

## FEATURES

- **High-speed graphics**
  - **Drawing rate: 200 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 coordinate 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 MPU memory.**
  - **Maximum Resolution: 4096 x 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 ARTCs 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: 25MHz**
- **CMOS, single +5V power supply**

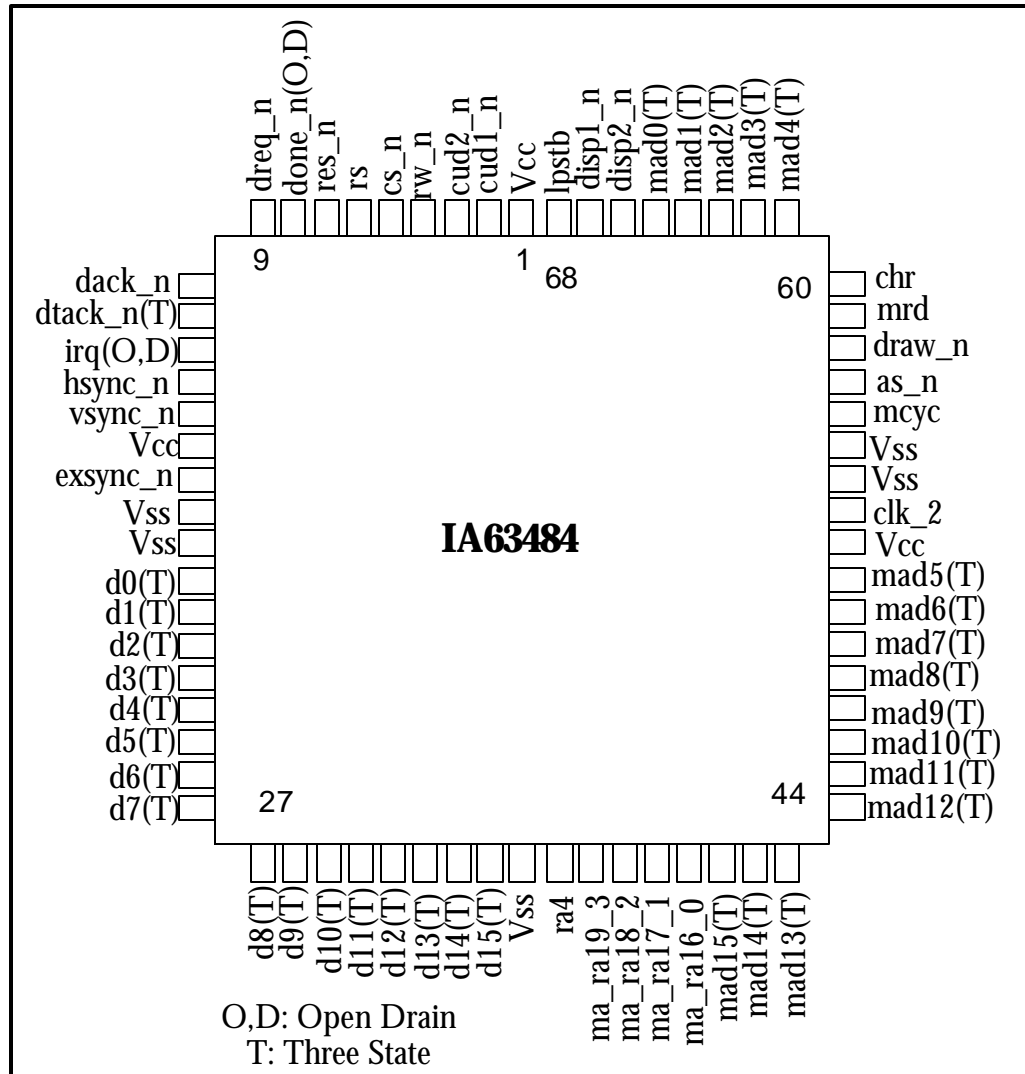
The IA63484 is a "plug-and-play" drop-in replacement for the original Hitachi© HD63484. This replacement IC has been developed using innovASIC's MILES™, or Managed IC Lifetime Extension System, cloning technology. This technology produces replacement ICs far more complex than "emulation" while ensuring they are compatible with the original IC. MILES™ captures the design of a clone so it can be produced even as silicon technology advances. MILES™ also verifies the clone against the original IC so that even the "undocumented features" are duplicated. This data sheet documents all necessary engineering information about the IA63484 including functional and I/O descriptions, electrical characteristics, and applicable timing.

# IA63484

## Advanced CRT Controller

### 68 Pin Package: ACRTC PLCC PINOUT

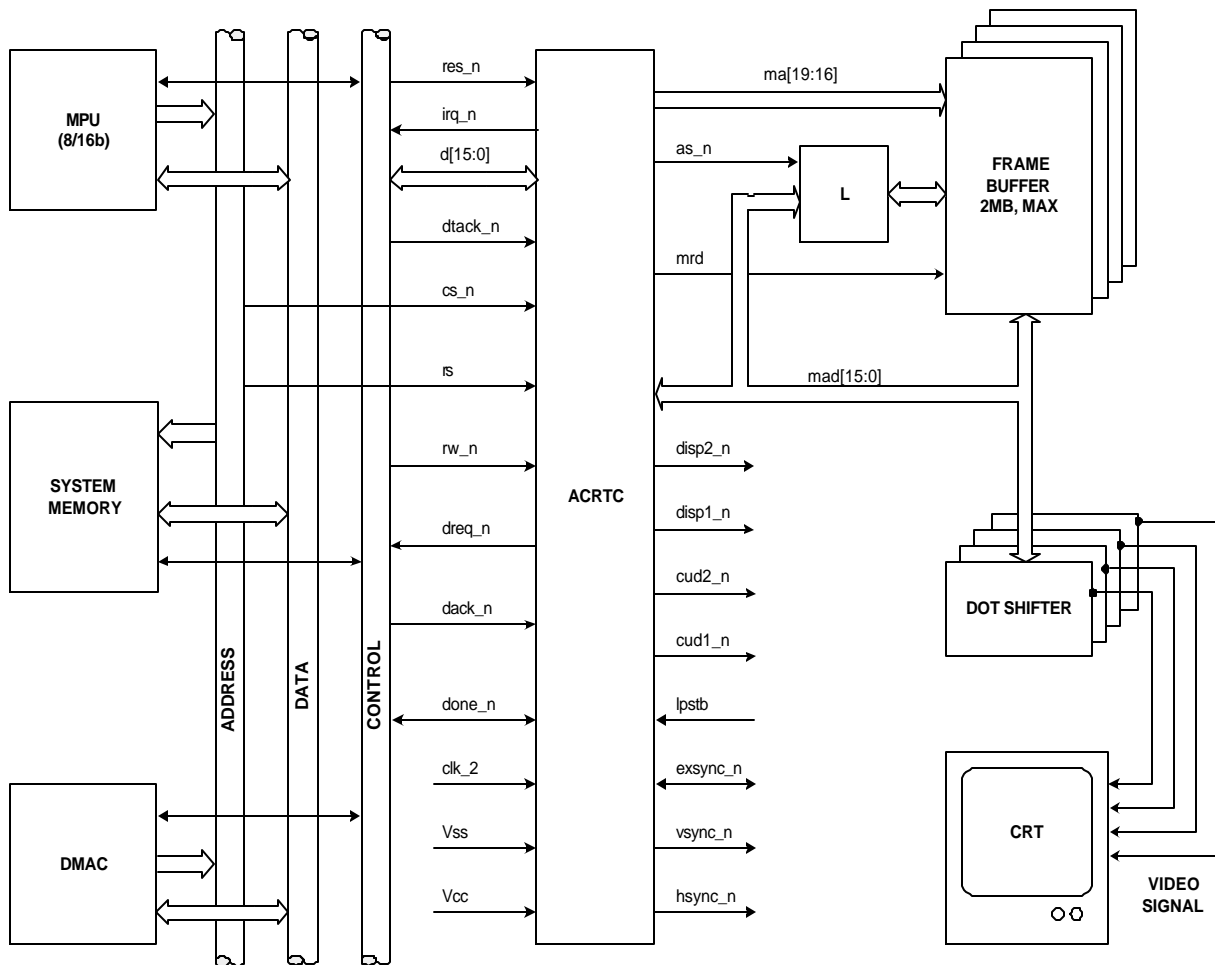
#### Pin Arrangement:



**BLOCK DIAGRAM**

**Figure 1: System Block Diagram**

Figure 2 illustrates the ACRTC system environment. The following paragraphs will further describe the system block diagram and design in more detail.



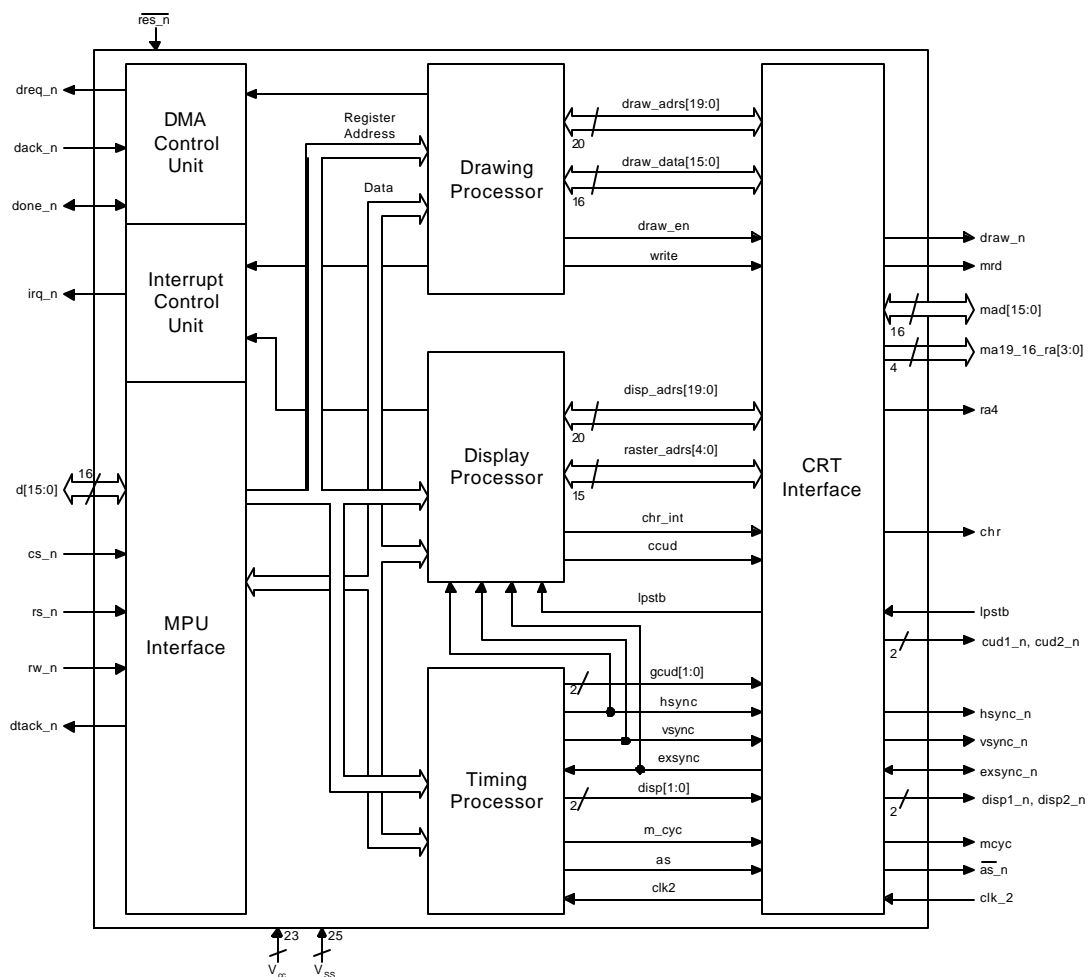
**I/O SIGNAL DESCRIPTION:**

The diagram below describes the I/O characteristics for each signal on the IC. The signal names correspond to the signal names on the pinout diagrams provided.

**I/O Characteristics:**

Signal Name	I/O	Group	Description
res_n	I	MPU Interface	ACRTC reset:
d[15,0]	I/O		Data bus (three state): are the bidirectional data bus to the host mpu or dmac. D <sub>0</sub> -D <sub>15</sub> are used in 8-bit data bus mode.
rw_n	I		Read/write strobe: controls the direction of host/ACRTC transformers.
cs_n	I		Chip Select: enables transfers between the host and the ACRTC.
rs	I		Register Select: selects the ACRTC register to be accessed. It is usually connected to the least significant bit of the host address bus.
dtack_n	O		Data transfer acknowledge (three state): output provides asynchronous bus cycle timing. It is compatible with the HD68000 mpu dtack output.
irq_n	O		Interrupt request (open drain): output generates interrupt service requests to the host MPU.
dreq_n	I	DMAC Interface	DMA request: receives DMA acknowledge timing from the host DMAC.
dack	I/O		DMA acknowledge:
done_n	I		DMA done: terminates DMA transfer. It is compatible with the HD68450 DMAC DONE signal.
clk_2	I/O	CRT Interface	ARTC clock: is the basic operating clock, twice the frequency of the dot clock.
mad[15,0]	O		Multiplexed frame buffer address/data bus: are the multiplexed frame buffer address/data bus.
as_n	O		Address strobe: output demultiplexes the address/data bus.
MA <sub>16</sub> /R <sub>0</sub> * MA <sub>19</sub> /RA <sub>3</sub>	O		Higher-order address bits/character screen raster address: MA <sub>16</sub> /R <sub>0</sub> - MA <sub>19</sub> /RA <sub>3</sub> are the upper bits of the graphics screen address multiplexed with the lower bits of the character screen raster address.
RA <sub>4</sub>	O		Higher-order character screen raster address bit: is the high bit of the character screen raster address (up to 32 rasters.)
chr	O		Graphic or character screen access: output indicates whether a graphic or character screen is being accessed.
mcyc	O		Frame buffer memory access timing signal: is the frame buffer access timing output, 1/2 the frequency of clk_2.
mrd	O		Frame buffer memory read: output controls the frame buffer data bus direction.
draw_n	O		Draw/refresh signal: output differentiates between drawing and CRT display refresh cycles.
disp1, disp2	O		Display enable: programmable display enable outputs can enable, disable, and blank logical screens.
cud1, cud2			Cursor Display: outputs provides cursor timing programmed by ACRTC parameters such as cursor definition, cursor mode, cursor address, etc.
vsync_n	O		CRT vertical sync pulse: outputs the crt vertical synchronization pulse.
hsync_n			CRT horizontal sync pulse: outputs the crt horizontal synchronization pulse.
exsync_n	I/O		External sync: allows synchronization between multiple ACRTSs and other video signal generators.
lpstb	I		Lightpen strobe: is the lightpen input

**Figure 2: ACRTC Block Diagram**



## ACRTC System Description:

Some 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.

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 use of the MPU system bus by the ACRTC. A related benefit is that a large frame buffer (2 MB 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 Direct Memory Access Controller (DMAC) to increase 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.

More cost-sensitive or less performance-sensitive applications might not require a DMAC. In these cases, 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.

---

## **FUNCTIONAL REQUIREMENTS:**

### **Drawing Processor:**

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

The drawing processor uses three operation control units; the Drawing Algorithm Control unit, the Drawing Address Generation unit and the Logical Operation unit.

The Drawing Algorithm Control Unit interprets graphic commands and parameters and executes the appropriate micro-programmed drawing algorithm. This control unit calculates coordinates using logical pixel X-Y addressing.

The Drawing Address Generation Unit converts logical X-Y addresses from the Drawing Algorithm Control unit to a bit address in the frame buffer. The frame buffer is organized as sequential 16 bit words. The bit address consists of 20 bits and bits 0-4 specifying the logical pixel bit address within the physical frame buffer word.

Logical Operation Unit, using the address calculated in the drawing algorithm control and drawing address generation units, performs logical operations between the existing read data in the frame buffer and the drawing pattern in the pattern RAM, and rewrites the results into the frame buffer. A detailed description of the Drawing Processor is contained in its module specification.

### **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 and row address) modes.

### **Display Functions:**

The ACRTC allows the frame buffer to be divided into four separate logical screens:

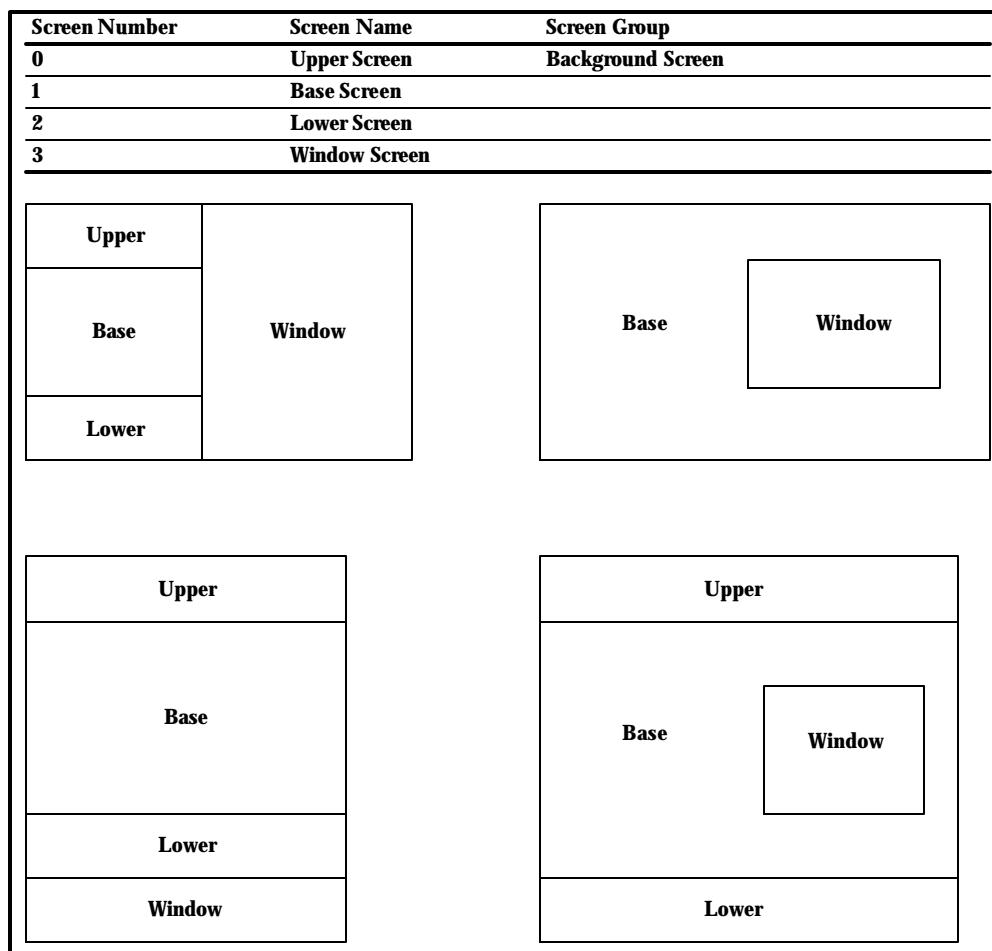
- Upper
- Base
- Lower
- Window

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. The window screen is unique, since the ACRTC usually gives it higher priority than the background screens. A typical application might be to use the base screen for the bulk of the user interaction, while using the upper screen for pull-down menus and the lower screen for status line indicators. The exception is in the ACRTC superimpose 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. Figure 3 is an example of the screen combinations.

**Figure 3: Screen Combination Examples**



**Display Control:**

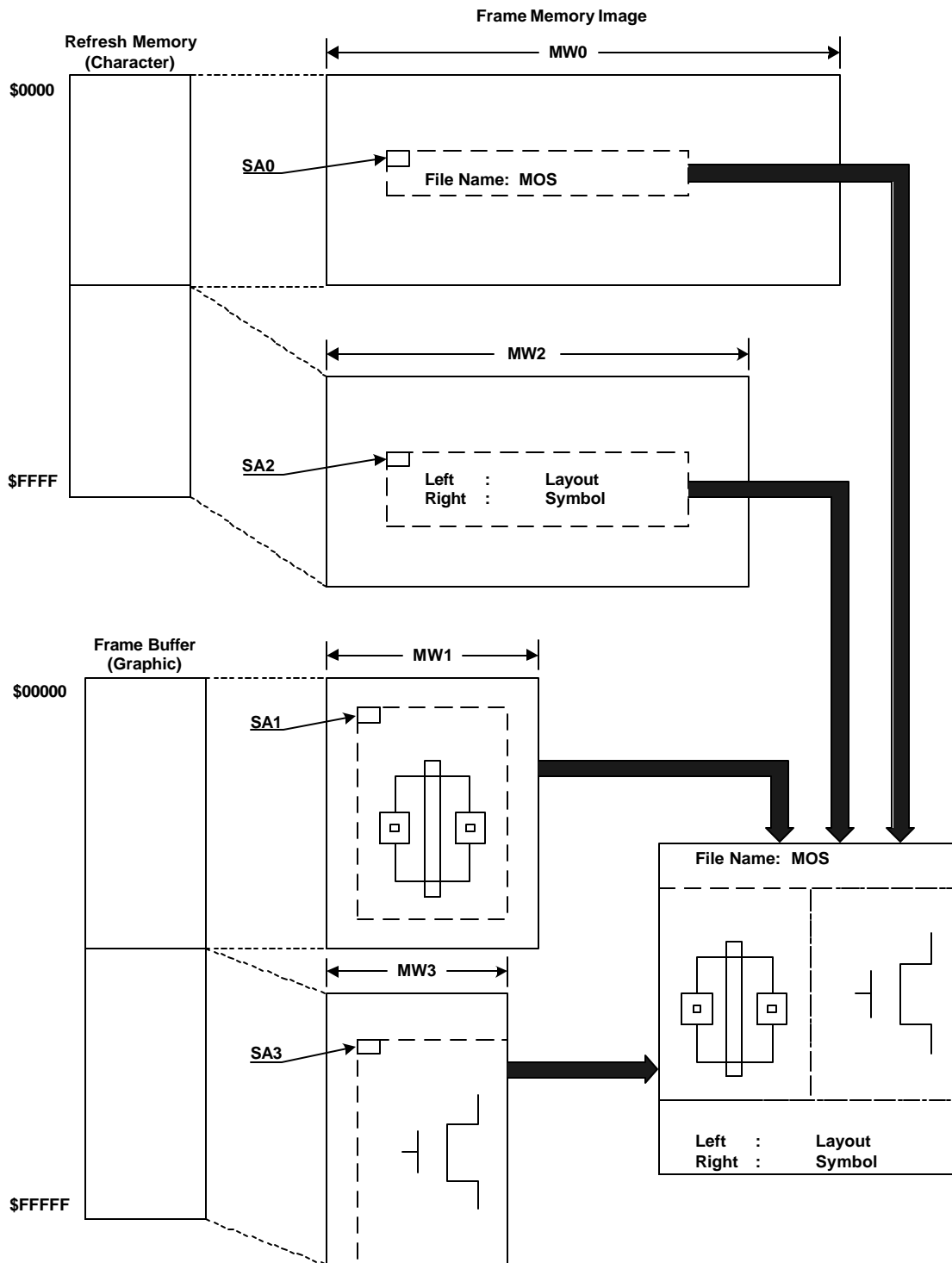
The ACRTC can have two types of external frame memory: 2 Mbyte frame buffer and 128 kbyte refresh memory. The chr signal controls which memory is accessed.

Each screen has its own memory width, vertical display width, and character/graphic attribution 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. Figure 4 illustrates the relation between the frame memory and the display screens, while Figure 5 illustrates the timing.

Note that display width of registers marked with an (\*) in Figure 4 is:

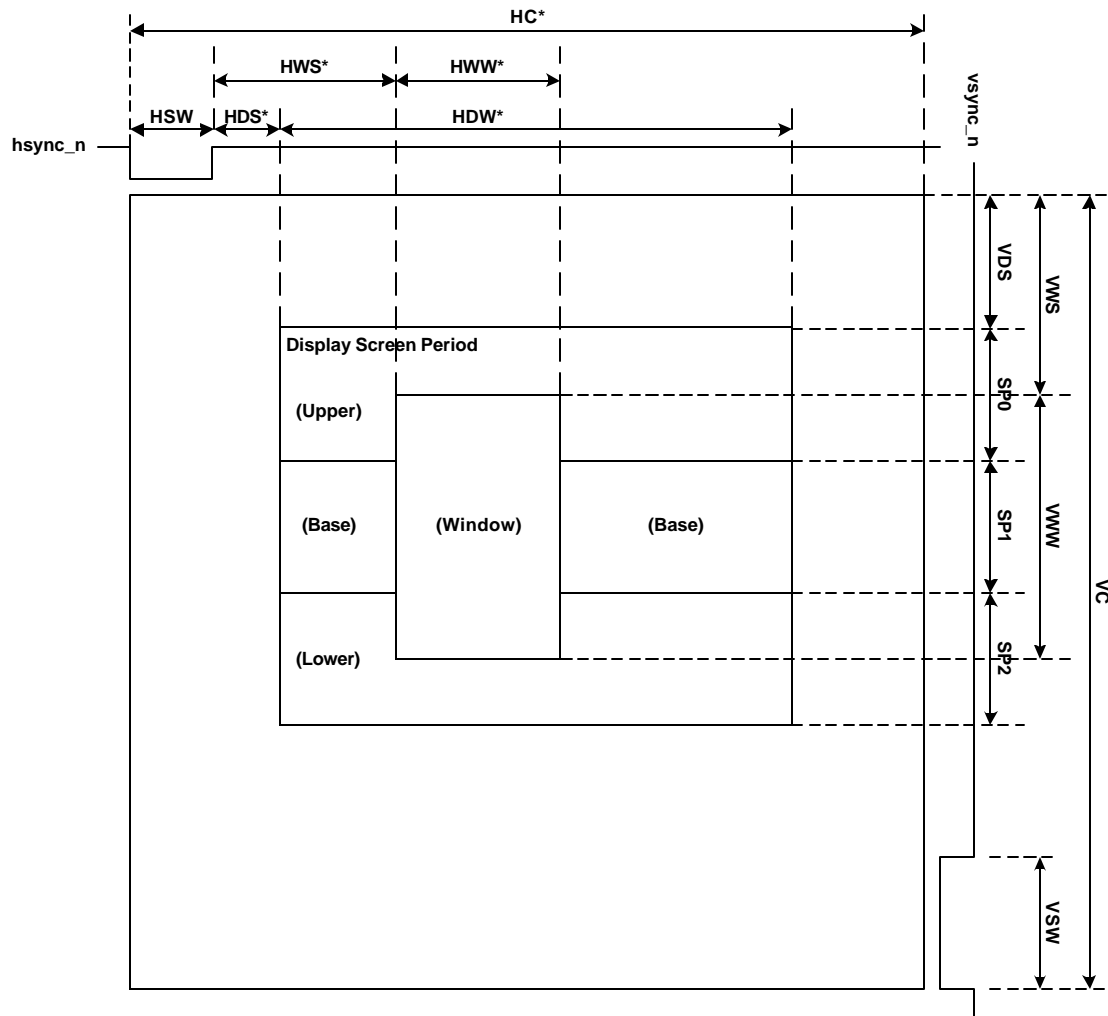
$$\text{Display width} = \text{Register value} + 1 \text{ memory cycle.}$$

**Figure 4: Frame Memory and Display Screens**





**Figure 5: Display Screen Specification**



**Timing Processor:**

The Timing Processor generates the CRT synchronization signals and signals used internally by the ACRTC. The details for this block are contained in the module specification for the Display Processor.

**CRT Interface:**

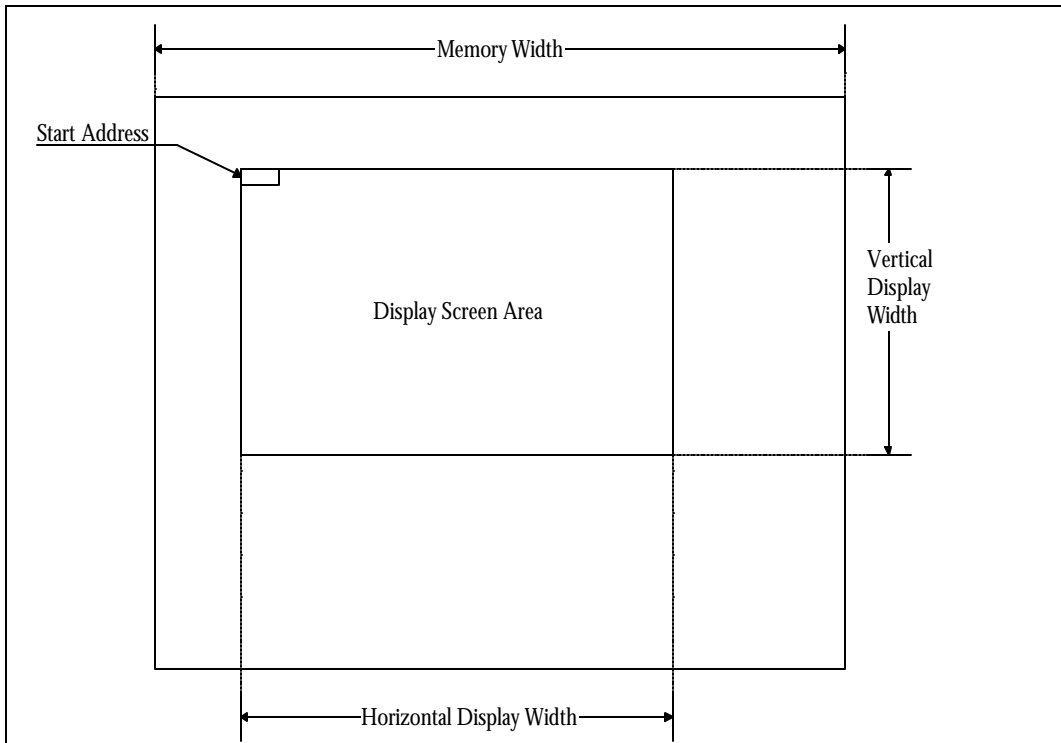
The CRT Interface manages the communication between the frame buffer, the light pen and the CRT. The frame buffer interface manages the frame buffer bus and selects display drawing or refreshes address outputs. The light pen interface uses a 20-bit address register and a strobe input pin (lpstb).

**Frame Buffer Interface:**

The ACRTC allows for two types of independent frame memories. The first type is up to a 2 Mbyte frame buffer and the second is a 128 Kbytes refresh memory. The chr output pin can access either the Graphic or Character screen.

The width of the frame memory is defined by setting-up the memory width register (mwr) and independently, the horizontal display width is defined by the horizontal display register (hdr). This allows for the frame buffer area to be bigger than the display area; reference Figure 6.

**Figure 6: Frame Memory and Display Screen Area**



The ACRTC has two ways to access the frame memory (or buffer); (1) Display Memory Access (three types) and (2) Graphic Address Increment mode.

**Display Memory Access Modes:**

In Single Access Mode, a display or drawing cycle is defined as two cycles of clk\_2. During the first cycle, the frame buffer display or drawing address is output. During the second clk\_2 cycle, the frame buffer data is read (display cycles and/or drawing cycles) or written (drawing cycles).

Display and drawing cycles contend for access to the frame buffer. The ACRTC allows the priority to be defined as display priority or drawing priority. If display has priority, drawing cycles are only allowed to occur during the horizontal or vertical fly back periods (a 'flash less' display is obtained). If drawing has priority, drawing may occur during display (display may flash).

In Interleaved Access Mode (dual access mode 0), display cycles and drawing cycles are interleaved. A display or drawing cycle is defined as four cycles of clk\_2.

- During the first clk\_2 cycle, the ACRTC outputs the frame buffer display address.
- During the second clk\_2 cycle, the display data is output from the frame buffer.
- During the third clk\_2, the ACRTC outputs the frame buffer drawing address.
- During the fourth clk\_2 cycle, the ACRTC reads or writes the drawing data.

In Superimposed Access Mode (dual access mode 1), two separate logical screens are accessed during each display cycle. The display cycle is defined as four clk\_2 cycles. If the third and fourth cycles are not used for window display, they can be used for drawing; similar to the Interleaved Mode.

- During the first clk\_2 cycle, the ACRTC outputs the background screen frame buffer address.
- During the second clk\_2 cycle, the background screen displays data.
- During the third clk\_2 cycle, the ACRTC outputs the window screen frame buffer address or the drawing frame buffer address.
- During the fourth clk\_2 cycle, the ACRTC reads (display or drawing) or writes (drawing) the window screen display or drawing data.

**Graphic Address Increment (GAI) Mode:**

The ACRTC can be programmed to control the graphic display address in one of six ways, by incrementing by 1, 2, 4, 8, and 16 words, 1 word every two display cycles, and no increment. Setting GAI to increment by 2, 4, 8, or 16 words per display cycle achieves 2, 4, 8, or 16 times the video data rate corresponding to GAI = 1. This allows the number of bits/logical pixel and logical pixel resolution to be increased while meeting the clk\_2 maximum frequency constraint.

When the frame buffer memory uses dynamic RAMs (DRAMs), the ACRTC automatically provides DRAM refresh addressing.

During hsync\_n low, the ACRTC outputs the values of an 8-bit DRAM refresh counter on the multiplexed frame buffer address and data bus mad[15:0]. The counter is decremented on each frame buffer access. The refresh address pin assignment (mad[15:0]) depends on the GAI mode. The remaining mad and ma19\_16\_ra outputs not used for refresh addressing are cleared to a low value.

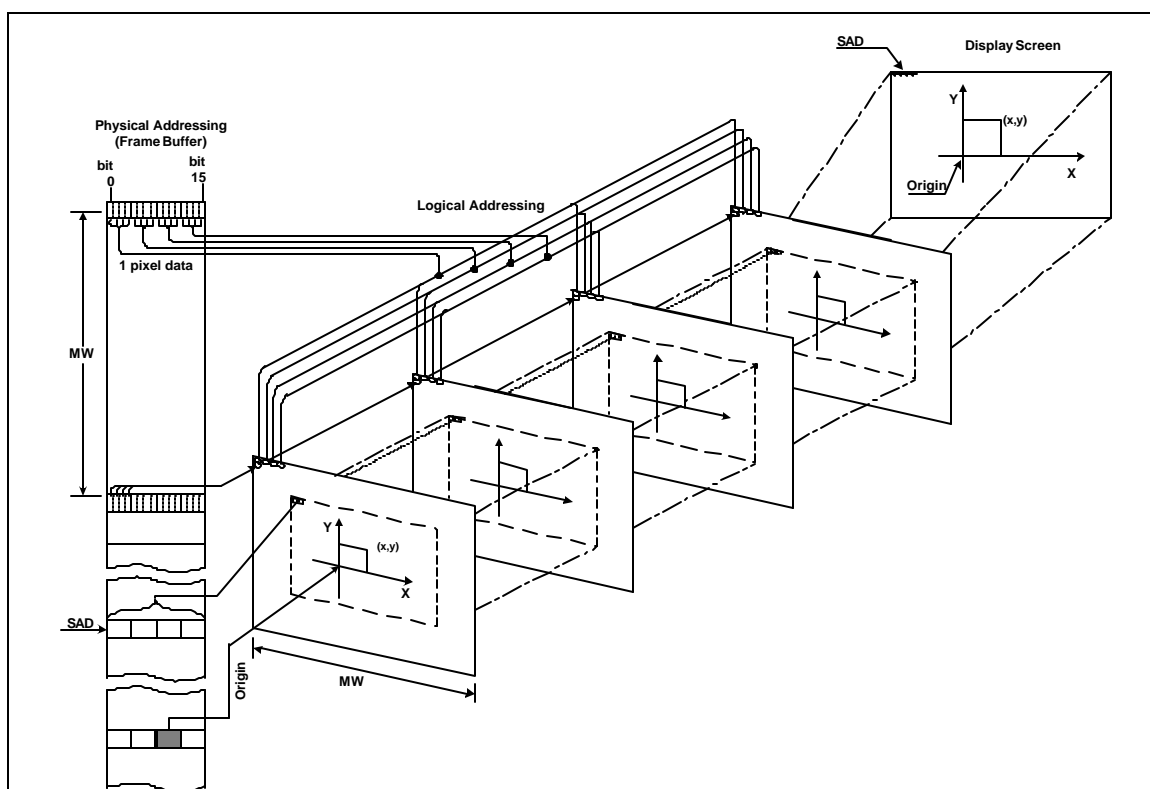
**Table 1: GAI and DRAM Refresh Addressing**

Address Increment Mode	Refresh Address Output Terminal
+1 (GAI = 000)	mad[7:0]
+2 (GAI = 001)	mad[8:1]
+4 (GAI = 010)	mad[9:2]
+8 (GAI = 011)	mad[10:3]
+16 (GAI = 100)	mad[11:4]
+0 (GAI = 101)	mad[7:0]
+1/2 (GAI = 11X)	mad[7:0]

## Address Space:

The ACRTC allows the host to issue commands in logical X-Y coordinates. The ACRTC then converts the physical linear word addresses with bit field offsets in the frame buffer. Figure 7 shows the relationship between the logical X-Y screen address and the frame buffer memory. The frame buffer memory is organized as sequential 16 bit words. The host may specify 1, 2, 4, 8, or 16 physical bits in the frame buffer. The system in the figure uses 4 bit logical pixels, allowing for 16 colors or tones.

**Figure 7: Logical/Physical Addressing**



Up to 4 logical screens may be mapped onto the ACRTC physical address space. The four screens are the upper, base, lower, and window screens. The host first specifies the following:

- A logical screen starting address.
- A logical screen physical memory width (memory words per raster).
- A logical pixel physical memory width (bit per pixel).
- A logical origin physical address.

Then the ACRTC converts the logical pixel X-Y addresses issued by the host MPU or the drawing processor to physical frame buffer addresses. The device also performs bit extraction and masking to map logical pixel operations to 16 bit word frame buffer addresses.

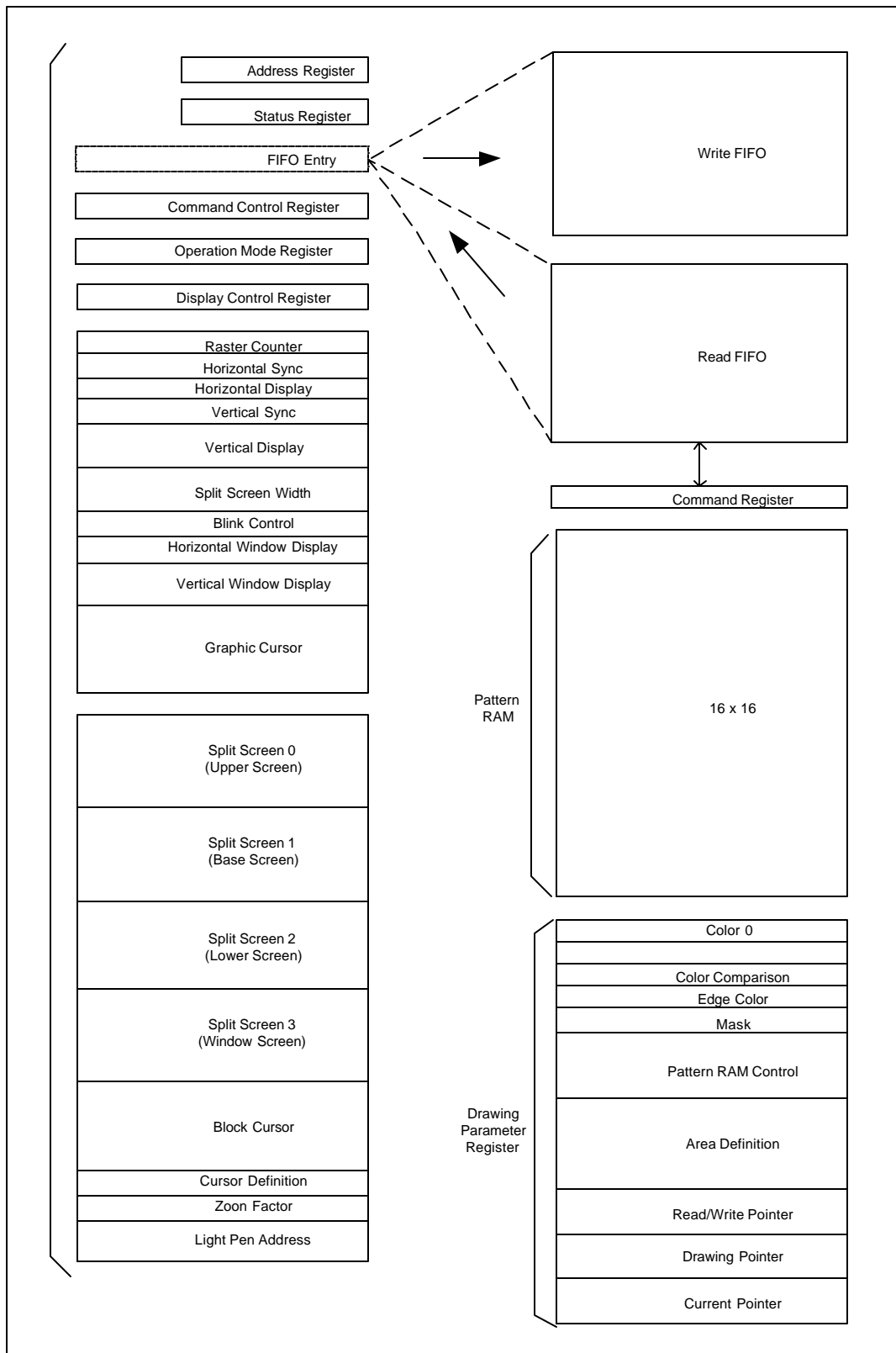
### Memory Map:

The ACTRC has over 200 bytes of accessible registers organized as Hardware, Direct, and FIFO Access. Figure 8 illustrates the programming memory map model.

- The ACRTC registers are initialized by res\_n as follows:
- Drawing and display operations are stopped
- Status register (SR) is initialized to \$FF23
- Command control register (CCR) is initialized to \$8000.
- Operation mode register bits MS and STR are reset to 0.
- All other registers are unaffected by res\_n.
- The FIFO Entry (FE) pointer is cleared, and the written command/parameter and the read data are lost.
- The DRAM refresh address is placed on the mad lines determined by graphic address increment (GAI). Refresh continues to function until the start bit (STR) is set to 1. hsync\_n is also held low during the period from res\_n until str is set by the MPU.

For directly accessible registers, the register address is shown as 'rXX', and FIFO accessible registers are shown as 'PrXX', where XX is interpreted as an 8 bit hexadecimal value. Hexadecimal numbers are denoted by a leading '\$'.

**Figure 8: Programming Model**



### 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. When  $rs=0$ , 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 into the ACRTC address register.

### Direct Access:

The MPU accesses the direct access registers by loading the register address into the address register. Then, when the MPU accesses the ACRTC with  $rs=1$ , 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 ( $hsync_n$ ,  $vsync_n$ , etc.), logical display screen size and display period, and blink period.

The display control RAM contains registers that define logical screen display parameters, such as start address, raster address, and memory width. It also includes the cursor 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. Figure and Figure illustrate the hardware and direct access register information.

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. Figure illustrates the drawing parameter registers.

**Figure 9: Hardware Access and Direct Access Registers**

Reg Name	Reg #	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	CS_n, IS, rw_n
Address Reg(AR)	AR	\$0								Address								0, 0, 0
Status Reg(SR)	ST	\$0								CER	ARD	CED	LPD	RFF	RFR	WFR	WFE	0, 0, 0
FIFO Entry(FE)	\$00	FIFO Entry																0, 1, 0/1
Command Control (CCR)	\$02	ABT	PSE	DDM	CDM	DRC	GBM			CRE	ARE	CEE	LPE	RFE	RRE	WRE	WEE	0, 1, 0/1
Operation Mode (OMR)	\$04	MS	STR	ACP	WSS	CSK		DSK	RAM	GAI			ACM		RSM		0, 1, 0/1	
Display Control (DCR)	\$06	DSP	SE1	SE0		SE2	SE3		ATR								0, 1, 0/1	
Undefined	\$08-\$7E, \$9E-\$BE, \$F0-\$FE	\$0																0, 1, 0/1
Raster Count(RCR)	\$80	\$0				RC												0, 1, 1
Horizontal Sync(HSR)	\$82	HC								\$0				HSW				0, 1, 0/1
Horizontal Display (HDR)	\$84	HDS								HDW								0, 1, 0/1
Vertical Sync(VSR)	\$86	\$0				VC												0, 1, 0/1
Vertical Display (VDR)	\$88	VDS								\$0				VSW				0, 1, 0/1
Split Screen Width(SSW)	\$8A	\$0				SP1												0, 1, 0/1
	\$8C	\$0				SP0												0, 1, 0/1
	\$8E	\$0				SP2												0, 1, 0/1
Blink Control (BCR)	\$90	BON1				BOFF1				BON2				BOFF2				0, 1, 0/1
Horz. Window Disp(HWR)	\$92	HWS								HWW								0, 1, 0/1
Vert. Window Disp(VDR)	\$94	\$0				VWS												0, 1, 0/1
	\$96	\$0				VWW												0, 1, 0/1
Graphic Cursor (GCR)	\$98	CXE								CXS								0, 1, 0/1
	\$9A	\$0				CSY												0, 1, 0/1
	\$9C	\$0				CYE												0, 1, 0/1



# IA63484 Advanced CRT Controller

# Data Sheet

Figure 10: Hardware Access and Direct Access Registers (cont.)

Reg Name	Reg #	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	cs_n, rs, rw_n
Raster Addr 0 (RAR0)	SC0(Upper Scm)	\$0			LRA0				\$0			FRA0				0, 0, 0		
Memry Wdth 0 (MWR0)	SC2(Upper Scm)	CH R	\$0			MW0											0, 0, 0	
Strt Addr 0 (SAR0)	SC4(Upper Scm)	\$0			SDA0			\$0			SA0H/SRA0				0, 1, 0/1			
	SC6(Upper Scm)	SA0L															0, 1, 0/1	
Raster Addr 1 (RAR1)	SC8(Base Scm)	\$0			LRA1				\$0			FRA1				0, 1, 0/1		
Mem Width 1 (MWR1)	SCA(Base Scm)	CH R	\$0			MW1											0, 1, 0/1	
Strt Addr 1 (SAR1)	SCC(Base Scm)	\$0			SDA1			\$0			SA1H/SRA1				0, 1, 0/1			
	SCE(Base Scm)	SA1L															0, 1, 1	
Raster Addr 2 (RAR2)	SD0(Lower Scm)	\$0			LRA2				\$0			FRA2				0, 1, 0/1		
Memry Wdth 2 (MWR2)	SD2(Lower Scm)	CH R	\$0			MW2											0, 1, 0/1	
Strt Addr 2 (SAR2)	SD4(Lower Scm)	\$0			SDA2			\$0			SA2H/SRA2				0, 1, 0/1			
	SD6(Lower Scm)	SA2L															0, 1, 0/1	
Raster Addr 3 (RAR3)	SD8(Wndw Scm)	\$0			LRA3				\$0			FRA3				0, 1, 0/1		
Memry Wdth 3 (MWR3)	SDA(Wndw Scm)	CH R	\$0			MW3											0, 1, 0/1	
Strt Addr 3 (SAR3)	SDC(Wndw Scm)	\$0			SDA3			\$0			SA3H/SRA3				0, 1, 0/1			
	SDE(Wndw Scm)	SA3L															0, 1, 0/1	
Blk Cursor 1 (BCUR1)	SE0	BCW1		BCSR1				\$0			BCER1				0, 1, 0/1			
	SE2	BCA1															0, 1, 0/1	
Blk Cursor 2 (BCUR2)	SE4	BCW2		BCSR2				\$0			BCER2				0, 1, 0/1			
	SE5	BCA2															0, 1, 0/1	
Cursor Def. (CDR)	SE8	CM	CON1		COFF1			\$0			CON2		COFF2		0, 1, 0/1			
Zoom Factor (ZFR)	SEA	HZF			VZF				\$0						0, 1, 0/1			
Lightpen Addr (LPAR)	SEC	\$0							CH R	\$0			FRA3				0, 1, 0/1	
	SEE	LPAL															0, 1, 0/1	

# IA63484 Advanced CRT Controller

# Data Sheet

**Figure 11: Drawing Parameter Registers**

Reg Name	Reg #	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Read/Write
Color 0 (CL0)	Pr00	CL0																R/W
Color1 (CL1)	Pr01	CL1																R/W
Color Cmpr (CCMP)	Pr02	CCMP																R/W
Edge Color (EDG)	Pr03	EDG																R/W
Mask (MASK)	Pr04	MASK																R/W
Pattern RAM Control (PRC)	Pr05	PPY			PZCY			PPX			PZCX						R/W	
	Pr06	PSY			S0			PSX			S0						R/W	
	Pr07	PEY			PZY			PEX			PZX						R/W	
Area Def(ADR)-> Set 2's Comp. for neg. values of X and Y axis.	Pr08	XMIN																R/W
	Pr09	YMIN																R/W
	Pr0A	XMAX																R/W
	Pr0C	YMAX																R/W
Read Write Pntr (RWP)	Pr0C	DN	S0					RWPH										R/W
	Pr0D	RWPL										S0					R/W	
Undefined	Pr0E-Pr0F, Pr14-Pr15	S0																R/W
Drawing Pntr (DP)	Pr10	DN	S0					DPAH										R
	Pr11	DPAL										DPD					R	
Current Pntr(CP)-> Set 2's Comp. for neg. values	Pr12	X																R
	Pr13	Y																R

---

**COMMAND TRANSFER MODES:**

Program Transfer and DMA Transfer are the two modes used to transfer commands and associated parameters issued by the MPU to the ACRTC.

**Program Transfer:**

Program transfer occurs when the MPU specifies the FIFO entry address and then writes operation code/parameters to the write FIFO under program control. The MPU writes are normally synchronized with ACRTC FIFO status by software polling or interrupts.

**Software Polling (WFR, WFE interrupts disabled):**

- MPU program checks the SR for WFR=1, and then writes 1-word operation code/parameters, or
- MPU program checks the SR for write WFE=1, and the writes 1- to 8-word operation code/parameters.

**Interrupt Driven (WFR, WFE interrupts enabled):**

- MPU WFR interrupt service routine writes 1-word operation code/parameters, or
- MPU WFE interrupt service routine writes 1- to 8-word operation code/parameters.

**DMA Transfer:**

Commands and parameters can be transferred from MPU system memory by an external DMAC. The MPU initiates and terminates command DMA transfer mode under software control. Command DMA can also be terminated by assertion of the done\_n input.

Using command DMA transfer, the ACRTC will issue cycle stealing DMA requests to the DMAC when the write FIFO is empty. The DMA data is automatically sent from system memory to the ACRTC write FIFO regardless of the contents of the address register.

**Command Function:**

The ACRTC commands are divided into three groups, register access commands, data transfer commands, and graphic drawing commands.

**Register access commands:**

Access to the drawing processor drawing parameter registers and the pattern RAM is through the read/write FIFOs using register access commands. When writing register access commands to an initially 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.

**Data transfer commands:**

Data is moved between the host system memory and the frame buffer, or within the frame buffer using the data transfer commands. Before issuing these commands, a physical 20-bit frame buffer address must be specified in the RWP (read/write pointer) drawing parameter register.

**Graphic Drawing Commands:**

The graphic drawing commands cause the ACRTC to draw. Graphic drawing is performed by modifying the contents of the frame buffer based on micro coded drawing algorithms in the ACRTC drawing processor. Parameters for these commands are specified using logical X-Y addressing. The display processor performs the complex task of translating a logical pixel address to a linear frame buffer word address, and further, selecting the proper sub field of the word.

Many instructions allow specification in either absolute or relative X-Y coordinates. In both cases, two's complement numbers represent both positive and negative values.

Table 2 and Table 3 tabulate the ACRTC drawing commands and Op-Codes available.

**Table 2: ACRTC Command Table**

Type	Mnemonic	Command Name	# (words)	CLK_2 Cycles
Register Access Command	ORG	Origin	3	8
	WPR	Write Parameter Reg	2	6
	RPR	Read Parameter Reg	1	6
	WPTN	Write Pattern RAM	n+2	4n+8
	RPTN	Read Pattern RAM	2	4n+10
Data Transfer Command	DRD	DMA Read	3	$(4x+8)y+12(x*y/8) \uparrow (62-68)$
	DWT	DMA Write	3	$(4x+8)y+16(x*y/8) \uparrow +34$
	DMOD	DMA Modify	3	$(4x+8)y+16(x*y/8) \uparrow +34$
	RD	Read	1	12
	WT	Write	2	8
	MOD	Modify	2	8
	CLR	Clear	4	$(2x+8)y+12$
	SCLR	Selective Clear	4	$(4x+8)y+12$
	CPY	Copy	5	$(6x+8)y+12$
SCPY	Selective Copy	5	$(6x+8)y+12$	
Graphic Drawing Command	AMOVE	Absolute Move	3	56
	RMOVE	Relative Move	3	56
	ALINE	Absolute Line	3	$P*L+18$
	RLINE	Relative Line	3	$P*L+18$
	ARCT	Absolute Rectangle	3	$2P(A+B)+54$
	RRCT	Relative Rectangle	3	$2P(A+B)+54$
	APLL	Absolute Polyline	$2n+2$	$\sum (P * L + 16) + 8$
	RPLL	Relative Polyline	$2n+2$	$\sum (P * L + 16) + 8$
	APLG	Absolute Polygon	$2n+2$	$\sum (P * L + 16) + P * L_o + 20$
	RPLG	Relative Polygon	$2n+2$	$\sum (P * L + 16) + P * L_o + 20$
	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	$(P*A+8)B+18$
	RFRC	Relative Filled Rectangle	3	$(P*A+8)B+18$
	PAINT	Paint	1	$(18A+102)B-58$ (Applies to rectangular figures, varies for other shapes)
	DOT	Dot	1	8
	PTN	Pattern	2	$(P*A+10)B+20$
	AGCPY	Absolute Graphic Copy	5	$((P+2)A+10)B+70$
RGCPY	Relative Graphic Copy	5	$((P+2)A+10)B+70$	

**Table 3: Opcode Map**

Type	Mnemonic	Operation Code	Parameter					
Register Access Command	ORG	0000010000000000	DPH DPL					
	WPR	000010000000RN	D					
	RPR	000011000000RN						
	WPTN	000110000000 PRA	n	D1,...,Dn				
	RPTN	000111000000 PRA	n					
Data Transfer Command	DRD	0010010000000000	AX	AY				
	DWT	0010100000000000	AX	AY				
	DMOD	00101100000000MM	AX	AY				
	RD	0100010000000000						
	WT	0100100000000000	D					
	MOD	01001100000000MM	D					
	CLR	0101100000000000	D	AX	AY			
	SCLR	01011100000000MM	D	AX	AY			
	CPY	00110S DSD 00000000	SAH	SAL	AX	AY		
	SCPY	01111S DSD 000000MM	SAH	SAL	AX	AY		
Graphic Drawing Command	AMOVE	1000000000000000	X	Y				
	RMOVE	1000010000000000	dX	dY				
	ALINE	10001000 AREA COL OPM	X	Y				
	RLINE	10001100 AREA COL OPM	dX	dY				
	ARCT	10010000 AREA COL OPM	X	Y				
	RRCT	10010100 AREA COL OPM	dX	dY				
	APLL	10011000 AREA COL OPM	n	X1,Y1,...,XN,YN				
	RPLL	10011100 AREA COL OPM	n	dX1,dY1,...,dXN,dYN				
	APLG	10100000 AREA COL OPM	n	X1,Y1,...,XN,YN				
	RPLG	10100100 AREA COL OPM	n	dX1,dY1,...,dXN,dYN				
	CRCL	1010100C AREA COL OPM	r					
	ELPS	1010110C AREA COL OPM	a	b	DX			
	AARC	1011000C AREA COL OPM	Xc	Yc	Xe	Ye		
	RARC	1011010C AREA COL OPM	dXc	dYc	dXe	dYe		
	AEARC	1011100C AREA COL OPM	a	b	Xc	Yc	Xe	Ye
	REARC	1011110C AREA COL OPM	a	b	dXc	dYc	dXe	dYe
	AFRCT	11000000 AREA COL OPM	X	Y				
	RFRCT	11000100 AREA COL OPM	dX	dY				
	PAINT	1100100E AREA 0 0 000						
	DOT	11001100 AREA COL OPM						
	PTN	1101SL SD AREA COL OPM	SZ					
	AGCPY	1110S DSD AREA 0 0 OPM	Xs	Ys	DX	DY		
	RGCPY	1111S DSD AREA 0 0 OPM	dXs	dYs	dDX	dDY		

**AC/DC PARAMETERS:**

Absolute maximum ratings:

Operating Temp (Comm'l).....0°C to +70°C  
 Storage Temperature.....- 65°C to 150°C  
 V<sub>CC</sub> Supply Voltage..... - 0.3V to +4.6V  
 Input Voltage Range..... - 0.3V to +4.6V  
 Allowable Input Current..... TBD  
 Total Allowable Input Current.....TBD

Recommended Operating Conditions (@ 9.8 MHz):

Power Supply V<sub>CC</sub>.....4.75V to 5.25V  
 Input Low Voltage V<sub>IL</sub>.....0V to 0.8V  
 Input High Voltage V<sub>IH</sub>.....2.0V to V<sub>CC</sub>  
 Operating Temperature Range.....0°C to 70°C

**DC Characteristics:**

Item		Symbol	Min	Max	Unit	Test Conditions
Input High Level Voltage	All Inputs	V <sub>IH</sub>	2.0	-	V	9.8 MHz
Input Low Level Voltage	All Inputs	V <sub>IL</sub>	-	0.8	V	9.8 MHz
Input Leak Current	rw_n, cs_n, rs, res_n, dack_n, clk_2, lpstb	I <sub>in</sub>	-10	10	uA	V <sub>SS</sub> to V <sub>CC</sub>
Hi-Z Input Current	d[15:0], mad[15:0], exsync_n	I <sub>TSI</sub>	-10	10	uA	V <sub>SS</sub> to V <sub>CC</sub>
Output High Level Voltage	d[15:0], mad[15:0], exsync_n, cud1_n, cud2_n, dreq_n, dtack_n, hsync_n, vsync_n, mrd, draw_n, as_n, disp1_n, disp2_n, chr, mcyc, ra4, ma16/ra0, ma19/ra3	V <sub>OH</sub>	0.7V <sub>CC</sub>		uA	I <sub>OH</sub> = -4mA CMOS Output
Output Low Level Voltage	d[15:0], mad[15:0], exsync_n, cud1_n, cud2_n, dreq_n, dtack_n, hsync_n, vsync_n, mrd, draw_n, as_n, disp1_n, disp2_n, chr, mcyc, ra4, ma16/ra0, ma19/ra3	V <sub>OL</sub>		0.3V <sub>CC</sub>	V	I <sub>OL</sub> = 4mA CMOS Output
		V <sub>OL</sub>		0.3V <sub>CC</sub>	V	I <sub>OL</sub> = 4mA Open Drain
Output Leak Current(Hi-Z)	irq_n, done_n	I <sub>LOD</sub>		TBD	uA	V <sub>OH</sub> = V <sub>CC</sub>

# IA63484

## Advanced CRT Controller

# Data Sheet

Item		Symbol	Min	Max	Unit	Test Conditions
Input Capacitance	d[15:0], mad[15:0], exsync_n, rw_n, cs_n, rs, res_n, dack_n, clk_2, lpstb	C <sub>IN</sub>		TBD	pF	TBD
Output Capacitance	irq_n, done_n	C <sub>OUT</sub>		TBD	pF	TBD
Current-Consumption		I <sub>CC</sub>		TBD	mA	9.8 MHz

(V<sub>CC</sub> = 5.0V±5%, V<sub>SS</sub> = 0V, T<sub>a</sub> = 0 to 70°C, unless otherwise noted.)

## AC Characteristics:

### Clock Timing:

Item	Symbol	9.8 MHz Version		Unit
		Min	Max	
Operation Frequency of clk_2	f	1	9.8	MHz
Clock Cycle Time	t <sub>CYC</sub>	102	1000	ns
Clock High Level Pulse Width	t <sub>PWCH</sub>	46	500	ns
Clock Low Level Pulse Width	t <sub>PWCL</sub>	46	500	ns
Clock Rise Time	t <sub>cr</sub>		5	ns
Clock Fall Time	t <sub>cf</sub>		5	ns

### MPU Read / Write Cycle Timing:

Item	Symbol	9.8 MHz Version		Unit
		Min	Max	
rw_n Setup Time	t <sub>RWS</sub>	50		ns
rw_n Hold Time	t <sub>RWH</sub>	0		ns
rs Setup Time	t <sub>RSS</sub>	50		ns
rs Hold Time	t <sub>RSH</sub>	0		ns
cs_n Setup Time	t <sub>CSS</sub>	40		ns
cs_n High Level Width	t <sub>WCSH</sub>	60		ns
Read Wait Time	t <sub>RWAI</sub>	0		ns
Read Data Access Time	t <sub>RDAC</sub>		80	ns
Read Data Hold Time	t <sub>RDH</sub>	10		ns
Read Data Turn Off Time	t <sub>RDZ</sub>		60	ns
dtack_n Delay Time (Z to L)	t <sub>DTKZL</sub>		70	ns
dtack_n Delay Time (D to L)	t <sub>DTKDL</sub>	0		ns
dtack_n Release Time (L to H)	t <sub>DTKLH</sub>		80	ns
dtack_n Turn Off Time (H to Z)	t <sub>DTKZ</sub>		100	ns
Data Bus 3-State Recovery Time 1	t <sub>DBRT1</sub>	0		ns
Write Wait Time	t <sub>WWAI</sub>	0		ns
Write Data Setup Time	t <sub>WDS</sub>	40		ns
Write Data Hold Time	t <sub>WDH</sub>	10		ns



**AC Characteristics (continued):**

**DMA Read / Write Cycle Timing:**

Item	Symbol	9.8 MHz Version		Unit
		Min	Max	
dreq_n Delay Time 1	$t_{DRQD1}$		110	ns
dreq_n Delay Time 2	$t_{DRQD2}$		70	ns
DMA r / w_n Setup Time	$t_{DMRWS}$	50		ns
DMA r / w_n Hold Time	$t_{DMRWH}$	0		ns
dack_n Setup Time	$t_{DAKS}$	40		ns
dack_n Hold Time	$t_{WDAKH}$	60		ns
DMA Read Wait Time	$t_{DRW}$	0		ns
DMA Read Data Access Time	$t_{DRDAC}$		80	ns
DMA Read Data Hold Time	$t_{DRDH}$	10		ns
DMA Read Data Turn Off Time	$t_{DRDZ}$		60	ns
DMA dtack_n Delay Time (Z to L)	$t_{DDTZL}$		70	ns
DMA dtack_n Delay Time (D to L)	$t_{DDTDL}$	0		ns
DMA dtack_n Release Time (L to H)	$t_{DDTLH}$		80	ns
DMA dtack_n Turn Off Time (H to Z)	$t_{DDTHZ}$		100	ns
done_n Output Delay Time	$t_{DND}$		70	ns
done_n Output Turn Off Time	$t_{DNL2}$		80	ns
Data Bus 3-State Recovery Time 2	$t_{DBRT2}$	0		ns
done_n Input Pulse Width	$t_{DNPW}$	2		$t_{CYC}$
DMA Write Wait Time	$t_{DWW}$	0		ns
DMA Write Data Setup Time	$t_{DWDS}$	40		ns
DMA Write Data Hold Time	$t_{DWDH}$	10		ns

**AC Characteristics (continued):**

**Frame Memory Read / Write Cycle Timing:**

Item	Symbol	9.8 MHz Version		Unit
		Min	Max	
as_n "Low" Level Pulse Width	$t_{PWASL}$	20		ns
Memory Address Hold Time 2	$t_{MAH2}$	5		ns
as_n Delay Time 1	$t_{ASD1}$		50	ns
as_n Delay Time 2	$t_{ASD2}$	5	40	ns
Memory Address Delay Time	$t_{MAD}$	10	50	ns
Memory Address Hold Time 1	$t_{MAH1}$	15		ns
Memory Address Turn Off Time (A to Z)	$t_{MAAZ}$		35	ns
Memory Read Data Setup Time	$t_{MRDS}$	30		ns
Memory Read Data Hold Time	$t_{MRDH}$	0		ns
ma_ra Delay Time	$t_{MARAD}$		60	ns
ma_ra Delay Time	$t_{MARAH}$	5		ns
MCYC Delay Time	$t_{MCYCD}$	5	40	ns
mrd Delay Time	$t_{MRDD}$		50	ns
mrd Hold Time	$t_{MRH}$	5		ns
draw_n Delay Time	$t_{DRWD}$		50	ns
draw_n Hold Time	$t_{DRWH}$	5		ns
Memory Write Data Delay Time	$t_{MWDD}$		50	ns
Memory Write Data Hold Time	$t_{MWDH}$	5		ns
Memory Address Setup Time 1	$t_{MAS1}$	10		ns
Memory Address Setup Time 2	$t_{MAS2}$	10		ns

NOTE:  $t_{MAD}$  is independent of clk\_2 operation frequency (f) and timing of  $t_{ASD2}$  and  $t_{MAS1}$

### AC Characteristics (continued):

#### Display Control Signal Output Timing:

Item	Symbol	9.8 MHz Version		Unit
		Min	Max	
hsync_n Delay Time	$t_{HSD}$		50	ns
vsync_n Delay Time	$t_{VSD}$		50	ns
disp1_n, disp2_n Delay Time	$t_{DSPD}$		50	ns
cud1_n, cud2_n Delay Time	$t_{CUDD}$		50	ns
exsync_n Output Delay Time	$t_{EXD}$	15	50	ns
chr delay time	$t_{CHD}$		50	ns

#### exsync\_n Input Timing:

Item	Symbol	9.8 MHz Version		Unit
		Min	Max	
exsync_n Input Pulse Width	$t_{EXSW}$	3		$t_{CYC}$
exsync_n Input Setup Time	$t_{EXS}$	30		ns
exsync_n Input Hold Time	$t_{EXH}$	10		ns

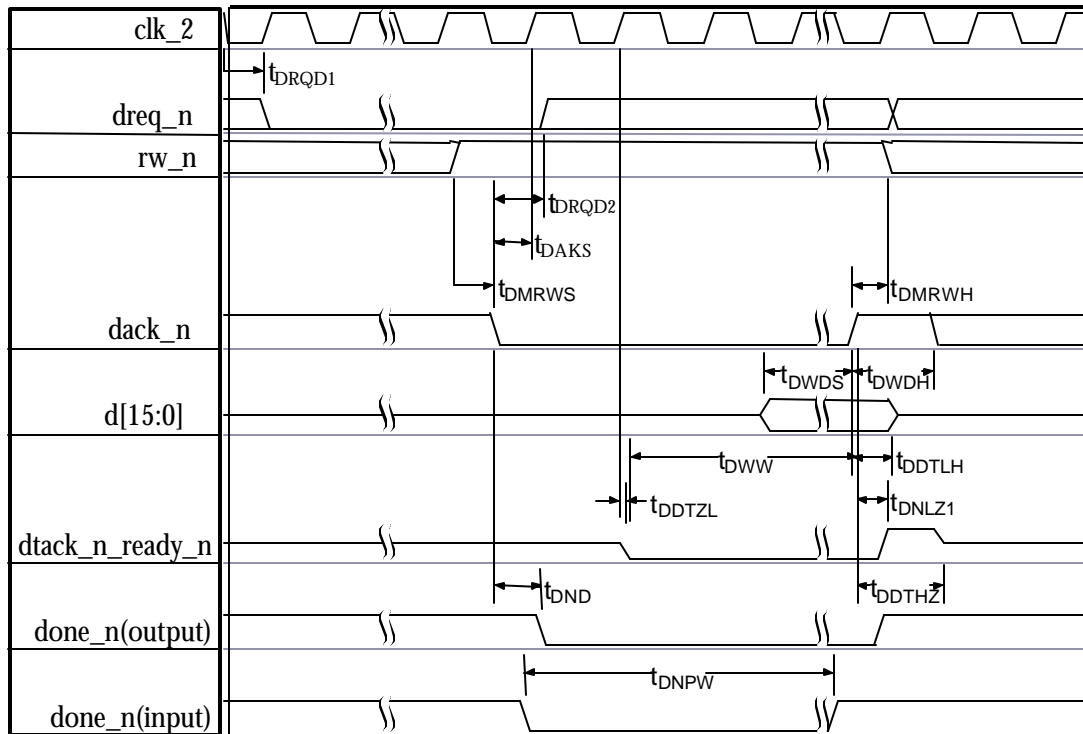
#### lpstb Input Timing:

Item	Symbol	9.8 MHz Version		Unit
		Min	Max	
lpstb Uncertain Time 1	$t_{LPD1}$	45		ns
lpstb Uncertain Time 2	$t_{LPD2}$	10		ns
lpstb Input Hold Time	$t_{LPH}$	10		ns
lpstb Input Inhibit Time	$t_{LPI}$	4		$t_{CYC}$

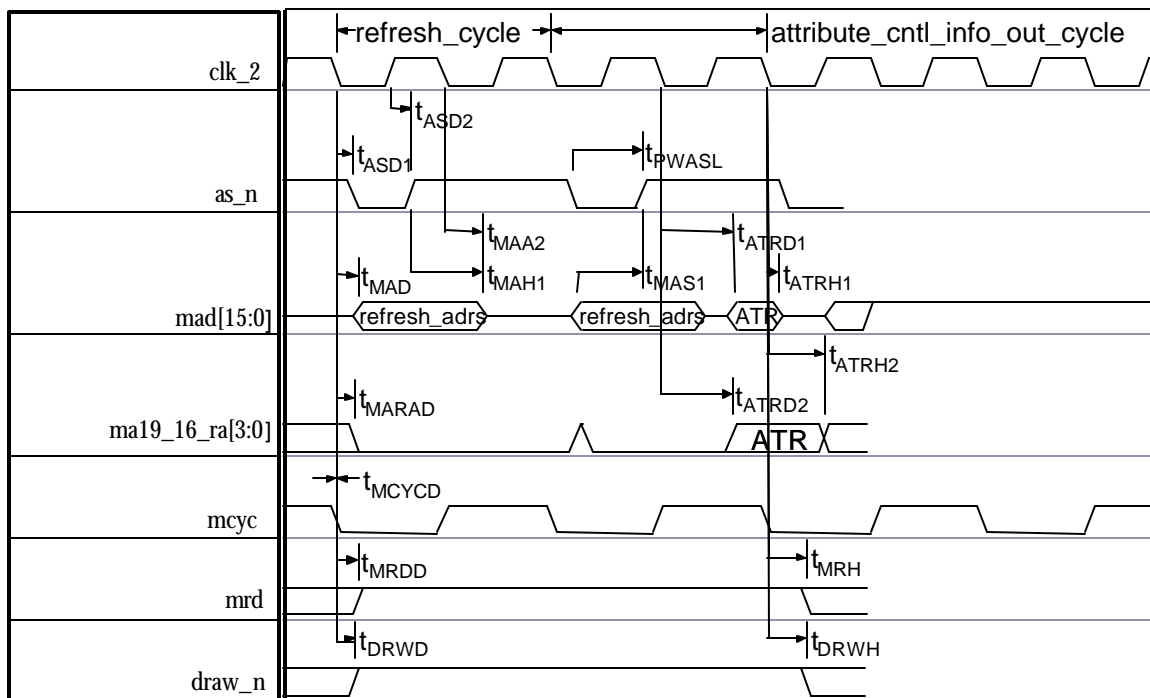
#### res\_n and dack\_n Input Timing:

Item	Symbol	9.8 MHz Version		Unit
		Min	Max	
dack_n Setup Time for res_n	$t_{DAKSR}$	100		ns
dack_n Holt Time for res_n	$t_{DAKHR}$	0		ns
res_n Input Pulse Width	$t_{RES}$	10		$t_{CYC}$

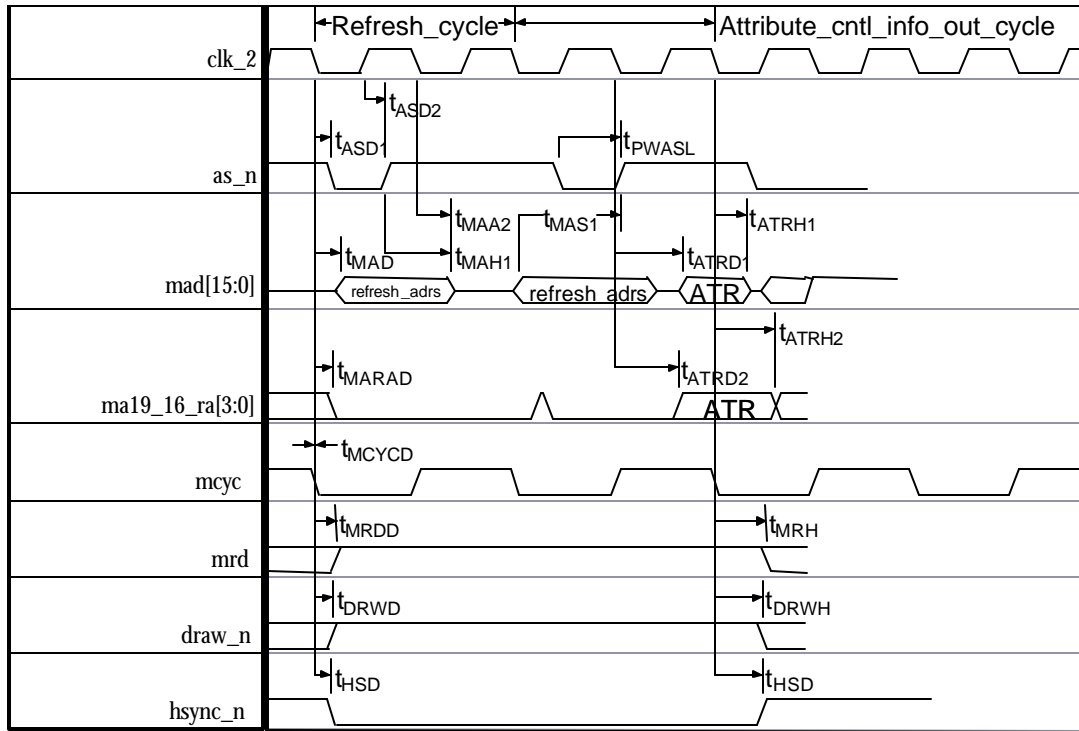
**Figure 12: DMA Write Cycle Timing**



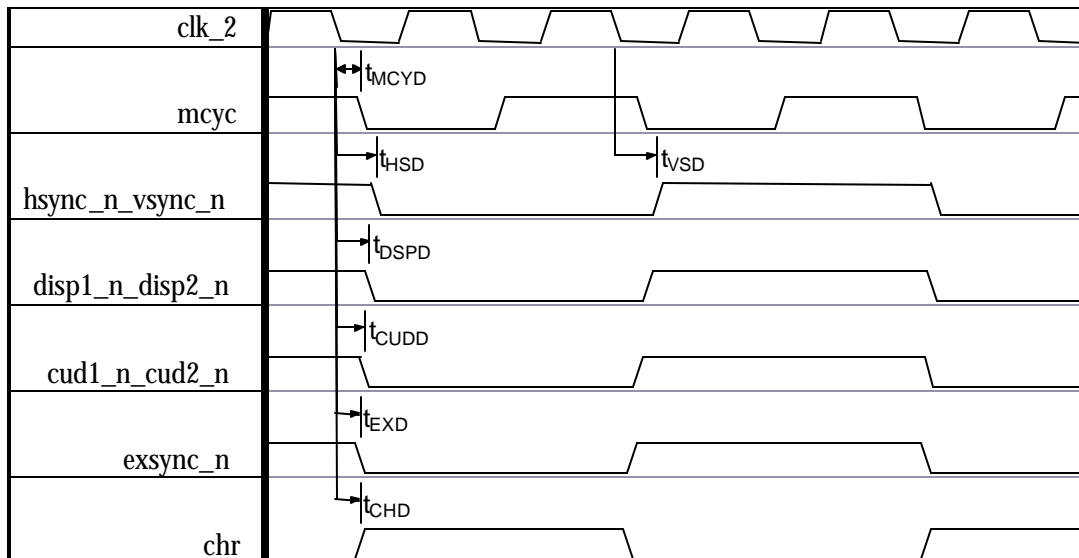
**Figure 13: Display Cycle Timing**



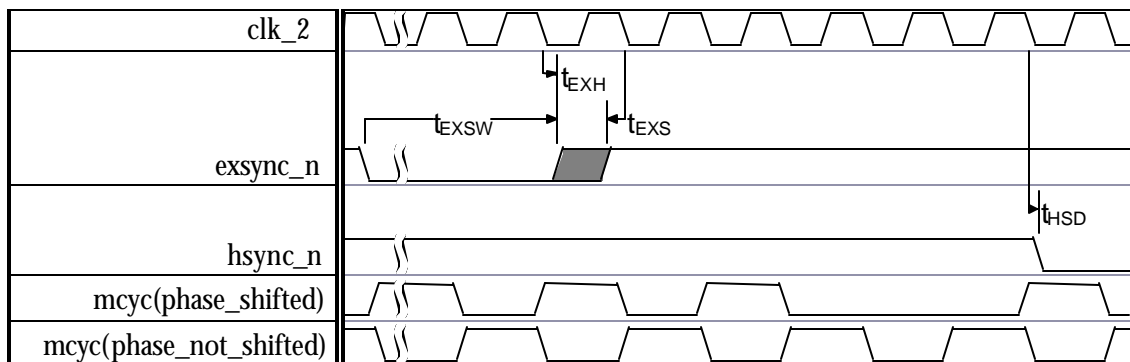
**Figure 14: Frame Memory Refresh & Video Attributes Output Cycle Timing**



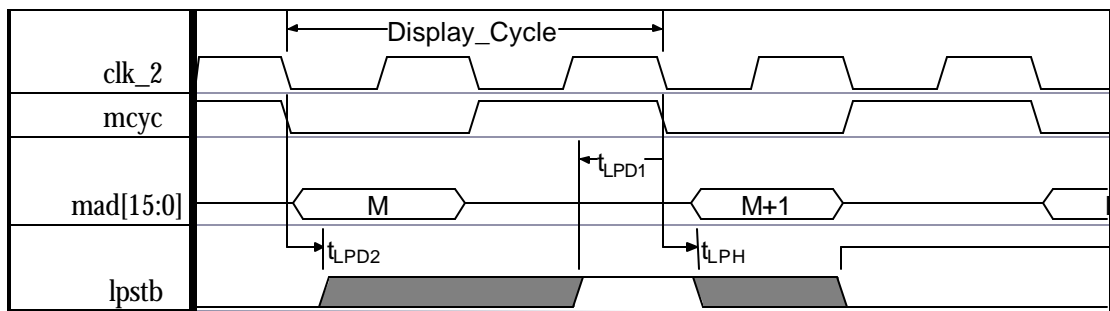
**Figure 15: Display Control Signal Output Timing**



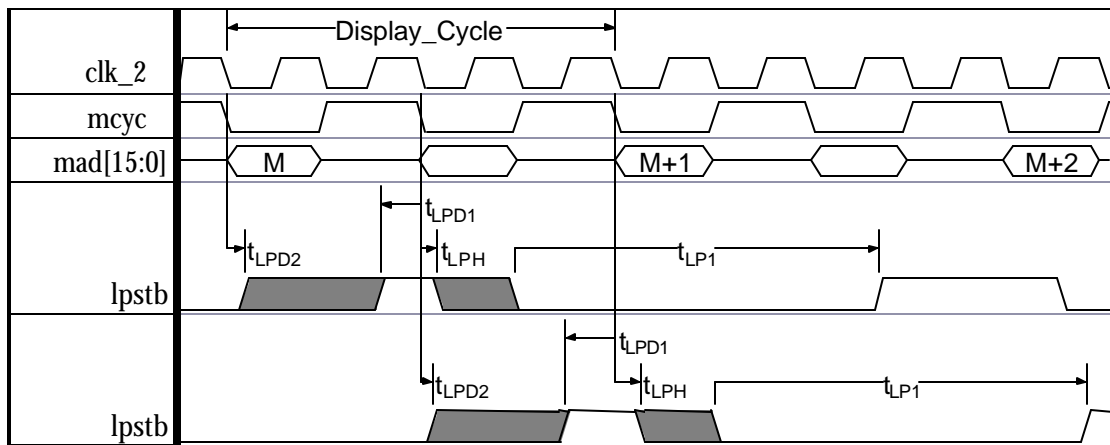
**Figure 16: Input Timing exsync\_n**



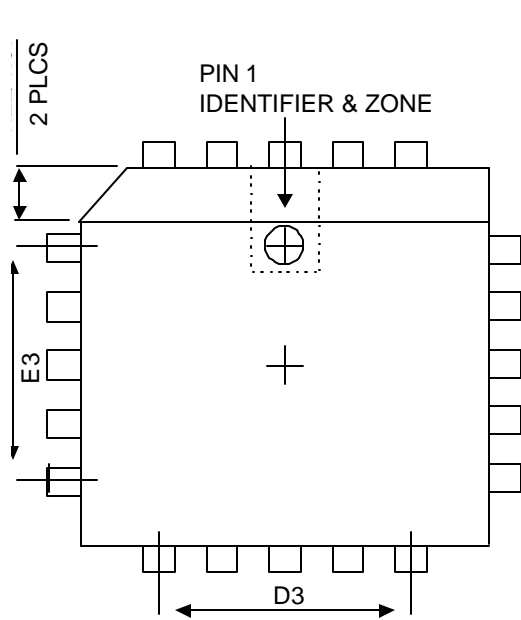
**Figure 17: Input Timing (Single Access Mode) lpstb**



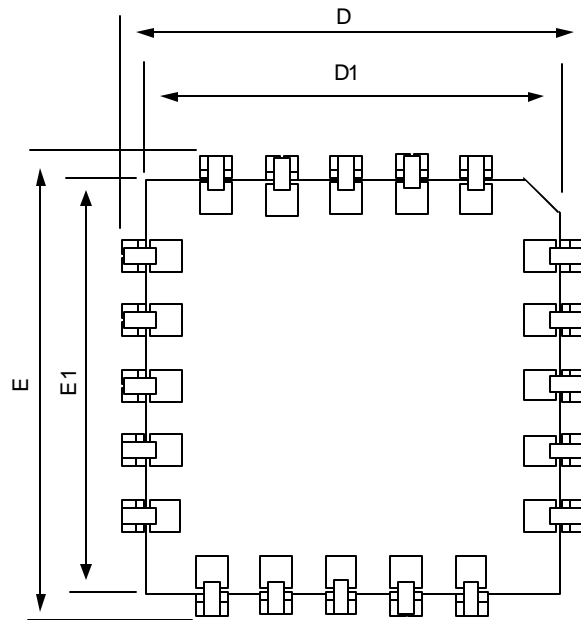
**Figure 18: Input Timing (Dual Access Mode) lpstb**



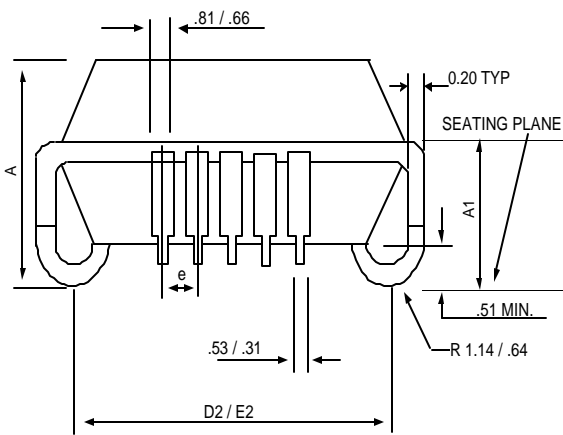
**PLCC Packaging Dimensions:**



TOP VIEW



BOTTOM VIEW



SIDE VIEW

Symbol	Lead Count = 68	
	MIN (Millimeters)	MAX (Millimeters)
A	4.20	5.08
A1	2.29	3.30
D	25.02	25.27
D1	24.13	24.33
E	25.02	25.27
E1	24.13	24.33
e	1.27 BSC	

**ORDERING INFORMATION:**

**Table 1:**

<b>Part Number</b>	<b>Temperature Grade</b>	<b>Package</b>
IA63484-PLC68I	Industrial	68 lead Plastic Leaded Chip Carrier (PLCC)

Contact **innovASIC** for other package and processing options.