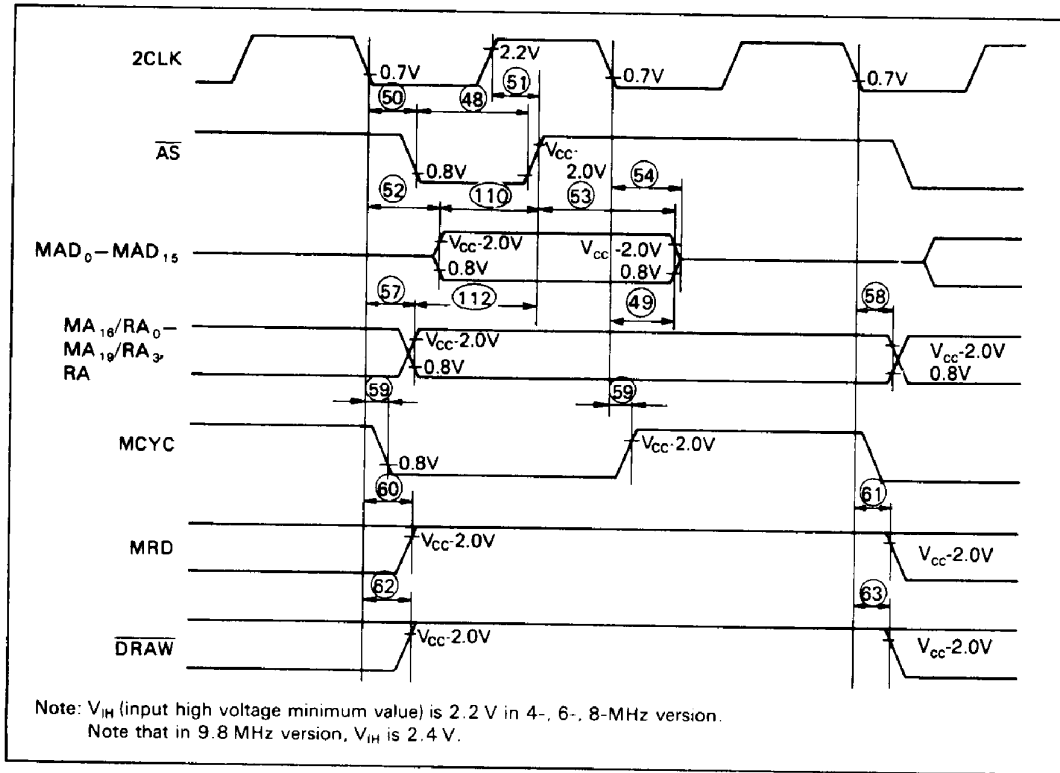


Figure 25. Display Cycle Timing



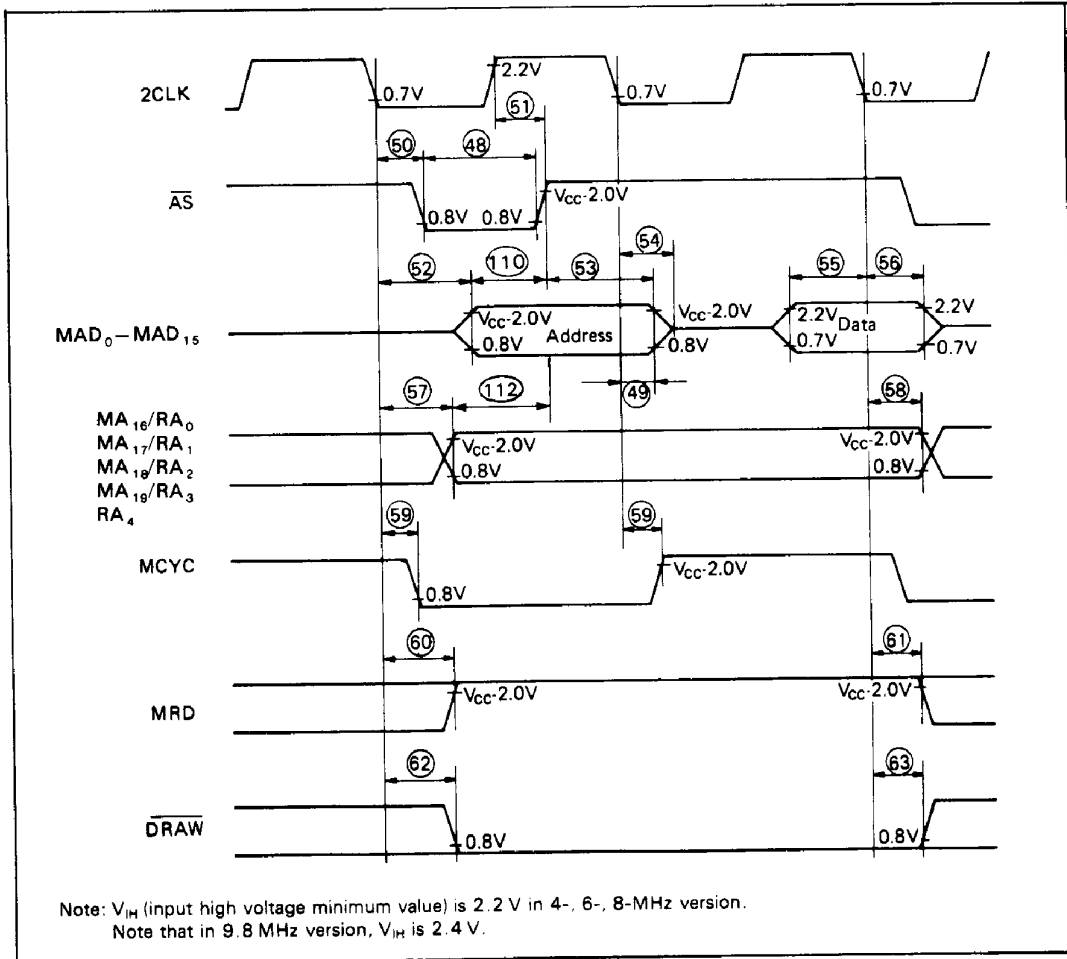


Figure 26. Frame Memory Read Cycle Timing (ACRTC - Frame Memory)

2



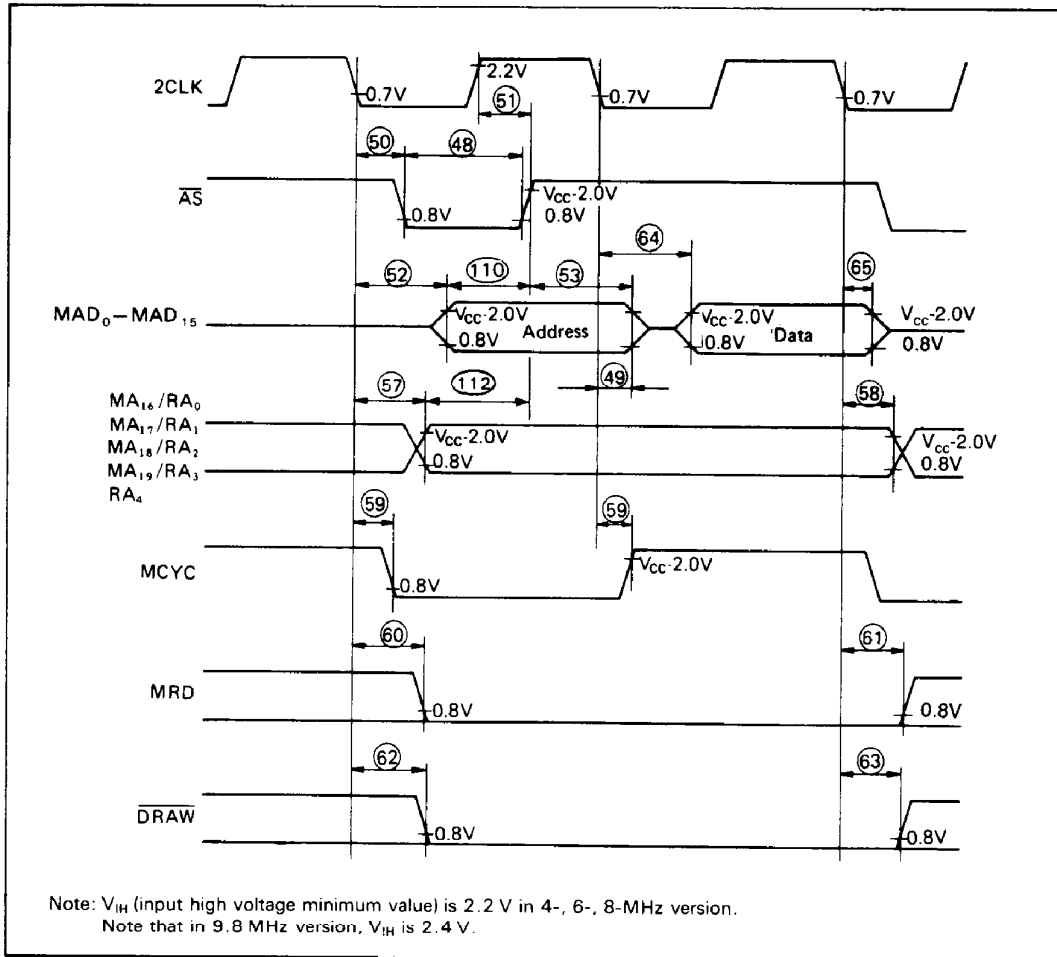


Figure 27. Frame Memory Write Cycle Timing (ACRTC → Frame Memory)



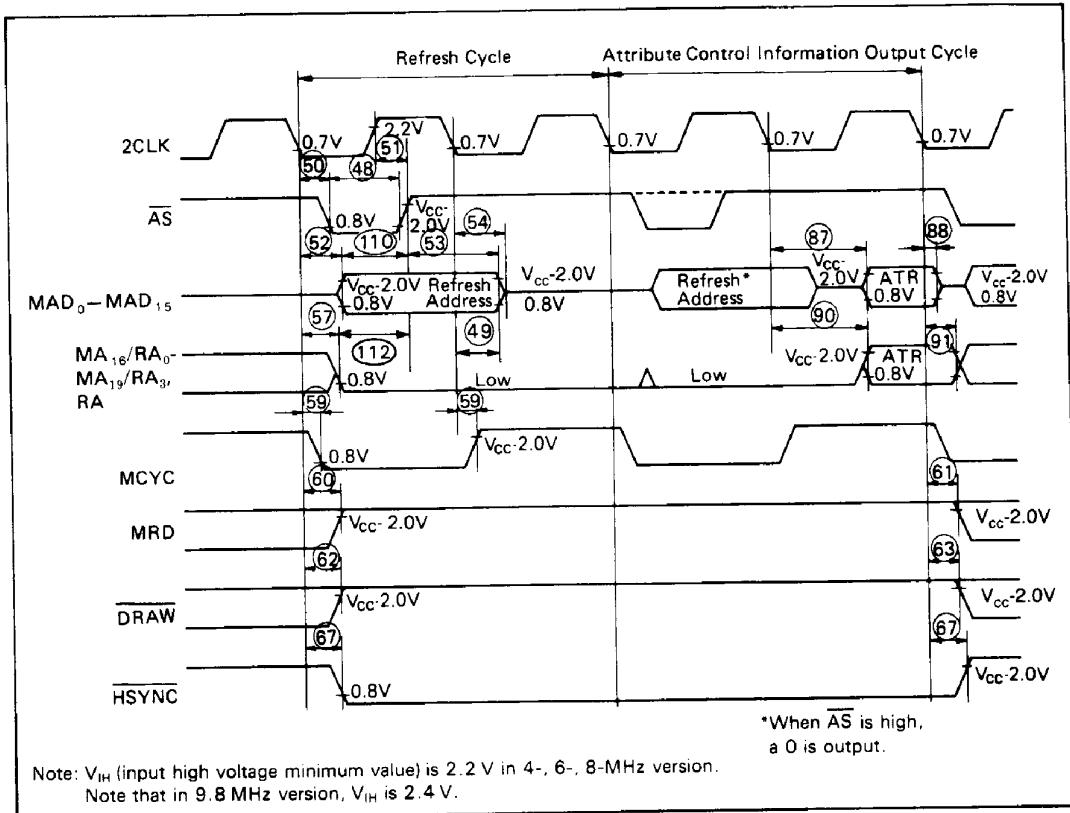


Figure 28. Frame Memory Refresh/Video Attributes Output Cycle Timing

2



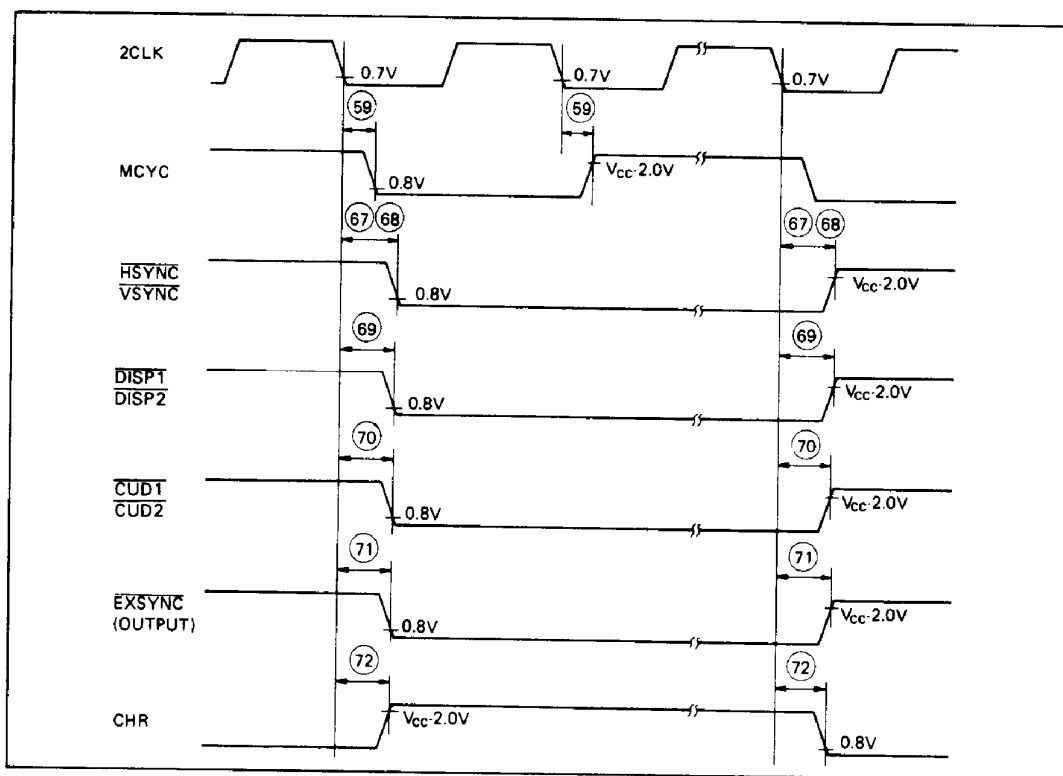


Figure 29. Display Control Signal Output Timing



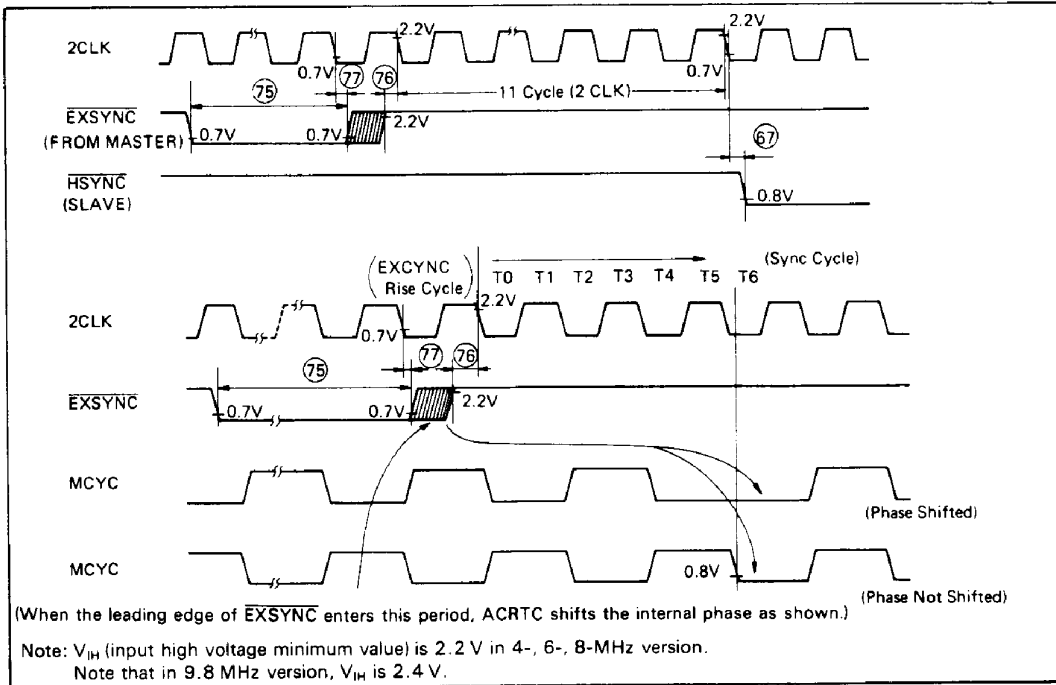


Figure 30. EXSYNC Input Timing

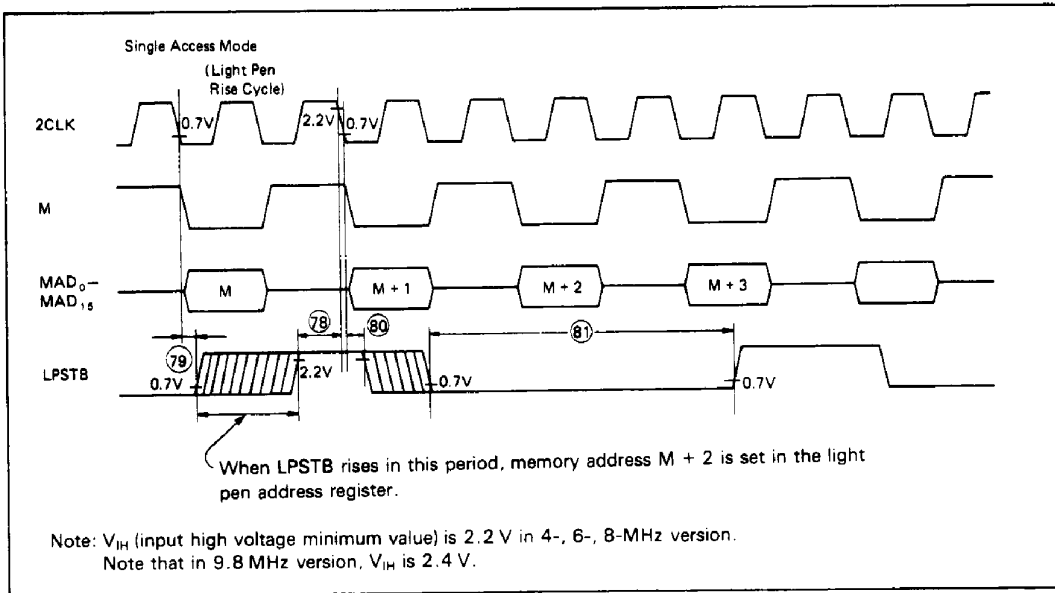


Figure 31. LPSTB Input Timing (Single Access Mode)

2



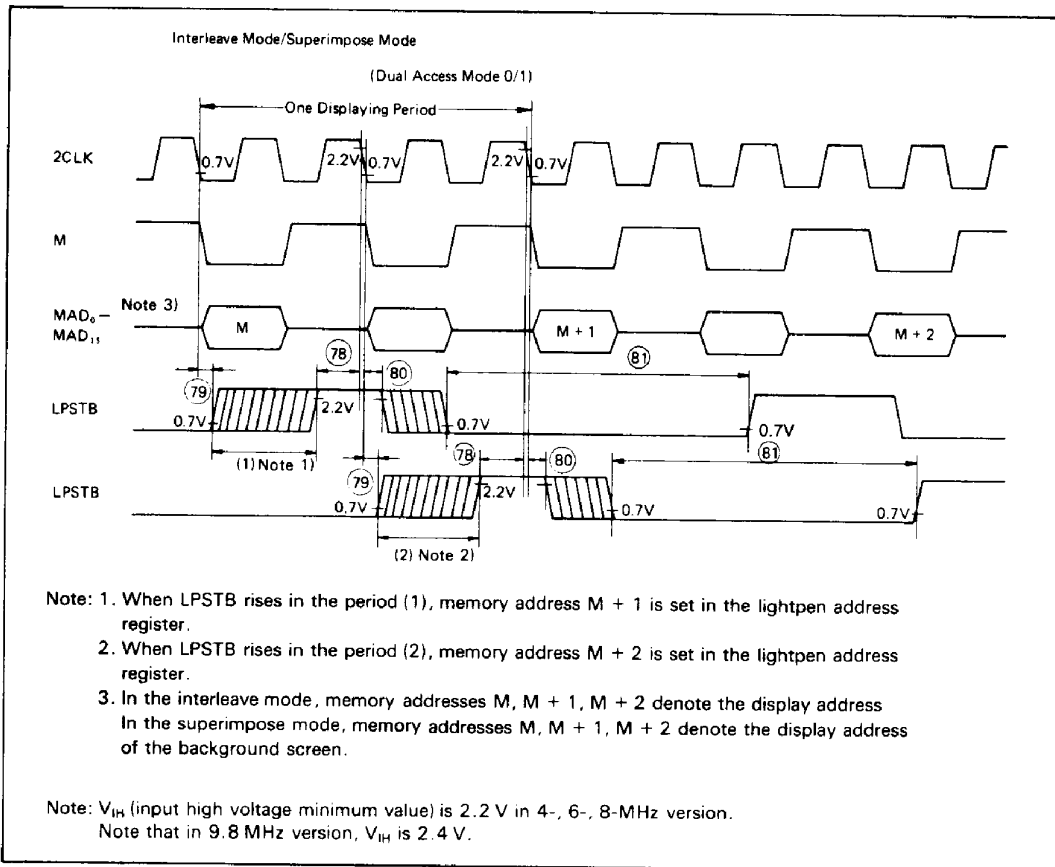


Figure 32. LPSTB Input Timing (Dual Access Mode)

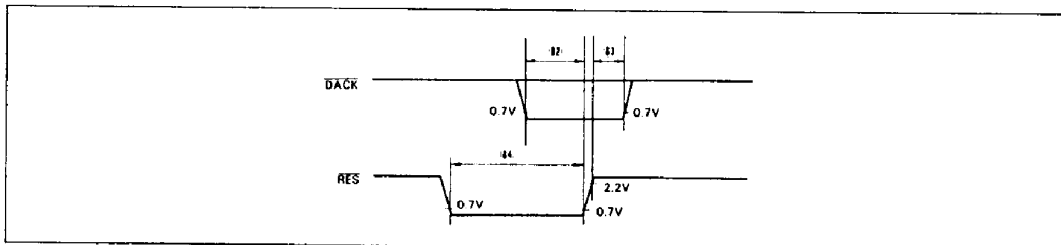


Figure 33. RES Input and DACK Input Timing (System Reset and 16-Bit/8-Bit Selection)

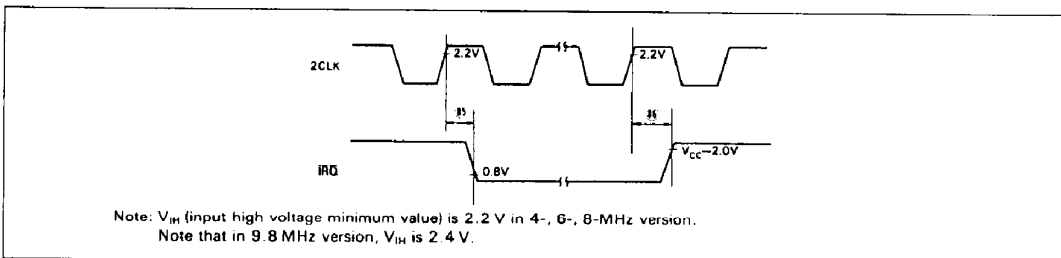


Figure 34. IRQ Output Timing



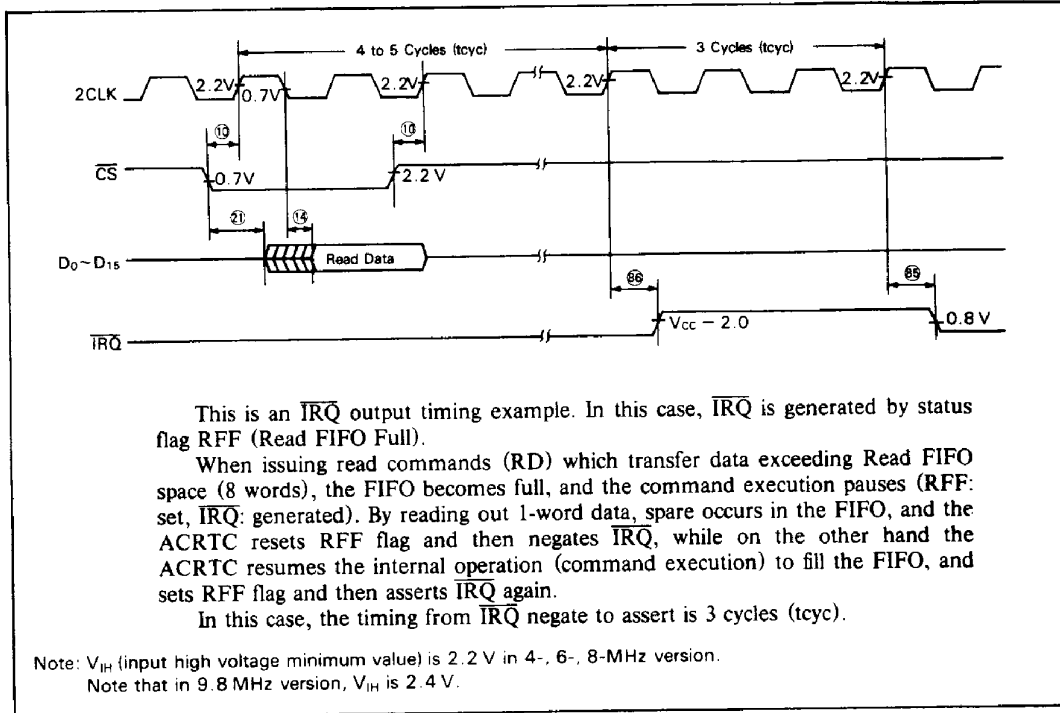


Figure 35. \overline{IRQ} Output Timing (Example: Read FIFO Full Interrupt Enable)

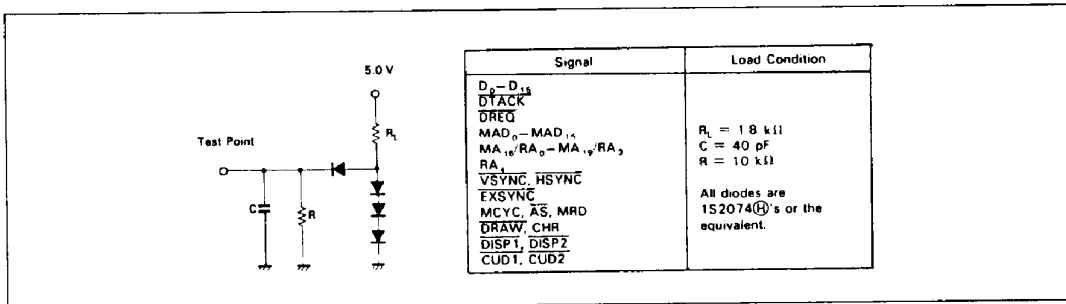


Figure 36. Test Load Circuit A

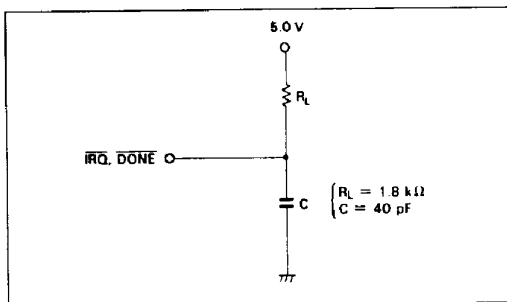


Figure 37. Test Load Circuit B



2