

**MOTOROLA
SEMICONDUCTOR
TECHNICAL DATA**

MC68487

Advance Information

Raster Memory Controller (RMC)

PRELIMINARY

JUNE, 1985

This document contains information on a new product. Specifications and information here are subject to change without notice.

COPYRIGHT MOTOROLA INC., 1985

TABLE OF CONTENTS

<u>Paragraph Number</u>	<u>Title</u>	<u>Page Number</u>
	Section 1 Introduction	
1.1	Key Features	1-1
1.2	Discussion of Terms	1-2
1.3	General Description	1-5
	Section 2 Signal Description	
2.1	Clocks	2-2
2.1.1	Video Timing Clock (VTCLK)	2-2
2.1.2	Picture Element Clock (PCLK)	2-2
2.1.3	Memory Timing Clock (MTCLK)	2-3
2.1.4	Horizontal Sync (HSYNC)	2-3
2.2	Address/Data Buses (X, A, B, S, and Z)	2-3
2.2.1	Address Bus X (X0-X9)	2-3
2.2.2	Data Buses A and B	2-11
2.2.2.1	Data Bus A (A0-A7)	2-11
2.2.2.2	Data Bus B (B0-B7)	2-11
2.2.3	S and Z Buses	2-11
2.3	Memory Handshake	2-12
2.3.1	Address Select (ADSEL)	2-12
2.3.2	Data Bus Enable (DBEN)	2-12
2.3.3	Composite CAS Strobe (CASTB)	2-12
2.3.4	Read/Write (R/W)	2-12
2.4	MPU Interrupts	2-13
2.4.1	Real-Time Input (RTI)	2-13
2.4.2	Interrupt (INT)	2-13
2.5	Video Outputs	2-13
2.5.1	Red, Green, and Blue (R, G, B)	2-13
2.5.2	Video Enable (VIDEN)	2-14
2.6	Sync (SYNC)	2-14
2.7	Reset Enable (REN)	2-14
2.8	Test Points (T1, T2)	2-15
2.9	VCC and GND	2-15

TABLE OF CONTENTS (Continued)

<u>Paragraph Number</u>	<u>Title</u>	<u>Page Number</u>
Section 3 Reset and Initialization		
3.1	General	3-1
3.2	Video Reset	3-3
Section 4 RMS Microprocessor Interface		
Section 5 Register Description		
5.1	RMS Register Map	5-2
5.2	RMC Control Registers	5-15
5.2.1	\$FFE00 - Memory Map	5-15
5.2.2	\$FFE01 - Display Data Mode	5-15
5.2.3	\$FFE02 - Interrupt Control/Status	5-16
5.2.4	\$FFE03 - Border Color	5-17
5.2.5	\$FFE04 - Object Available	5-17
5.2.6	\$FFE05 - Paging	5-18
5.2.7	\$FFE06 - Page Independent Blocks (PIB)	5-18
5.2.8	\$FFE07 - Vertical Scroll	5-19
5.2.9	\$FFE08 - Horizontal Scroll	5-19
5.2.10	\$FFEOA, \$FFE0B - DRC Image Table Start Address	5-19
5.2.11	\$FFE0C, \$FFE0D - True Object Image Table Start Address	5-20
5.2.12	\$FFE0E, \$FFE0F - Fixed Object Image Table Start Address	5-20
5.2.13	\$FFE10-\$FFE17 - Collision Status	5-20
5.2.14	\$FFE18 - Collision Enable	5-20
5.2.15	\$FFE1C-\$FFE1F - Real-Time Output	5-20
5.2.16	\$FFE20-\$FFE23 - Real-Time Input	5-21
5.2.17	\$FFE24 - Memory Organization	5-21
5.2.18	\$FFE25 - Video Operation	5-22
5.2.19	\$FFE26 - Sync Mode	5-23
5.2.20	\$FFE28-\$FFE2B - Virtual Screen Start Address	5-23
5.2.21	\$FFE2C-\$FFE2F - Vertical Offset	5-23
5.2.22	\$FFE30-\$FFE33 - Horizontal Offset	5-24
5.2.23	\$FFE34-\$FFE37 - Virtual Screen Size	5-24
5.2.24	\$FFE38-\$FFE3B - Virtual Screen Width	5-24
5.2.25	\$FFE40-\$FFE7F - Color-Mapping RAM (CMR)	5-24
5.2.26	\$FFE90-\$FFEAF - True-Object Position	5-25
5.2.27	\$FFEBO-\$FFEB7 - True-Object Names	5-25

TABLE OF CONTENTS (Continued)

<u>Paragraph Number</u>	<u>Title</u>	<u>Page Number</u>
	Section 6 Display Modes	
6.1	Bit-Plane Mode	6-1
6.2	List Mode	6-1
6.2.1	List Mode 0	6-3
6.2.2	List Mode 1	6-4
6.2.3	List Mode 2	6-6
6.2.4	List Mode 3	6-8
6.2.5	List Mode 4	6-9
6.2.6	List Mode 5	6-11
6.3	Character Types	6-14
6.3.1	Alphanumeric Characters	6-15
6.3.2	Mosaic Characters	6-15
6.3.3	Redefinable Characters	6-15
6.3.4	Dynamically Redefinable Characters	6-21
6.3.5	Fixed Objects	6-21
6.3.6	Attributes	6-22
6.3.7	Foreground and Background Color	6-22
6.3.8	Color Mapping Offset	6-23
6.3.9	Flash	6-23
6.3.10	Invert	6-23
6.3.11	Underline	6-24
6.3.12	Double High	6-24
6.3.13	Double Wide	6-24
6.3.14	Color/Resolution	6-25
6.3.15	Separation	6-25
6.3.16	Priority	6-26
6.3.17	Collision Enable	6-26
6.3.18	Color Collision	6-26
6.3.19	Shading	6-27
6.4	True Objects	6-27
6.4.1	Object Position	6-28
6.4.2	Names	6-28
6.4.3	Pattern Data	6-28
6.4.3.1	Run Length	6-29
6.4.3.2	Bit-Plane Encoding	6-29
6.4.4	Last-Line Flag	6-30
6.4.5	Zoom Factors	6-30
6.4.6	Priority	6-30
6.4.7	Collision	6-31

===== MOTOROLA
MC68487 Front Matter - 3

TABLE OF CONTENTS (Concluded)

<u>Paragraph Number</u>	<u>Title</u>	<u>Page Number</u>
Section 7 Memory Organization		
7.1	Overview	7-1
7.2	Memory Requirements	7-1
7.3	System Memory Map	7-2
7.3.1	M68000 Family Map	7-4
7.3.2	MC6809E Memory Map	7-4
Section 8 Machine 2		
8.1	Machine 2 Initialization	8-3
8.1.1	Register Page 00	8-3
8.1.2	Register Page 01	8-4
8.1.3	Register Page 10	8-5
8.2	Machine 2 Display Modes	8-5
8.2.1	Alpha/Semographics	8-6
8.2.2	Semographics	8-6
8.2.3	Alphanumerics	8-8
8.3	Graphic Modes	8-10
Section 9 Electrical Specifications		
9.1	Maximum Ratings	9-1
9.2	Thermal Characteristics	9-1
9.3	Power Considerations	9-2
9.4	DC Electrical Characteristics	9-3
9.5	DRAM Timing	9-4
9.6	AC Electrical Characteristics	9-6
9.7	525-Line Timing	9-15
9.8	625-Line Timing	9-16
Section 10 Ordering Information and Mechanical Data		
10.1	Ordering Information	10-1
10.1	Pin Assignment	10-2
10.2	Package Dimensions	10-3

LIST OF ILLUSTRATIONS

<u>Figure Number</u>	<u>Title</u>	<u>Page Number</u>
1-1	Raster Memory Controller Block Diagram	1-6
2-1	Functional Diagram	2-1
2-2	MTCLK and PTCLK Timing vs HRES Mode	2-4
2-3	MC6809E X-Bus Timing for HRES 0, 1, 2, 7	2-5
2-4	MC6809E X-Bus Timing for HRES 3, 6	2-6
2-5	MC6809E X-Bus Timing for HRES 4	2-7
2-6	MC68000/MC68008 X-Bus Timing for HRES 0, 1, 2, 7	2-8
2-7	MC68000/MC68008 X-Bus Timing for HRES 3, 6	2-9
2-8	MC68000/MC68008 X-Bus Timing for HRES 4	2-10
3-1	X Bus Reset	3-2
4-1	MC6809E System Block Diagram	4-2
4-2	MC68008 System Block Diagram	4-3
4-3	MC68000 System Block Diagram	4-4
4-4	MC68000 Data Bus Logic	4-5
5-1	Memory Map	5-1
6-1	Attributes by Character and List Mode	6-14
6-2	ASCII Character Format	6-16
6-3	Character Position vs Lines per Character Row	6-19
6-4	Mosaics 4 and 6	6-20
6-5	Mosaic Separation	6-21
7-1	MC68000 Bank Organization	7-3
8-1	Semigraphic Block Patterns	8-7
8-2	Machine 2 Alphanumeric Format	8-9
9-1	Read Cycle Timing Diagram	9-9
9-2	Write Cycle Timing Diagram	9-10
9-3	Page Mode Read Cycle Timing Diagram	9-11
9-4	Page Mode Write Cycle Timing Diagram	9-12
9-5	Raster Memory System Timing Diagram	9-13
9-6	RMC Clock Relationships	9-14
9-7	PCLK Relationships	9-14

LIST OF TABLES

<u>Table Number</u>	<u>Title</u>	<u>Page Number</u>
2-1	Picture Element Clock	2-2
3-1	X Bus Initialization Types	3-3
5-1	List Mode Encoding	5-15
5-2	Lines-per-Row Encoding	5-16
5-3	Bits-per-Pel Encoding	5-16
5-4	PIB Size Encodings	5-18
5-5	Memory Type Encoding	5-21
5-6	DRAM Bank Encoding	5-21
5-7	Horizontal and Vertical Resolution Combinations	5-22
5-8	Sync Encoding	5-23
5-9	Horizontal and Vertical Zoom Encoding	5-25
6-1	List Mode 0 Characters	6-3
6-2	List Mode 1 Alphanumeric, DRCs, and Mosaics	6-4
6-3	List Mode 1 Fixed Objects	6-5
6-4	List Mode 2 Alphanumeric and Mosaics	6-6
6-5	List Mode 2 DCRs and Fixed Objects	6-7
6-6	List Mode 3 Characters	6-8
6-7	List Mode 4 Alphanumeric and Mosaics	6-9
6-8	List Mode 4 DRCs and Fixed Objects	6-10
6-9	List Mode 5 Alphanumeric and Mosaics	6-12
6-10	List Mode 5 DRCs and Fixed Objects	6-13
6-11	CMR Offsets in Color Mapping Address Space	6-23
6-12	Run-Length Encoding	6-29
6-13	Bit-Plane Encoding	6-29
7-1	M68000 Family Memory Map	7-5
7-2	MC6809 Memory Map	7-7
8-1	MC6883 Control Bits	8-2
8-2	Display Mode Selection	8-5
8-3	Semigraphic Colors	8-6
8-4	Semigraphic Block	8-6
8-5	Alphanumeric Character Selection	8-8
8-6	Alphanumeric Colors	8-8
8-7	Color-Mode Colors	8-10
8-8	Resolution-Mode Colors	8-11
8-9	Display Memory Requirements	8-11

SECTION 1

INTRODUCTION

This section provides a general overview of the MC68487 Raster Memory Controller (RMC).

1.1 KEY FEATURES

The following lists the key features available on the RMC.

- Two Machine Modes to Allow MC6883/MC6847 Emulation or Full Feature Operation
- Compatible with NTSC or PAL Displays
- Supports Interlace and Non-Interlace Data and Sync
- Supports up to 1 Megabyte of Dynamic Memory
- Horizontal Resolution from 64 to 640 Pixels
- Vertical Resolution from 64 to 500 Lines
- Bit-Plane Mode and Six Character Oriented List Modes
- 32 Available Colors from a Palette of 4096 Colors
- ASCII and Two Types of Mosaics in Internal ROM
- Up to 50 Character Rows, 32, 40, or 80 Characters Each
- From 32 to 32K User-Definable Characters
- Virtual Screen Much Larger than the Visible Screen
- Smooth Scrolling in Vertical and Horizontal Directions
- Supports Video Overlay
- Eight Hardware Objects Positioned by X and Y Registers
 - Underline
 - Flash
 - Invert

- **Text Oriented Attributes (Continued)**
 - Color
 - Double Height
 - Double Width
- **Game Oriented Attributes**
 - Collision
 - Priority
 - Shading
 - Color Offset
- **Intended for Personal and Home Computers, Engineering Stations, and Teletext/Videotex**

1.2 DISCUSSION OF TERMS

This section provides an orientation to the RMS capabilities and defines some of the terms used to describe them. Other terms are defined when they are introduced.

Pixel and Pel

The terms pixel and pel both mean picture element. Traditionally, pixel refers to the smallest physical picture element, which is the resolution of the CRT being used, and pel refers to the smallest logical picture element, which is set by the rest of the video system. This document uses pixel to mean the element set by the particular horizontal and vertical screen resolutions selected by the user; these set the limit on system resolution, and it is assumed that they will be picked to match the CRT resolution. Pel is used to refer to all other picture elements.

Banks

The RMS uses dynamic random-access memory (DRAM) organized in banks. Each bank consists of identical DRAM's connected in a byte-wide data bus (data inputs tied to data outputs) with a multiplexed address bus of 14, 16, or 18 lines (for 16K, 64K, or 256K byte banks).

Screen Memory, Virtual Screen, Displayed Screen, and Scroll

The RMS uses a block of DRAM as screen memory to contain the display information. The user locates this block using RMS registers as pointers. The data may be larger than the user's display can show at one time; the full set of data is the virtual screen, and the data currently being displayed is the displayed screen. The user may scroll smoothly through the virtual screen, moving the displayed screen as little as one pixel at a time vertically or horizontally or both by changing at most four RMS registers, while leaving the screen memory unchanged.

Bit-Plane Mode, List Mode, Image Table, Display List,

Attributes, Characters, and Fixed Objects

The user can display individual pels in the bit-plane mode, or characters and fixed objects in any of the six list modes. The screen memory in the bit-plane mode is arranged in scan lines. Within each scan line, the color of the first pel is followed by the color of the second, and so on. The characters and fixed objects used in the list modes are defined in image tables which contain their pel-by-pel descriptions. The list mode screen memory is arranged as a display list of pointers to entries in the image tables, character row by character row. The list modes allow the display list to include attributes for the images; these allow each individual occurrence of an image to be altered, for example by underlining or flashing. Characters and fixed objects differ from each other in the attributes that they may use, but the main difference is that fixed objects may interact with true objects and characters may not.

True Object

The true object follows few of the rules of the other images. Its many unique features are explained in 6.4 TRUE OBJECTS, but its primary distinction is that it is designed to move. Each of the eight true objects is placed on the screen using pointers and X/Y registers in the RMS. The pointer indicates the object's location in its image table in memory, and the X/Y registers located the object on the displayed screen. Flags indicate when each object has been completely displayed; the MPU may then change the X/Y registers to either move it smoothly on the screen or to create another object further down the screen. In this way, by changing only RMS registers, many more than eight true objects may appear on the screen, but only eight can appear in any single scan line. Registers also report overlap of true objects and fixed objects or of two true objects, and priorities allow control of the third dimension: objects can pass in front of or behind other objects. True objects may also be used in the bit-plane mode.

Alphanumerics, Mosaics, and Dynamically Redefinable Character Set (DRCS)

The RMS offers the user a variety of character types. It has 96 ASCII alphanumerics, 32 international characters, and two types of mosaics in internal ROM and allows a dynamically redefinable character set (DRCS) to be user defined. The DRC's come in several varieties, depending on the list mode. Some are eight pels wide and use exactly the same attributes as the alphanumerics, some are eight pels wide but with different attributes, and others are 16 pels wide.

HRES and VRES

Independent of the display mode, the user can set the displayed screen's horizontal resolution (HRES) and the vertical resolution (VRES).

Color Mapping RAM (CMR)

The color mapping RAM (CMR) consists of 32 registers in the RMS. Each register can be set to any one of 4096 colors. The bits that select colors in the bit-plane screen memory and the list mode image tables are used to select CMR registers, rather than actual colors.

1.3 GENERAL DESCRIPTION

The MC68487 raster memory controller (RMC) is a video display generator which is combined with the MC68486 Raster Memory Interface (RMI) to create a powerful chip set designated the Raster Memory System (RMS). When interfaced with a microprocessor and memory, the raster memory system becomes a sophisticated video display controller and memory management tool which offers application flexibility and ease of programming. Refer to Figure 1-1 for a block diagram of the RMC.

New operating modes inherent in the RMS are designed for word processing, high resolution color graphics for engineering work stations, video games applications, or teletext/videotex applications. New features in the RMS (such as virtual screen, color mapping RAM, and smooth scrolling) allow a designer to create a low cost text and graphics computer. With the addition of memory and more sophisticated software, the system evolves into a high level, high resolution personal computer with impressive game capabilities or a medium resolution computer-aided design station. The RMS is really two systems in one chip set: machine 1 provides the full RMS power for new designs and machine 2 provides backward compatibility with MC6809E systems using the MC6883-MC6847 combination.

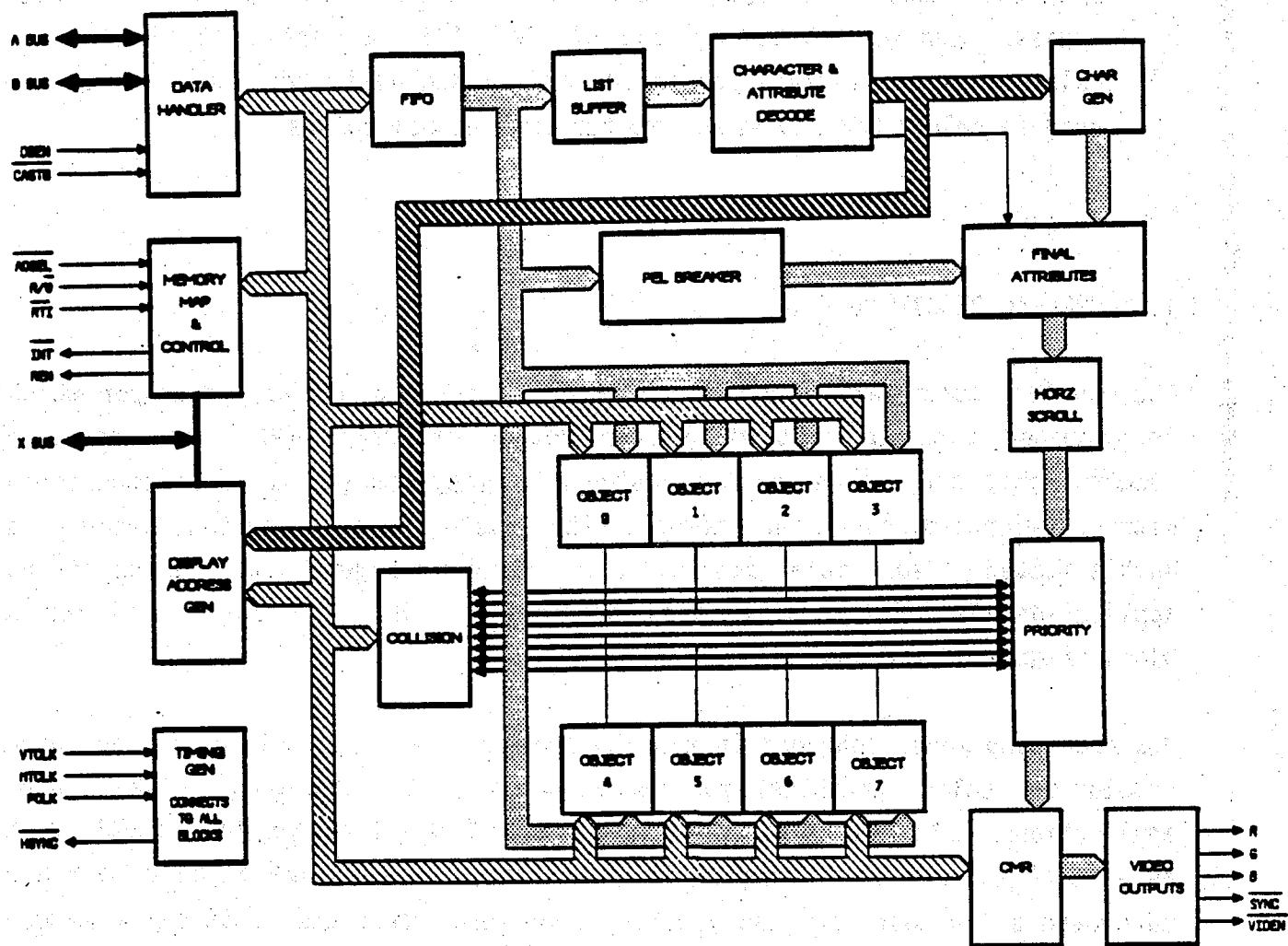


Figure 1-1. Raster Memory Controller Block Diagram

The RMS can operate with any of three Motorola microprocessors; the MC6809E, MC68008, or the MC68000. Memory can be expanded up to 1 megabyte with any microprocessor. An on-chip memory management unit is included for the MC6809E as well as the capacity to specify page-independent blocks (PIB) for I/O or scratchpad RAM. The microprocessor has transparent access to memory even during active display. The RMS also provides DRAM refresh automatically.

While the RMI handles clock generation, MPU interfacing, and memory management, the RMC specializes in creating a video display by performing display address calculations, video data processing, sync generation, and video outputs. The RMS multi-mode video has three sources of video; 1) bit-plane graphics, 2) list mode, and 3) true objects. An internal color mapping RAM (CMR) is used to generate color information in bit-plane and list modes, with a maximum of 32 colors from the 4096 color palette available simultaneously.

Six list modes are character oriented modes optimized for memory size, color selection, resolution, and type of application. Some of the list modes are animation oriented, while others are intended for word processing, graphics, or alpha-mosaic modes. The RMC contains 96 internal alpha/numeric characters, 64 mosaic characters, and 32 international characters that are available in modes 2 and 5 only. In addition to these, up to 32K dynamically redefinable characters are supported. All of these character types can have internally generated attributes applied to them such as color, color offset, underline, multiple flash rates, double high and double wide, or any combination of the above.

The RMS generates eight hardware objects that can be used in either bit-plane or any list mode. These objects are either bit-plane or run length encoded and can be positioned on the screen independent of the background display by X/Y position registers. Animation can be achieved by having several different views of the same object defined in DRAM and changing between them by changing only the name byte in one of the eight object name registers. The attributes associated with true objects include: color, binary zoom, priority, collision reporting, and interrupt capabilities. A specific subset of user definable characters, called fixed objects, can be used to generate a background display that will interact with the eight true objects. Collision detection, priority, and shading are attributes involved in this interaction.

The RMS incorporates a virtual screen that can be much larger than the displayed screen in both horizontal and vertical directions. Internal registers allow placement of the displayed screen anywhere within the virtual screen. Associated with the virtual screen, RMS includes smooth scrolling on a pixel-by-pixel basis in both directions, allowing the user to pan the virtual screen. When the displayed screen is positioned off the edge of the virtual screen, the display will automatically wrap around to the other side of the virtual screen. In the bit-plane mode, the wrap-around feature can be set so that the RMS fills the portion of the displayed screen out of the virtual screen with a constant color. When the displayed screen and the virtual screen are defined to be the same size, barrel scrolling is possible.

Horizontal and vertical resolutions are programmable independent of the list mode or bit-plane mode used. Resolution can range from 64 to 640 pixels horizontally and 64 to 500 pixels vertically. The RMS is designed to support both interlaced and non-interlaced modes with both PAL and NTSC timing available. It is also possible to synchronize the RMS to an external video source to allow overlay of RMS video. The outputs of the RMS are analog RGB. An additional chip, the MC1377, can be used to convert the RGB outputs to composite PAL or NTSC video, while the MC1378 should be used in overlay applications.

SECTION 2 SIGNAL DESCRIPTION

The input and output signals can be functionally organized into the groups shown in Figure 2-1. The following paragraphs provide a brief description of the signals and a reference (if applicable) to other paragraphs that contain more detail about the function being performed.

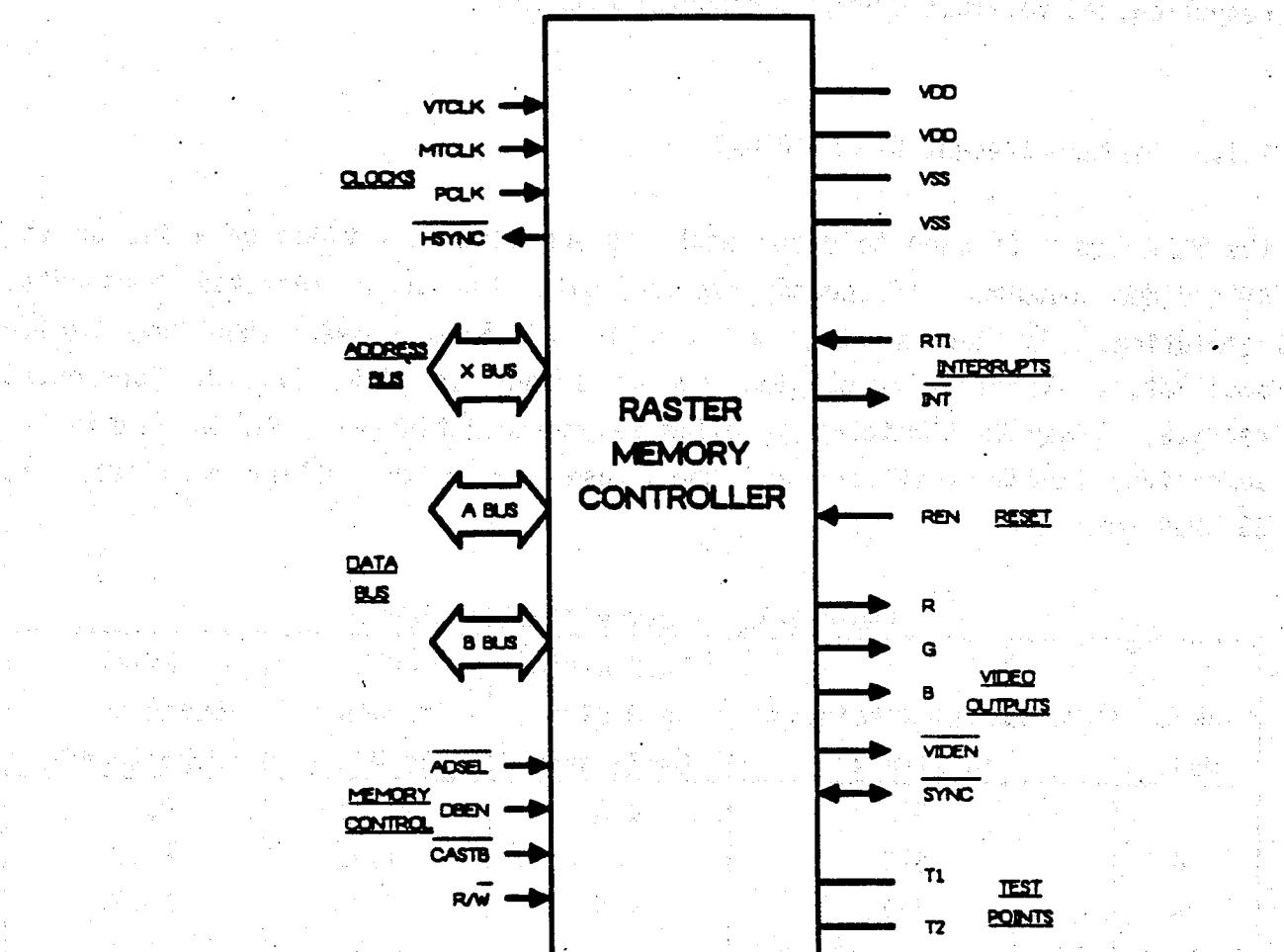


Figure 2-1. Functional Diagram

2.1 CLOCKS

The following paragraphs describe the four clock functions available on the RMC.

2.1.1 Video Timing Clock (VTCLK)

This input signal is used by the RMC to generate horizontal and vertical sync pulses, blanking and other internal timing signals. VTCLK is always equal to the master oscillator frequency divided by five. It is a free-running clock with a 50% duty cycle and is never resynchronized. All other clocks are resynchronized to VTCLK during horizontal retrace.

2.1.2 Picture Element Clock (PCLK)

The PCLK input is used to clock each picture element's video data out of the RMC video outputs. Frequency varies with the user selected horizontal resolution. It has a range of 2.5 to 6 times slower than the master oscillator. Vertical resolution has no effect on PCLK. During horizontal retrace, it may be stretched to allow resynchronization with VTCLK. Table 2-1 summarizes PCLK's available and their usage when the master oscillator is 35.79545 MHz.

Table 2-1. Picture Element Clock

HRES Mode	Horizontal Resolution in Pixels	PCLK Ratio to Master Oscillator	PCLK Frequency in MHz	Pixel Duration in Nanoseconds
7	640	2.5	14.32	69.8
6	512	3	11.93	83.8
4	320	4.5	7.95	125.7
2, 1, 0	256 (Narrow), 128, 64	5	7.16	139.7
3	256 (Wide)	6	5.97	167.6

2.1.3 Memory Timing Clock (MTCLK)

This input provides memory cycle timing. A memory cycle is 1.01 to 1.34 microseconds in duration. Each memory cycle consists of nine MTCLK cycles. The first eight cycles are the master oscillator divided by four with a 50% duty cycle, and the ninth cycle is stretched, if necessary, to match PCLK cycles. MTCLK is also stretched during horizontal sync to resynchronize the memory cycle with VTCLK on each video line. A memory cycle is exactly 16 PCLK cycles in horizontal resolution modes of 640 or 512 pixels and eight PCLK cycles for all the lower resolution modes. Figure 2-2 provides timing relationships for MTCLK and PCLK for different horizontal resolution modes.

2.1.4 Horizontal Sync (Hsync)

This output is connected between the RMI and the RMC. It is used to maintain synchronization between the clocks in the system. Hsync timing: 1) occurs each horizontal video line, and 2) may be used for synchronization of external hardware. The trailing (rising) edge of Hsync occurs one VTCLK period before the end of the horizontal line. The RMI detects this event and waits until the next falling edge of VTCLK to restart PCLK, MTCLK, and the MPU clock. This sync point signals the start of a pel and a memory cycle. The amount of clock-stretching of the three clocks is dependent on the display mode in use.

2.2 ADDRESS/DATA BUSES (X, A, B, S, and Z)

Operation of the address/data buses are described in the following paragraphs.

2.2.1 Address Bus X (X0-X9)

The X bus is a bidirectional 10-bit bus that connects the RMI, the RMC, and 74ALS Family logic for MPU address bus interfacing. Time-division multiplexing of the X bus is controlled by MTCLK, ADEN, and ADSEL from the RMI. Refer to Figures 2-3 through 2-8.

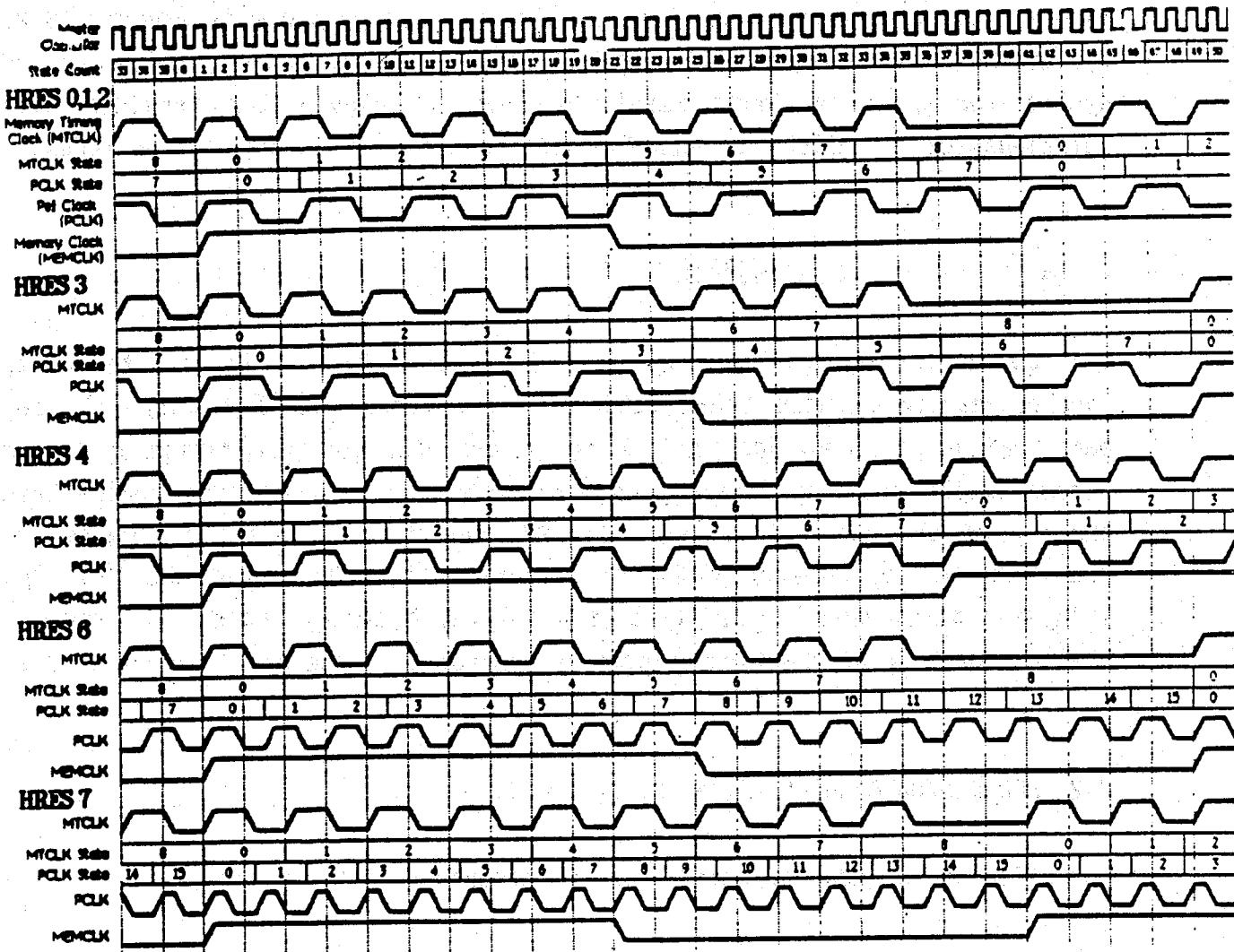


Figure 2-2. MTCLK and PCLK Timing vs HRES Modes

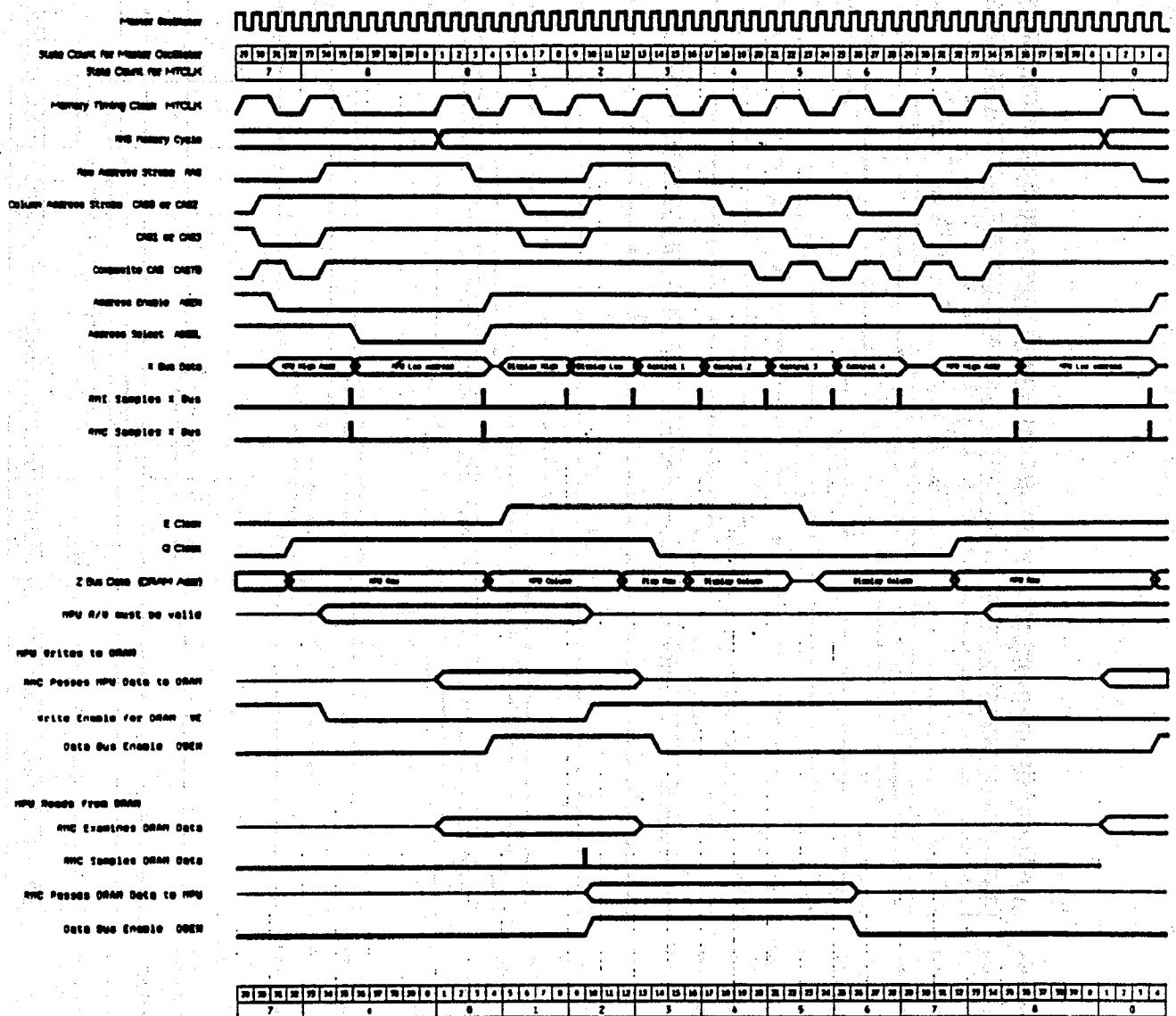


Figure 2-3. MC6809E X-Bus Timing for HRES 0, 1, 2, 7

MC68487

MOTOROLA

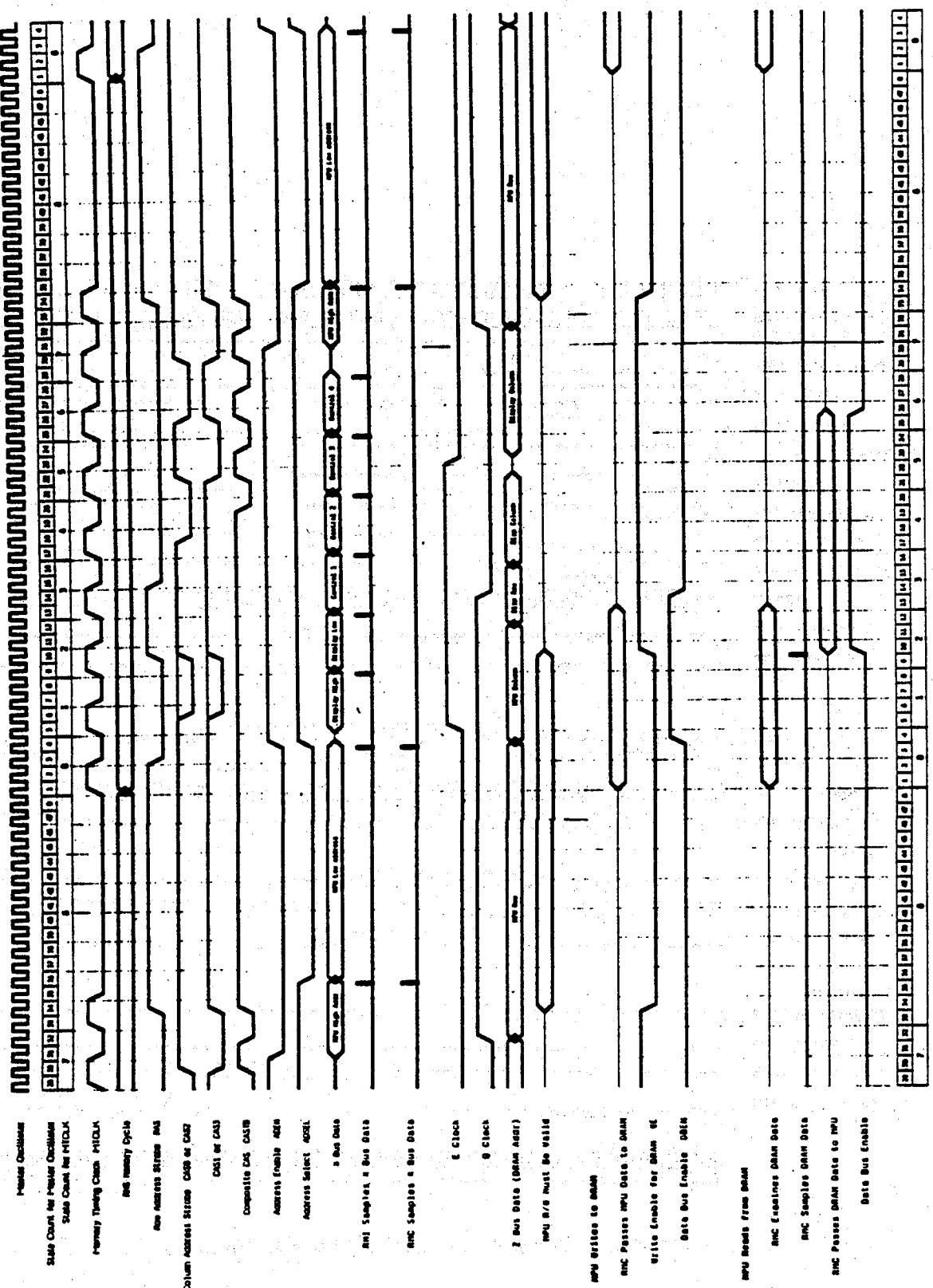


Figure 2-4. MC6809E X-Bus Timing for HRES 3, 6

MOTOROLA

2-6

MC68487

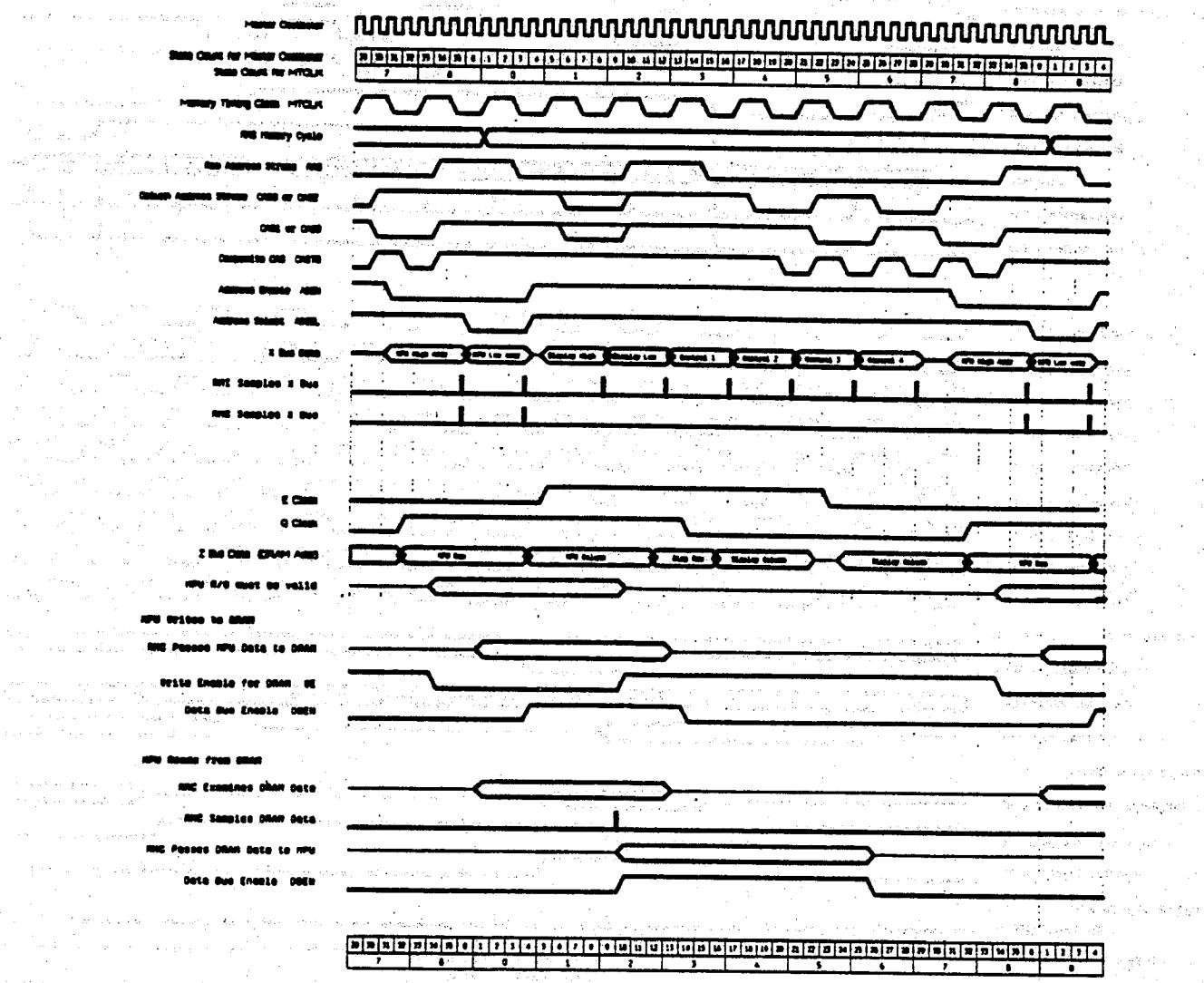


Figure 2-5. MC6809E X-Bus Timing for HRES 4

MC68497

MOTOROLA

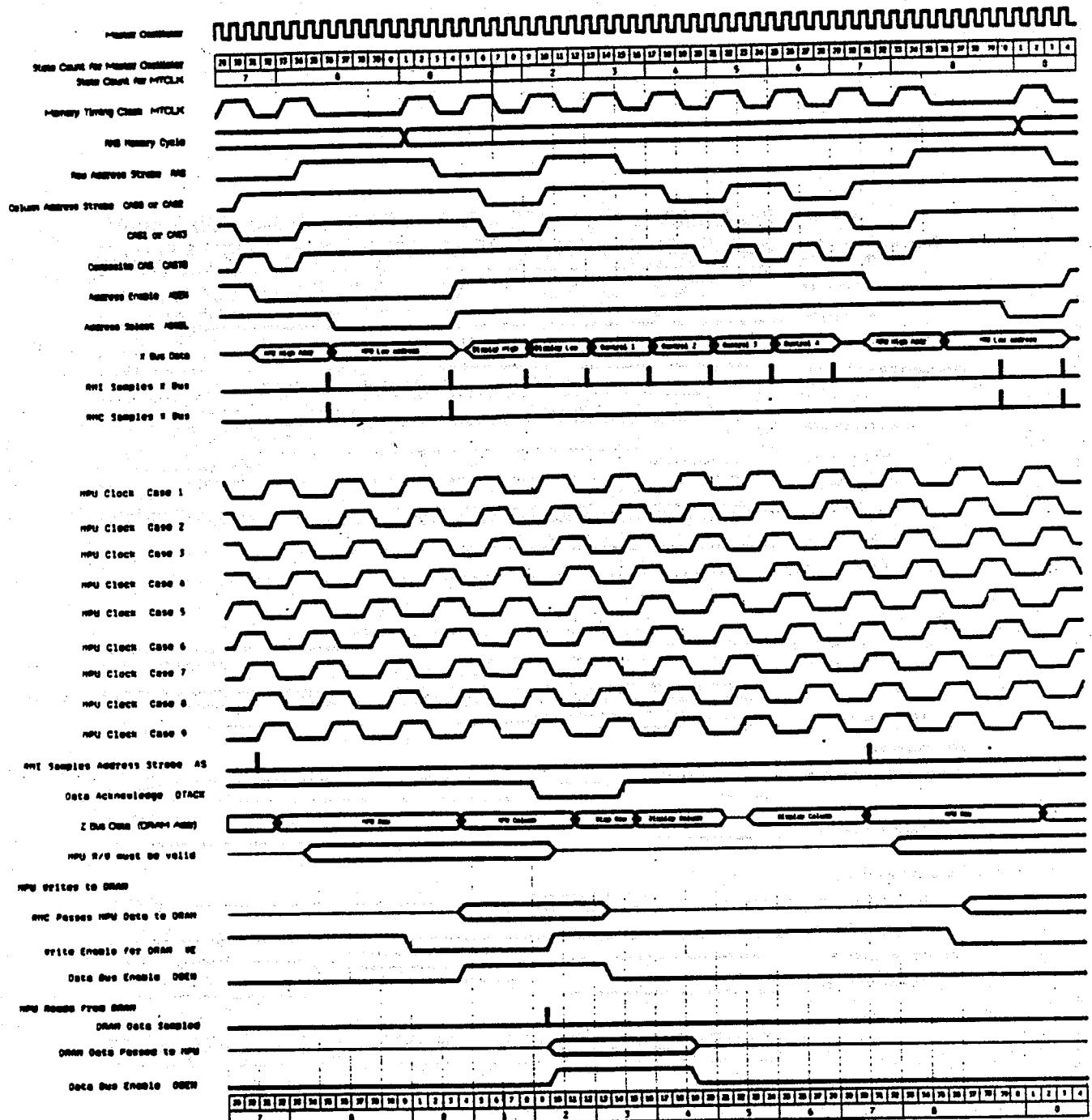


Figure 2-6. MC68000/MC68008 X-Bus Timing for HRES 0, 1, 2, 7

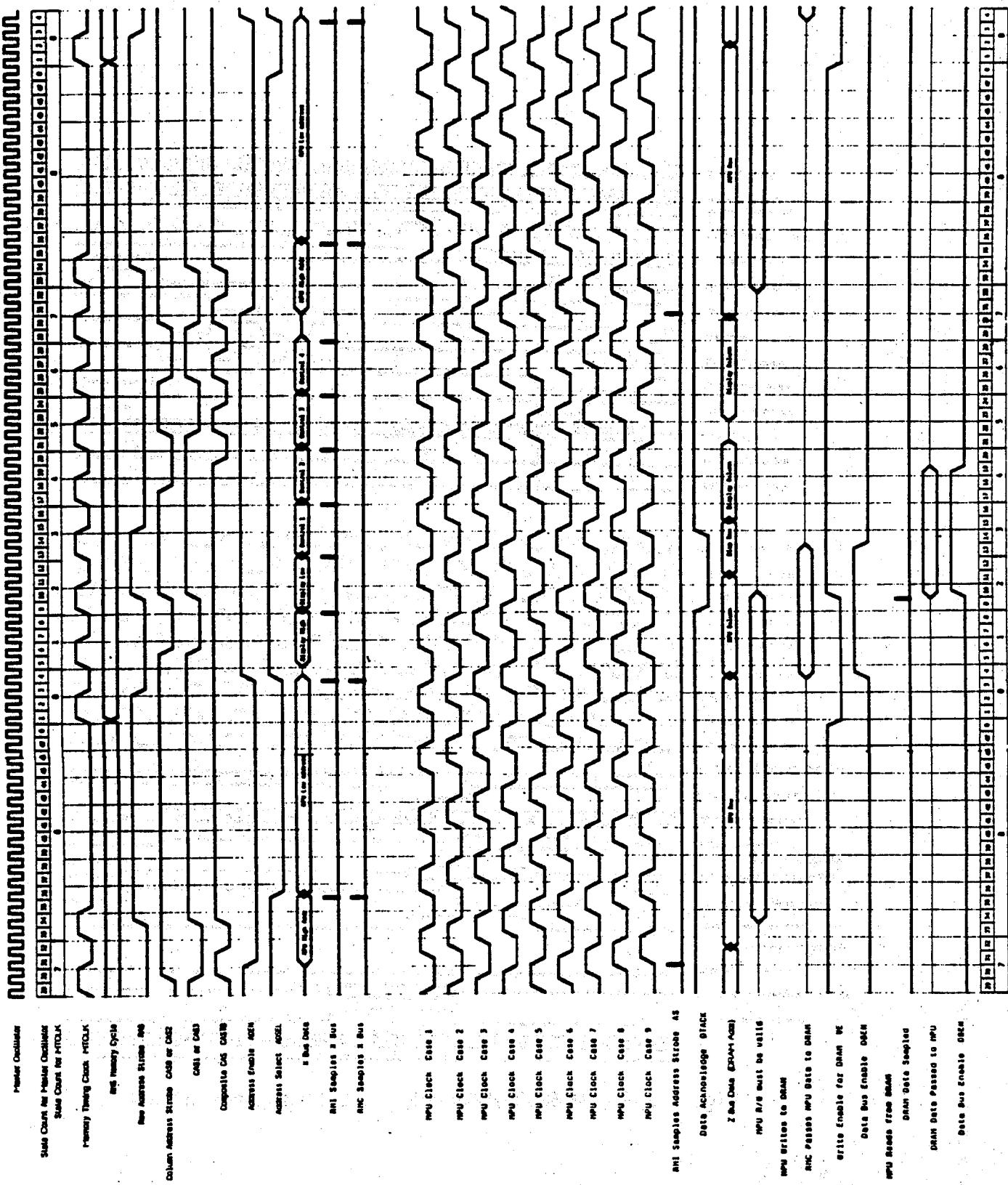


Figure 2-7. MC68000/MC68008 X-Bus Timing for HRES 3, 6

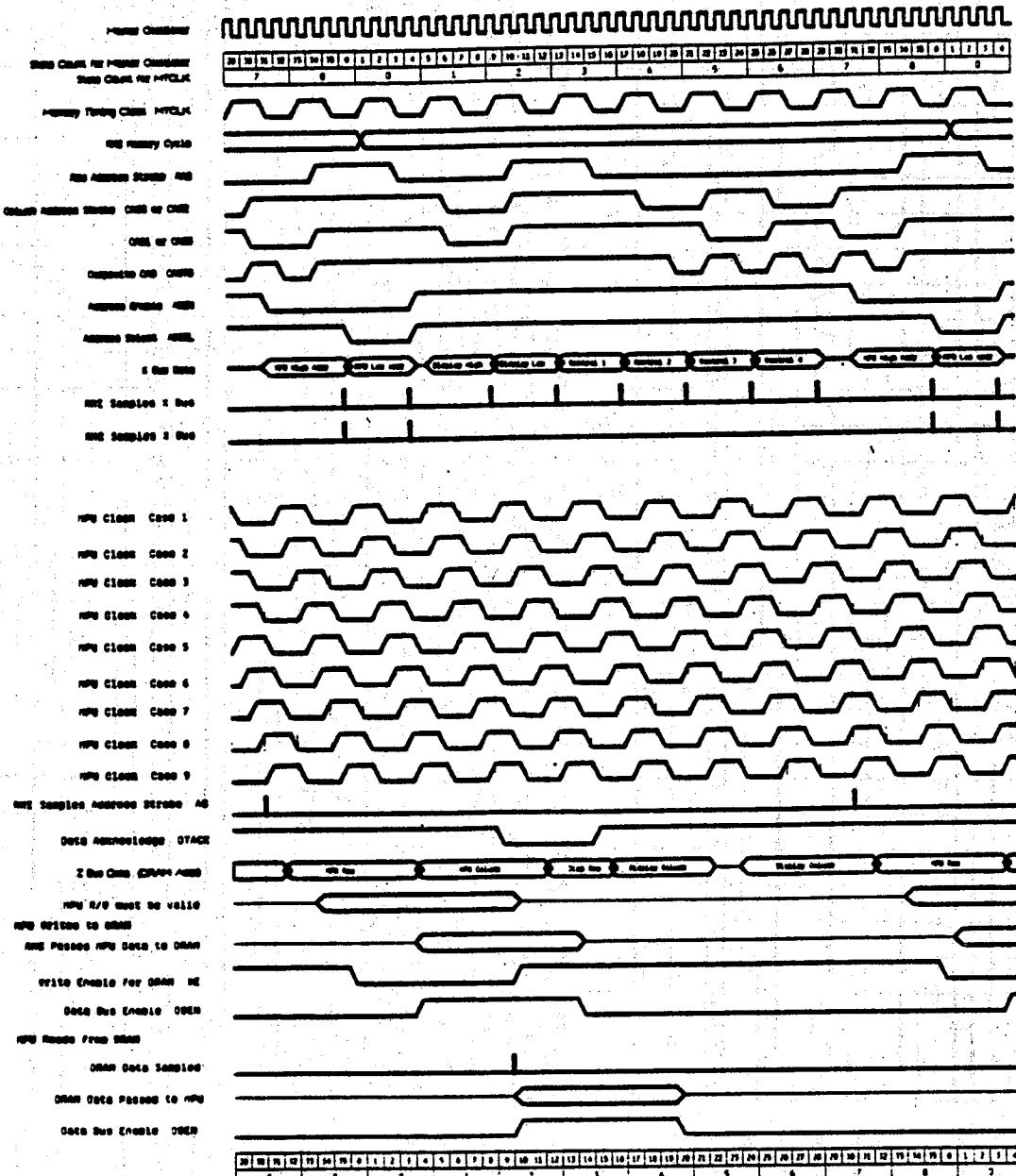


Figure 2-8. MC68000/MC68008 X-Bus Timing for HRES 4

MOTOROLA

2-10

MC68487

2.2.2 Data Buses A and B

There are two 8-bit buses on the RMC used for data transfer between MPU, RMS, and DRAM. Both the A and B buses are bidirectional. They are capable of driving the DRAM data bus directly and driving the MPU data bus in most applications. The way in which these buses are used is dependent on the type of MPU used. Additional 74ALS logic is required for interfacing if a 16-bit MPU is used. Refer to Figures 4-1, 4-2, and 4-3 for system block diagrams.

2.2.2.1 DATA BUS A (A0-A7). When the RMS is used with an 8-bit MPU (either the MC6909E or the MC60008) data bus A is connected to the DRAM data bus. It is used in both reads and writes of memory. If a 16-bit MPU is used, data bus A becomes the least-significant bits of the 16-bit MPU data bus and DRAM bank 1 (and bank 3 if used).

2.2.2.2 DATA BUS B (B0-B7). The B bus is connected to the MPU's data bus when an 8-bit microprocessor is used. For a 16-bit MPU, the B bus is connected via interface logic to the most-significant bits of the 16-bit of the MPU's data bus and to DRAM bank 0 (and bank 2 if used). Refer to Figure 4-4 for an interface schematic and to Figure 7-1 for an illustration of the DRAM bank organization for both A and B buses in a 16-bit system.

2.2.3 S and Z Buses

The S bus is used for external chip select decoding and the Z bus acts as the DRAM address bus for all DRAM reads/writes. For further information on these buses refer to the MC68486 (RMI) Advanced Information Data Sheet.

2.3 MEMORY HANDSHAKE

The following paragraphs describe the memory handshake signals: ADSEL, DBEN, CASTB, and R/W.

2.3.1 Address Select (ADSEL)

ADSEL is supplied to the RMC by the RMI. It is used by the RMC during MPU address placement of the high-order (ADSEL = 1) and low-order (ADSEL = 0) addresses on the X bus.

2.3.2 Data Bus Enable (DBEN)

DBEN is generated by the RMI for the RMC. In addition to R/W, it determines timing for RMC to enable the MPU to read or write either the DRAM or a RMS control register.

2.3.3 Composite CAS Strobe (CASTB)

CASTB is a composite strobe of CAS0-CAS3 generated by the RMI. It is used to strobe display data from the DRAM into the RMC.

2.3.4 Read/Write (R/W)

R/W is used to control the direction of data movement for MPU access to either DRAM or RMS control registers.

2.4 MPU INTERRUPTS

The following paragraphs describe the MPU interrupt signals.

2.4.1 Real-Time Input (RTI)

RTI is a negative edge input to the RMC that causes the current value of the X and Y display screen counters to be loaded into internal RMC registers. These registers can be read by the MPU. RTI may be used as a light-pen input.

2.4.2 Interrupt (INT)

INT is an open-drain output used to generate an interrupt to the MPU. INT can be programmed to generate interrupts on specific events or can be made inactive. INT has an internal passive pullup. Refer to SECTION 5 REGISTER DESCRIPTION for more complete details.

2.5 VIDEO OUTPUTS

The following paragraphs describe the video outputs.

2.5.1 Red, Green, and Blue (R, G, B)

The R, G, and B outputs are analog signals used to drive the red, green, and blue color guns of a CRT. The outputs vary from a low level representing blanking, to a high level representing peak luminance and the difference between these levels is 1.0 volt. These outputs are designed to drive a high-impedance load. Buffering is required when the load impedance is less than 100K ohms.

The G output can have an optional sync level which is 0.4 volts below blanking. The optional sync is programmed in the sync mode register in the RMC.

2.5.2 Video Enable (VIDEN)

The video enable signal is an output intended for use in overlay applications. The level on this pin reflects the status of the VEN bit of the current output color-mapping RAM (CMR). Setting the VEN bit within a CMR will cause VIDEN to go low when that CMR becomes the current output. This transition can be used as the control signal for a video multiplexer that would select either RMS video or an external source.

2.6 SYNC (SYNC)

SYNC is a dual-purpose pin. It can be used as an output for a sync signal or as a frame sync input. Its function is defined by the value of the sync-mode register.

When SYNC is programmed as an output, it provides a TTL level composite sync, block vertical sync, or horizontal sync. Composite sync will include equalizing pulses only when the RMS is operating in an interlaced mode.

In the input mode, a high-to-low transition will reset the RMC's internal vertical video timing generator to the trailing edge of vertical sync.

2.7 RESET ENABLE (REN)

The REN output is used in conjunction with external circuitry to reset the RMS. The function of this pin is discussed in SECTION 3 RESET AND INITIALIZATION.

2.8 TEST POINTS (T1, T2)

Test points T1 and T2 are used for testing during manufacture. They should be connected to ground.

2.9 V_{CC} and GND

Power is supplied to the RMC using these four pins. V_{CC} is power and GND is ground.

SECTION 3

RESET AND INITIALIZATION

This section describes reset and initialization functions of the RMC.

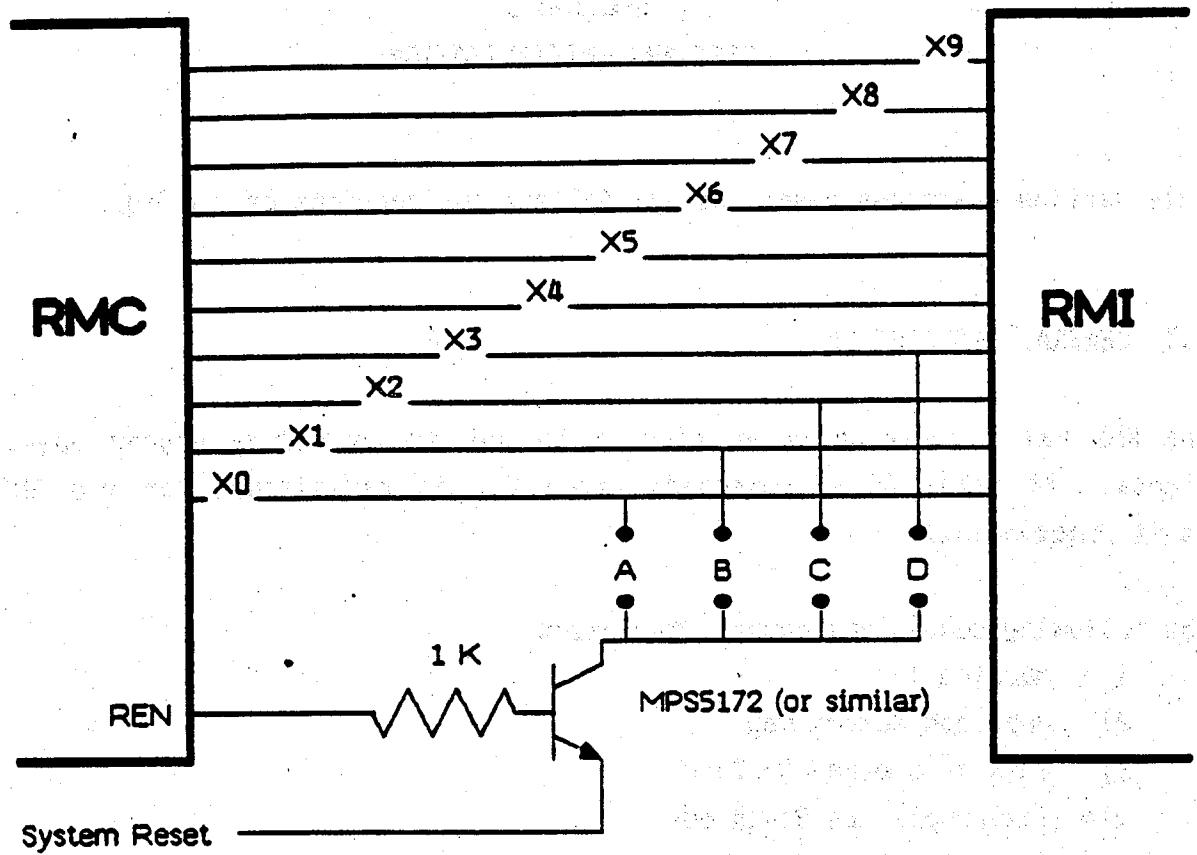
3.1 GENERAL DESCRIPTION

The RMS has a power-on reset circuit to set it up before proper operation starts. It takes 64 microseconds after V_{CC} is established for the RMS to start functioning.

The following conditions occur after reset.

- 1) Machine 1
- 2) Unfolded Memory Map
- 3) MAPA is Cleared to Zero
- 4) Interrupts are Disabled
- 5) Video Display Disabled
- 6) UPI and LPI Disabled, Paging Register Set to All Ones
- 7) Non-Interlaced Sync and Data
- 8) Horizontal Resolution 4
- 9) Border Color Bits are Reset
- 10) Swap Bit is Reset

Reset enable (REN) is an RMC output that goes high during state MT6 of the nine cycle memory timing and is used to reset the RMS. See Figure 3-1. The RMS monitors the X bus during MT6 cycle; therefore, if the MPU is in reset (emitter is low) the RMS recognizes a reset and sets itself according to the X0-X3 configuration (see Table 3-1).



USE JUMPER A FOR MC6809E AND 625 LINE DISPLAY

USE JUMPER B FOR MC6809E AND 525 LINE DISPLAY

USE JUMPER C FOR MC68008 OR MC68000 AND 625 LINE DISPLAY

USE JUMPER D FOR MC68008 OR MC68000 AND 525 LINE DISPLAY

Figure 3-1. X Bus Reset

MOTOROLA ======

Table 3-1. X Bus Initialization Types

Line (Low)	Meaning
X0	MC6809E MPU and 625 Line Video Timing
X1	MC6809E MPU and 525 Line Video Timing
X2	MC68000/MC68008 MPU and 625 Line Video Timing
X3	MC68000/MC68008 MPU and 525 Line Video Timing

The reset should last at least 64 microseconds for proper operation. The control registers which are not mentioned above are undefined after reset. Therefore, the user has to select the display mode, virtual screen and visible screen sizes, memory types, etc.

3.2 VIDEO RESET

If the RMC pin is programmed as an input (see 5.2.19 \$FFE26 Sync Mode), a falling edge input will reset the RMCs internal video timing generator to the trailing edge of vertical sync.

SECTION 4

RMS MICROPROCESSOR INTERFACE

The RMS is designed to work with several members of the Motorola Family of 8- and 16-bit MPUs. The user has a choice of the MC6809E 8-bit MPU, the MC68008 8-bit MPU, or the MC68000 16-bit MPU. The user's choice of MPU has a significant effect on the architecture and performance of the total system.

Refer to Figures 4-1 through 4-3 for system block diagrams and to Figure 4-4 for an MC68000 data bus logic diagram.

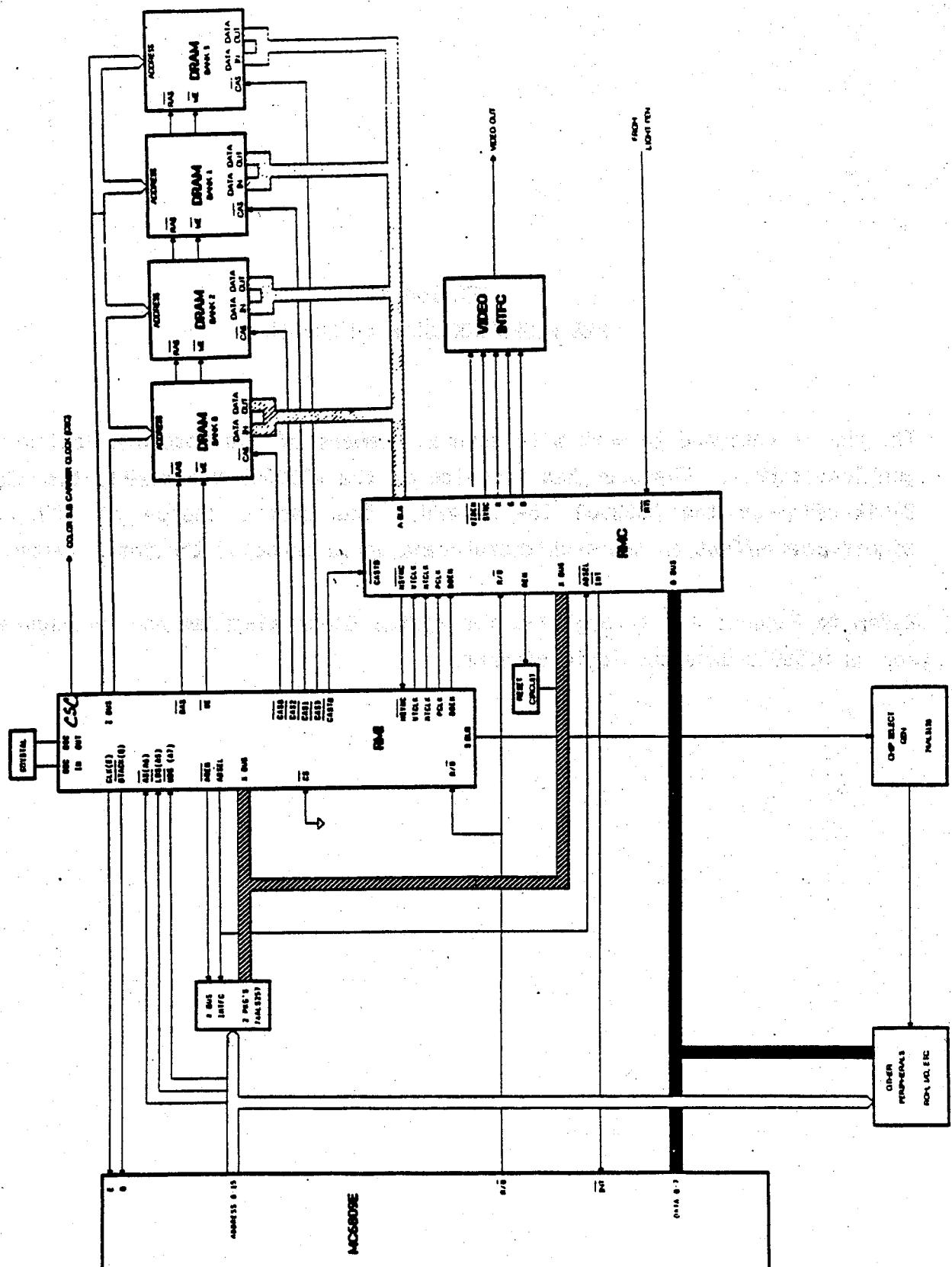


Figure 4-1. MC6809E System Block Diagram

MOTOROLA
4-2

MC68487

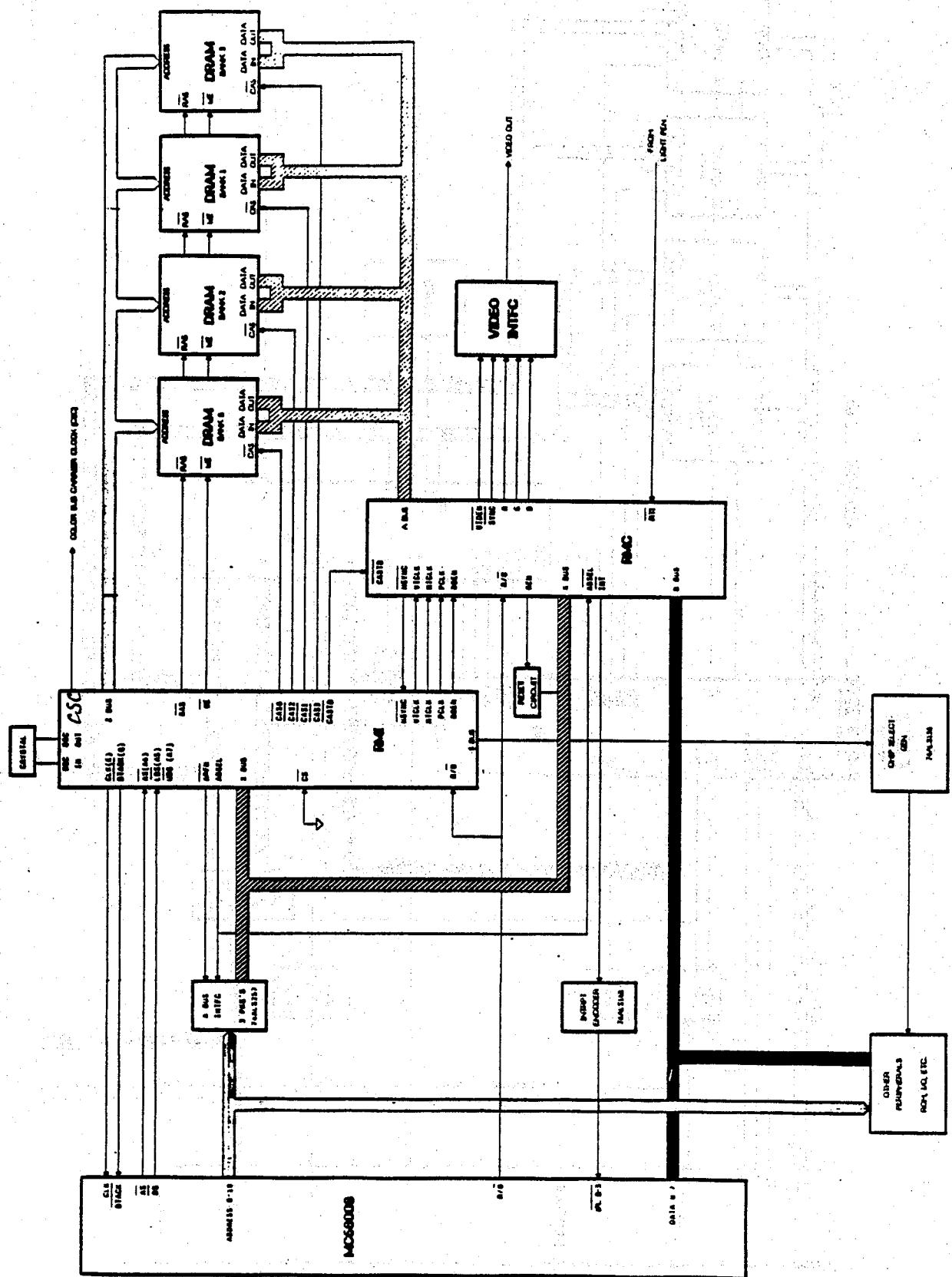


Figure 4-2. MC68008 System Block Diagram

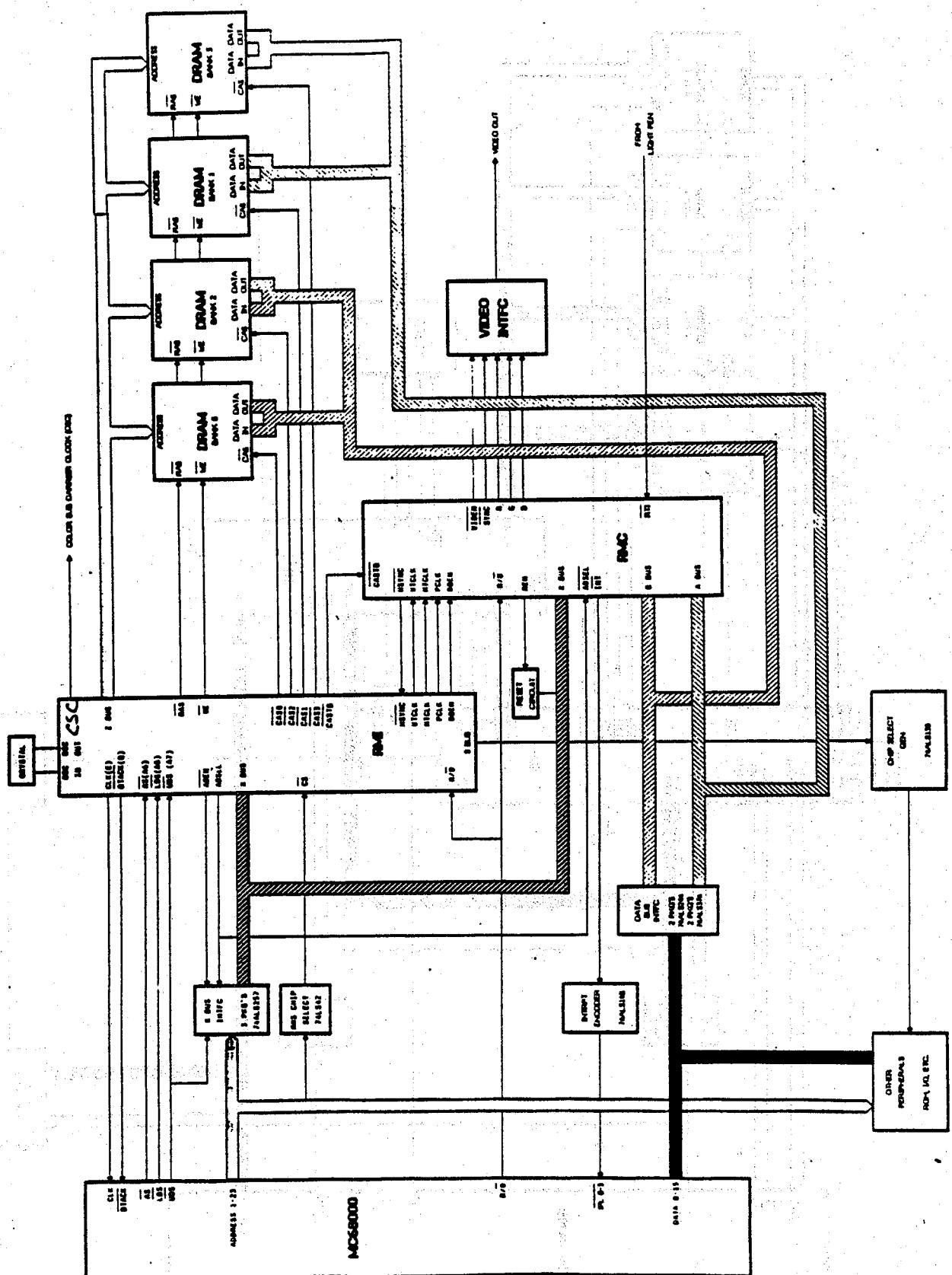


Figure 4-3. MC68000 System Block Diagram

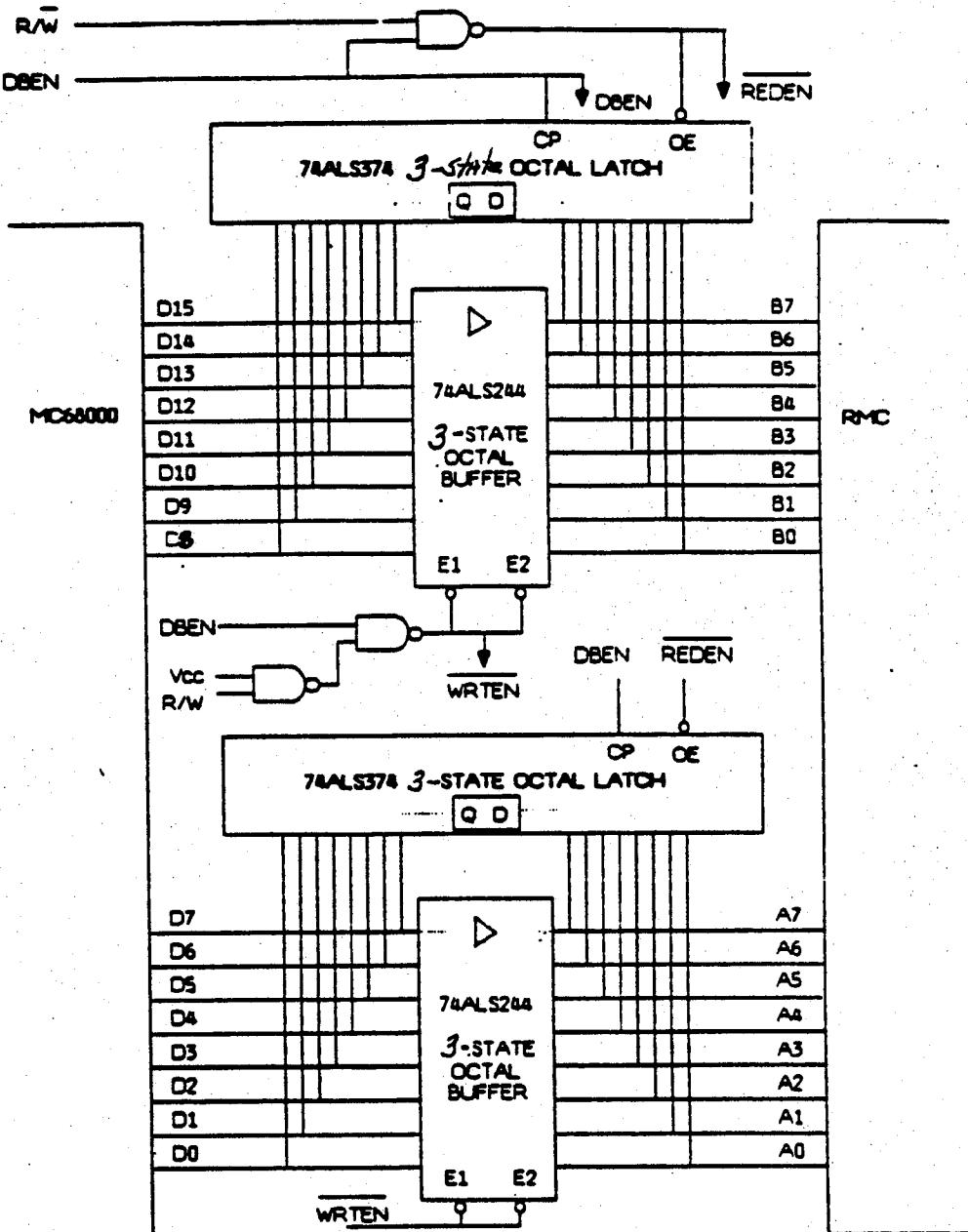


Figure 4-4. MC68000 Data Bus Logic

SECTION 5 REGISTER DESCRIPTION

The RMC control registers may be configured as 192 contiguous bytes or as three 64-byte pages. The three-page folded option reduces memory use and is intended for the MC6809E interface only. The three pages are mapped at locations \$FFF80-\$FFFBF. The page selection is done by using the memory map control register which is available in page 0 and 2. In page 1, the map bits are duplicated as bits 15 and 14 of CMR register \$00. This allows the user to change pages from any page. The MC6809E can run in folded or unfolded map in machine 1, but in machine 2 it has to run in the folded map to make it backward compatible with the MC6883/MC6847 map. Machine 2 is discussed in SECTION 8 MACHINE 2.

Figure 5-1 shows the memory map with the unfolded addresses on the left and the folded addresses on the right. The first 16 memory locations of pages 0 and 2 are duplicates. The address shown for each register in the description section is the unfolded address.

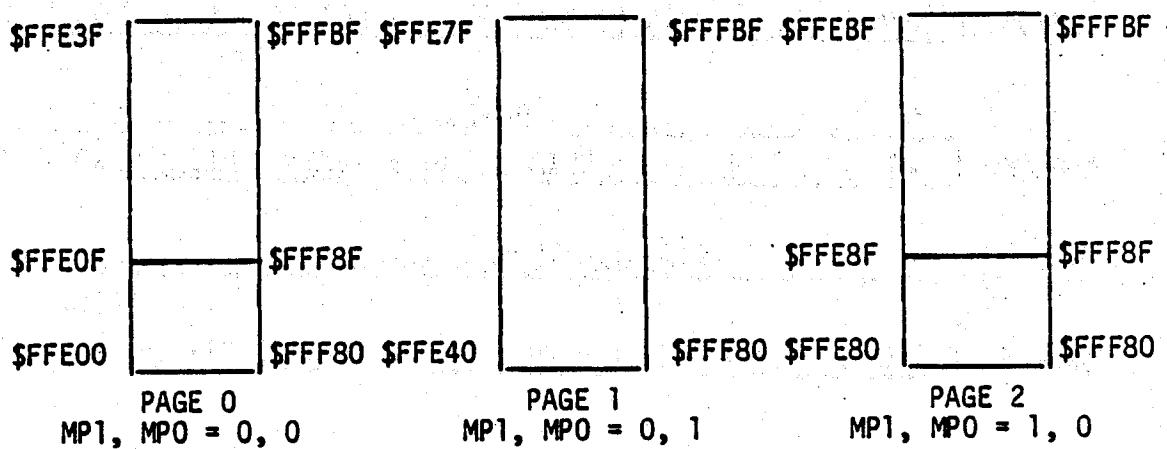


Figure 5-1. Memory Map

5.1 RMS REGISTER MAP

The following register map shows all 192 bytes in the RMS register space, including the unused addresses. Unfolded addresses are shown to the left of each byte, and folded addresses/page numbers to the right. Refer to 5.2 RMC CONTROL REGISTERS for more detailed information.

UNFOLDED ADDRESS

FOLDED ADDRESS/PAGE

MEMORY MAP

\$FFE00	MPT	MPO	UF	M2	--	--	--	\$FFF80/00
---------	-----	-----	----	----	----	----	----	------------

DISPLAY DATA MODE

\$FFE01	BP	LM2	LM1	LMO	LPR1	LPRO	BPP1	BPP0	\$FFF81/00
---------	----	-----	-----	-----	------	------	------	------	------------

INTERRUPT CONTROL/STATUS

\$FFE02	IPT	--	--	RTI	RTO	OFN	BLK	COL	\$FFF82/00
---------	-----	----	----	-----	-----	-----	-----	-----	------------

BORDER COLOR

\$FFE03	WC	MAPA	DV	BC4	BC3	BC2	BC1	BC0	\$FFF83/00
---------	----	------	----	-----	-----	-----	-----	-----	------------

OBJECT AVAILABLE

\$FFE04	07A	06A	05A	04A	03A	02A	01A	00A	\$FFF84/00
---------	-----	-----	-----	-----	-----	-----	-----	-----	------------

PAGING

\$FFE05	--	--	--	SWAP	PG3	PG2	PG1	PG0	\$FFF85/00
---------	----	----	----	------	-----	-----	-----	-----	------------

PAGE INDEPENDENT BLOCKS

\$FFE06	UEN	UPI2	UPI1	UPI0	LEN	LPI2	LPI1	VEC	\$FFF86/00
---------	-----	------	------	------	-----	------	------	-----	------------

UNFOLDED ADDRESSFOLDED ADDRESS/PAGE

VERTICAL SCROLL

\$FFE07	--	--	--	DHP	VSC3	VSC2	VSC1	VSC0	\$FFF87/00
---------	----	----	----	-----	------	------	------	------	------------

HORIZONTAL SCROLL

\$FFE08	--	--	--	DWP	HSC3	HSC2	HSC1	HSC0	\$FFF88/00
---------	----	----	----	-----	------	------	------	------	------------

\$FFE09	--	--	--	--	--	--	--	--	\$FFF89/00
---------	----	----	----	----	----	----	----	----	------------

DRC IMAGE TABLE START ADDRESS

\$FFE0A	--	--	--	--	DS19	DS18	DS17	DS16	\$FFF8A/00
---------	----	----	----	----	------	------	------	------	------------

\$FFE0B	DS15	DS14	DS13	DS12	DS11	DS10	0	0	\$FFF8B/00
---------	------	------	------	------	------	------	---	---	------------

TRUE OBJECT IMAGE TABLE START ADDRESS

\$FFE0C	--	--	--	--	TS19	TS18	TS17	TS16	\$FFF8C/00
---------	----	----	----	----	------	------	------	------	------------

\$FFE0D	TS15	TS14	TS13	TS12	TS11	TS10	0	0	\$FFF8D/00
---------	------	------	------	------	------	------	---	---	------------

FIXED OBJECT IMAGE TABLE START ADDRESS

\$FFE0E	--	--	--	--	FS19	FS18	FS17	FS16	\$FFF8E/00
---------	----	----	----	----	------	------	------	------	------------

\$FFE0F	FS15	FS14	FS13	FS12	FS11	FS10	0	0	\$FFF8F/00
---------	------	------	------	------	------	------	---	---	------------

OBJECT 0 COLLISION STATUS

\$FFE10	OC7	OC6	OC5	OC4	OC3	OC2	OC1	OC0	\$FFF90/00
---------	-----	-----	-----	-----	-----	-----	-----	-----	------------

OBJECT 1 COLLISION STATUS

\$FFE11	1C7	1C6	1C5	1C4	1C3	1C2	1C1	1C0	\$FFF91/00
---------	-----	-----	-----	-----	-----	-----	-----	-----	------------

OBJECT 2 COLLISION STATUS

\$FFE12	2C7	2C6	2C5	2C4	2C3	2C2	2C1	2C0	\$FFF92/00
---------	-----	-----	-----	-----	-----	-----	-----	-----	------------

OBJECT 3 COLLISION STATUS

\$FFE13	3C7	3C6	3C5	3C4	3C3	3C2	3C1	3C0	\$FFF93/00
---------	-----	-----	-----	-----	-----	-----	-----	-----	------------

UNFOLDED ADDRESSFOLDED ADDRESS/PAGE

OBJECT 4 COLLISION STATUS

\$FFE14	4C7	4C6	4C5	4C4	4C3	4C2	4C1	4C0	\$FFF94/00
---------	-----	-----	-----	-----	-----	-----	-----	-----	------------

OBJECT 5 COLLISION STATUS

\$FFE15	5C7	5C6	5C5	5C4	5C3	5C2	5C1	5C0	\$FFF95/00
---------	-----	-----	-----	-----	-----	-----	-----	-----	------------

OBJECT 6 COLLISION STATUS

\$FFE16	6C7	6C6	6C5	6C4	6C3	6C2	6C1	6C0	\$FFF96/00
---------	-----	-----	-----	-----	-----	-----	-----	-----	------------

OBJECT 7 COLLISION STATUS

\$FFE17	7C7	7C6	7C5	7C4	7C3	7C2	7C1	7C0	\$FFF97/00
---------	-----	-----	-----	-----	-----	-----	-----	-----	------------

COLLISION ENABLE

\$FFE18	CEN7	CEN6	CEN5	CEN4	CEN3	CEN2	CEN1	CENO	\$FFF98/00
---------	------	------	------	------	------	------	------	------	------------

\$FFE19	--	--	--	--	--	--	--	--	\$FFF99/00
---------	----	----	----	----	----	----	----	----	------------

\$FFE1A	--	--	--	--	--	--	--	--	\$FFF9A/00
---------	----	----	----	----	----	----	----	----	------------

\$FFE1B	--	--	--	--	--	--	--	--	\$FFF9B/00
---------	----	----	----	----	----	----	----	----	------------

REAL TIME OUTPUT X COORDINATE

\$FFE1C	--	--	--	--	--	--	OX9	OX8	\$FFF9C/00
---------	----	----	----	----	----	----	-----	-----	------------

\$FFE1D	OX7	OX6	OX5	OX4	OX3	OX2	OX1	OX0	\$FFF9D/00
---------	-----	-----	-----	-----	-----	-----	-----	-----	------------

REAL TIME OUTPUT Y COORDINATE

\$FFE1E	--	--	--	--	--	--	OY9	OY8	\$FFF9E/00
---------	----	----	----	----	----	----	-----	-----	------------

\$FFE1F	OY7	OY6	OY5	OY4	OY3	OY2	OY1	OYO	\$FFF9F/00
---------	-----	-----	-----	-----	-----	-----	-----	-----	------------

UNFOLDED ADDRESSFOLDED ADDRESS/PAGE

REAL TIME INPUT X COORDINATE

\$FFE20	--	--	--	--	--	--	IX9	IX8	\$FFFA0/00
\$FFE21	IX7	IX6	IX5	IX4	IX3	IX2	IX1	IX0	\$FFFA1/00

REAL TIME INPUT Y COORDINATE

\$FFE22	--	--	--	--	--	--	IY9	IY8	\$FFFA2/00
\$FFE23	IY7	IY6	IY5	IY4	IY3	IY2	IY1	IY0	\$FFFA3/00

MEMORY ORGANIZATION

\$FFE24	MTYP3	MTYP2	MTYP1	MTYP0	DB1	DB0	--	--	\$FFFA4/00
---------	-------	-------	-------	-------	-----	-----	----	----	------------

VIDEO OPERATION

\$FFE25	IS	ID	VRES2	VRES1	VRES0	HRES2	HRES1	HRES0	\$FFFA5/00
---------	----	----	-------	-------	-------	-------	-------	-------	------------

SYNC MODE

\$FFE26	--	--	--	--	VIS2	VIS1	VISO	GS	\$FFFA6/00
---------	----	----	----	----	------	------	------	----	------------

\$FFE27	--	--	--	--	--	--	--	--	\$FFFA7/00
---------	----	----	----	----	----	----	----	----	------------

VERTICAL SCREEN START ADDRESS

\$FFE28	--	--	--	--	--	--	--	--	\$FFFA8/00
\$FFE29	--	--	--	--	VS19	VS18	VS17	VS16	\$FFFA9/00
\$FFE2A	VS15	VS14	VS13	VS12	VS11	VS10	0	0	\$FFFAA/00
\$FFE2B	0	0	0	0	0	0	0	0	\$FFFAB/00

VERTICAL OFFSET REGISTER

\$FFE2C	--	--	--	--	--	--	--	--	\$FFFAC/00
\$FFE2D	SIGN	0	0	0	0	Y18	Y17	Y16	\$FFFAD/00
\$FFE2E	Y15	Y14	Y13	Y12	Y11	Y10	Y9	Y8	\$FFFAE/00
\$FFE2F	Y7	Y6	Y5	Y4	Y3	Y2	0	0	\$FFFAF/00

UNFOLDED ADDRESSFOLDED ADDRESS/PAGE

HORIZONTAL OFFSET

\$FFE30	--	--	--	--	--	--	--	--	\$FFF80/00
\$FFE31	--	--	--	--	--	--	--	--	\$FFF81/00
\$FFE32	SIGN	0	0	0	X10	X9	X8		\$FFF82/00
\$FFE33	X7	X6	X5	X4	X3	X2	X1	X0	\$FFF83/00

VIRTUAL SCREEN SIZE

\$FFE34	--	--	--	--	--	--	--	--	\$FFF84/00
\$FFE35	--	--	--	--	--	V18	V17	V16	\$FFF85/00
\$FFE36	V15	V14	V13	V12	V11	V10	V9	V8	\$FFF86/00
\$FFE37	V7	V6	V5	V4	V3	V2	0	0	\$FFF87/00

VIRTUAL SCREEN WIDTH

\$FFE38	--	--	--	--	--	--	--	--	\$FFF88/00
\$FFE39	--	--	--	--	--	--	--	--	\$FFF89/00
\$FFE3A	--	--	--	--	--	W10	W9	W8	\$FFFBA/00
\$FFE3B	W7	W6	W5	W4	W3	W2	0	0	\$FFFBB/00

\$FFE3C	--	--	--	--	--	--	--	--	\$FFFBC/00
\$FFE3D	--	--	--	--	--	--	--	--	\$FFFBD/00
\$FFE3E	--	--	--	--	--	--	--	--	\$FFFBE/00
\$FFE3F	--	--	--	--	--	--	--	--	\$FFFBF/00

COLOR MAPPING RAM \$00

\$FFE40	MP1	MPO	--	VEN	R3	R2	R1	RO	\$FFF80/01
\$FFE41	G3	G2	G1	GO	B3	B2	B1	BO	\$FFF81/01

COLOR MAPPING RAM \$01

\$FFE42	--	--	--	VEN	R3	R2	R1	RO	\$FFF82/01
\$FFE43	G3	G2	G1	GO	B3	B2	B1	BO	\$FFF83/01

UNFOLDED ADDRESSFOLDED ADDRESS/PAGE

COLOR MAPPING RAM \$02

\$FFE44	--	--	--	VEN	R3	R2	R1	RO	\$FFF84/01
\$FFE45	G3	G2	G1	GO	B3	B2	B1	BO	\$FFF85/01

COLOR MAPPING RAM \$03

\$FFE46	--	--	--	VEN	R3	R2	R1	RO	\$FFF86/01
\$FFE47	G3	G2	G1	GO	B3	B2	B1	BO	\$FFF87/01

COLOR MAPPING RAM \$04

\$FFE48	--	--	--	VEN	R3	R2	R1	RO	\$FFF88/01
\$FFE49	G3	G2	G1	GO	B3	B2	B1	BO	\$FFF89/01

COLOR MAPPING RAM \$05

\$FFE4A	--	--	--	VEN	R3	R2	R1	RO	\$FFF8A/01
\$FFE4B	G3	G2	G1	GO	B3	B2	B1	BO	\$FFF8B/01

COLOR MAPPING RAM \$06

\$FFE4C	--	--	--	VEN	R3	R2	R1	RO	\$FFF8C/01
\$FFE4D	G3	G2	G1	GO	B3	B2	B1	BO	\$FFF8D/01

COLOR MAPPING RAM \$07

\$FFE4E	--	--	--	VEN	R3	R2	R1	RO	\$FFF8E/01
\$FFE4F	G3	G2	G1	GO	B3	B2	B1	BO	\$FFF8F/01

COLOR MAPPING RAM \$08

\$FFE50	--	--	--	VEN	R3	R2	R1	RO	\$FFF90/01
\$FFE51	G3	G2	G1	GO	B3	B2	B1	BO	\$FFF91/01

COLOR MAPPING RAM \$09

\$FFE52	--	--	--	VEN	R3	R2	R1	RO	\$FFF92/01
\$FFE53	G3	G2	G1	GO	B3	B2	B1	BO	\$FFF93/01

UNFOLDED ADDRESS

FOLDED ADDRESS/PAGE

COLOR MAPPING RAM \$0A

\$FFE54	--	--	--	VEN	R3	R2	R1	RO	\$FFF94/01
\$FFE55	G3	G2	G1	GO	B3	B2	B1	BO	\$FFF95/01

COLOR MAPPING RAM \$0B

\$FFE56	--	--	--	VEN	R3	R2	R1	RO	\$FFF96/01
\$FFE57	G3	G2	G1	GO	B3	B2	B1	BO	\$FFF97/01

COLOR MAPPING RAM \$0C

\$FFE58	--	--	--	VEN	R3	R2	R1	RO	\$FFF98/01
\$FFE59	G3	G2	G1	GO	B3	B2	B1	BO	\$FFF99/01

COLOR MAPPING RAM \$0D

\$FFE5A	--	--	--	VEN	R3	R2	R1	RO	\$FFF9A/01
\$FFE5B	G3	G2	G1	GO	B3	B2	B1	BO	\$FFF9B/01

COLOR MAPPING RAM \$0E

\$FFE5C	--	--	--	VEN	R3	R2	R1	RO	\$FFF9C/01
\$FFE5D	G3	G2	G1	GO	B3	B2	B1	BO	\$FFF9D/01

COLOR MAPPING RAM \$0F

\$FFE5E	--	--	--	VEN	R3	R2	R1	RO	\$FFF9E/01
\$FFE5F	G3	G2	G1	GO	B3	B2	B1	BO	\$FFF9F/01

COLOR MAPPING RAM \$10

\$FFE60	--	--	--	VEN	R3	R2	R1	RO	\$FFFA0/01
\$FFE61	G3	G2	G1	GO	B3	B2	B1	BO	\$FFFA1/01

COLOR MAPPING RAM \$11

\$FFE62	--	--	--	VEN	R3	R2	R1	RO	\$FFFA2/01
\$FFE63	G3	G2	G1	GO	B3	B2	B1	BO	\$FFFA3/01

UNFOLDED ADDRESSFOLDED ADDRESS/PAGE

COLOR MAPPING RAM \$12

\$FFE64	--	--	--	VEN	R3	R2	R1	RO	\$FFFA4/01
\$FFE65	G3	G2	G1	GO	B3	B2	B1	BO	\$FFFA5/01

COLOR MAPPING RAM \$13

\$FFE66	--	--	--	VEN	R3	R2	R1	RO	\$FFFA6/01
\$FFE67	G3	G2	G1	GO	B3	B2	B1	BO	\$FFFA7/01

COLOR MAPPING RAM \$14

\$FFE68	--	--	--	VEN	R3	R2	R1	RO	\$FFFA8/01
\$FFE69	G3	G2	G1	GO	B3	B2	B1	BO	\$FFFA9/01

COLOR MAPPING RAM \$15

\$FFE6A	--	--	--	VEN	R3	R2	R1	RO	\$FFFAA/01
\$FFE6B	G3	G2	G1	GO	B3	B2	B1	BO	\$FFFAB/01

COLOR MAPPING RAM \$16

\$FFE6C	--	--	--	VEN	R3	R2	R1	RO	\$FFFAC/01
\$FFE6D	G3	G2	G1	GO	B3	B2	B1	BO	\$FFFAD/01

COLOR MAPPING RAM \$17

\$FFE6E	--	--	--	VEN	R3	R2	R1	RO	\$FFFAE/01
\$FFE6F	G3	G2	G1	GO	B3	B2	B1	BO	\$FFFAD/01

COLOR MAPPING RAM \$18

\$FFE70	--	--	--	VEN	R3	R2	R1	RO	\$FFFBO/01
\$FFE71	G3	G2	G1	GO	B3	B2	B1	BO	\$FFFBI/01

COLOR MAPPING RAM \$19

\$FFE72	--	--	--	VEN	R3	R2	R1	RO	\$FFFBI/01
\$FFE73	G3	G2	G1	GO	B3	B2	B1	BO	\$FFFBI/01

UNFOLDED ADDRESSFOLDED ADDRESS/PAGE

COLOR MAPPING RAM \$1A

\$FFE74	--	--	--	VEN	R3	R2	R1	RO	\$FFF84/01
\$FFE75	G3	G2	G1	GO	B3	B2	B1	BO	\$FFF85/01

COLOR MAPPING RAM \$1B

\$FFE76	--	--	--	VEN	R3	R2	R1	RO	\$FFF86/01
\$FFE77	G3	G2	G1	GO	B3	B2	B1	BO	\$FFF87/01

COLOR MAPPING RAM \$1C

\$FFE78	--	--	--	VEN	R3	R2	R1	RO	\$FFF88/01
\$FFE79	G3	G2	G1	GO	B3	B2	B1	BO	\$FFF89/01

COLOR MAPPING RAM \$1D

\$FFE7A	--	--	--	VEN	R3	R2	R1	RO	\$FFF8A/01
\$FFE7B	G3	G2	G1	GO	B3	B2	B1	BO	\$FFF8B/01

COLOR MAPPING RAM \$1E

\$FFE7C	--	--	--	VEN	R3	R2	R1	RO	\$FFF8C/01
\$FFE7D	G3	G2	G1	GO	B3	B2	B1	BO	\$FFF8D/01

COLOR MAPPING RAM \$1F

\$FFE7E	--	--	--	VEN	R3	R2	R1	RO	\$FFF8E/01
\$FFE7F	G3	G2	G1	GO	B3	B2	B1	BO	\$FFF8F/01

MOTOROLA
5-10

MC68487

UNFOLDED ADDRESSFOLDED ADDRESS/PAGE

\$FFE80	SEE REGISTER \$FFE00	\$FFF80/02
\$FFE81	SEE REGISTER \$FFE01	\$FFF81/02
\$FFE82	SEE REGISTER \$FFE02	\$FFF82/02
\$FFE83	SEE REGISTER \$FFE03	\$FFF83/02
\$FFE84	SEE REGISTER \$FFE04	\$FFF84/02
\$FFE85	SEE REGISTER \$FFE05	\$FFF85/02
\$FFE86	SEE REGISTER \$FFE06	\$FFF86/02
\$FFE87	SEE REGISTER \$FFE07	\$FFF87/02
\$FFE88	SEE REGISTER \$FFE08	\$FFF88/02
\$FFE89	SEE REGISTER \$FFE09	\$FFF89/02
\$FFE8A	SEE REGISTER \$FFE0A	\$FFF8A/02
\$FFE8B	SEE REGISTER \$FFE0B	\$FFF8B/02
\$FFE8C	SEE REGISTER \$FFE0C	\$FFF8C/02
\$FFE8D	SEE REGISTER \$FFE0D	\$FFF8D/02
\$FFE8E	SEE REGISTER \$FFE0E	\$FFF8E/02
\$FFE8F	SEE REGISTER \$FFE0F	\$FFF8F/02

TRUE OBJECT 0 X POSITION

\$FFE90	B/R	XZ1	XZ0	--	--	--	X9	X8	\$FFF90/02
\$FFE91	X7	X6	X5	X4	X3	X2	X1	X0	\$FFF91/02

TRUE OBJECT 0 Y POSITION

\$FFE90	OEN	YZ1	YZ0	--	--	--	Y9	Y8	\$FFF90/02
\$FFE91	Y7	Y6	Y5	Y4	Y3	Y2	Y1	Y0	\$FFF91/02

TRUE OBJECT 1 X POSITION

\$FFE94	B/R	XZ1	XZ0	--	--	--	X9	X8	\$FFF94/02
\$FFE95	X7	X6	X5	X4	X3	X2	X1	X0	\$FFF95/02

UNFOLDED ADDRESS

FOLDED ADDRESS/PAGE

TRUE OBJECT 1 Y POSITION

\$FFE96	OEN	YZ1	YZ0	--	--	--	Y9	Y8	\$FFF96/02
\$FFE97	Y7	Y6	Y5	Y4	Y3	Y2	Y1	Y0	\$FFF97/02

TRUE OBJECT 2 X POSITION

\$FFE98	B/R	XZ1	XZ0	--	--	--	X9	X8	\$FFF98/02
\$FFE99	X7	X6	X5	X4	X3	X2	X1	X0	\$FFF99/02

TRUE OBJECT 2 Y POSITION

\$FFE9A	OEN	YZ1	YZ0	--	--	--	Y9	Y8	\$FFF9A/02
\$FFE9B	Y7	Y6	Y5	Y4	Y3	Y2	Y1	Y0	\$FFF9B/02

TRUE OBJECT 3 X POSITION

\$FFE9C	B/R	XZ1	XZ0	--	--	--	X9	X8	\$FFF9C/02
\$FFE9D	X7	X6	X5	X4	X3	X2	X1	X0	\$FFF9D/02

TRUE OBJECT 3 Y POSITION

\$FFE9E	OEN	YZ1	YZ0	--	--	--	Y9	Y8	\$FFF9E/02
\$FFE9F	Y7	Y6	Y5	Y4	Y3	Y2	Y1	Y0	\$FFF9F/02

TRUE OBJECT 4 X POSITION

\$FFEA0	B/R	XZ1	XZ0	--	--	--	X9	X8	\$FFFA0/02
\$FFEA1	X7	X6	X5	X4	X3	X2	X1	X0	\$FFFA1/02

TRUE OBJECT 4 Y POSITION

\$FFEA2	OEN	YZ1	YZ0	--	--	--	Y9	Y8	\$FFFA2/02
\$FFEA3	Y7	Y6	Y5	Y4	Y3	Y2	Y1	Y0	\$FFFA3/02

UNFOLDED ADDRESSFOLDED ADDRESS/PAGE

TRUE OBJECT 5 X POSITION

\$FFEA4	B/R	XZ1	XZ0	--	--	--	X9	X8	\$FFFA4/02
\$FFEA5	X7	X6	X5	X4	X3	X2	X1	X0	\$FFFA5/02

TRUE OBJECT 5 Y POSITION

\$FFEA6	OEN	YZ1	YZ0	--	--	--	Y9	Y8	\$FFFA6/02
\$FFEA7	Y7	Y6	Y5	Y4	Y3	Y2	Y1	Y0	\$FFFA7/02

TRUE OBJECT 6 X POSITION

\$FFEA8	B/R	XZ1	XZ0	--	--	--	X9	X8	\$FFFA8/02
\$FFEA9	X7	X6	X5	X4	X3	X2	X1	X0	\$FFFA9/02

TRUE OBJECT 6 Y POSITION

\$FFEA9A	OEN	YZ1	YZ0	--	--	--	Y9	Y8	\$FFFAA/02
\$FFEA9B	Y7	Y6	Y5	Y4	Y3	Y2	Y1	Y0	\$FFFAB/02

TRUE OBJECT 7 X POSITION

\$FFEAC	B/R	XZ1	XZ0	--	--	--	X9	X8	\$FFFAC/02
\$FFEAD	X7	X6	X5	X4	X3	X2	X1	X0	\$FFFAD/02

TRUE OBJECT 7 Y POSITION

\$FFEAE	OEN	YZ1	YZ0	--	--	--	Y9	Y8	\$FFFAE/02
\$FFEAF	Y7	Y6	Y5	Y4	Y3	Y2	Y1	Y0	\$FFFaf/02

OBJECT 0 NAME REGISTER

\$FFEB0	N7	N6	N5	N4	N3	N2	N1	NO	\$FFFBO/02
---------	----	----	----	----	----	----	----	----	------------

OBJECT 1 NAME REGISTER

\$FFEB1	N7	N6	N5	N4	N3	N2	N1	NO	\$FFFb1/02
---------	----	----	----	----	----	----	----	----	------------

UNFOLDED ADDRESS

FOLDED ADDRESS/PAGE

OBJECT 2 NAME REGISTER

\$FFEB2	N7	N6	N5	N4	N3	N2	N1	NO	\$FFFB2/02
---------	----	----	----	----	----	----	----	----	------------

OBJECT 3 NAME REGISTER

\$FFEB3	N7	N6	N5	N4	N3	N2	N1	NO	\$FFFB3/02
---------	----	----	----	----	----	----	----	----	------------

OBJECT 4 NAME REGISTER

\$FFEB4	N7	N6	N5	N4	N3	N2	N1	NO	\$FFFB4/02
---------	----	----	----	----	----	----	----	----	------------

OBJECT 5 NAME REGISTER

\$FFEB5	N7	N6	N5	N4	N3	N2	N1	NO	\$FFFB5/02
---------	----	----	----	----	----	----	----	----	------------

OBJECT 6 NAME REGISTER

\$FFEB6	N7	N6	N5	N4	N3	N2	N1	NO	\$FFFB6/02
---------	----	----	----	----	----	----	----	----	------------

OBJECT 7 NAME REGISTER

\$FFEB7	N7	N6	N5	N4	N3	N2	N1	NO	\$FFFB7/02
---------	----	----	----	----	----	----	----	----	------------

\$FFEB8	--	--	--	--	--	--	--	--	\$FFFB8/02
---------	----	----	----	----	----	----	----	----	------------

\$FFEB9	--	--	--	--	--	--	--	--	\$FFFB9/02
---------	----	----	----	----	----	----	----	----	------------

\$FFEBA	--	--	--	--	--	--	--	--	\$FFFBA/02
---------	----	----	----	----	----	----	----	----	------------

\$FFEBB	--	--	--	--	--	--	--	--	\$FFFBB/02
---------	----	----	----	----	----	----	----	----	------------

\$FFEBC	--	--	--	--	--	--	--	--	\$FFFBC/02
---------	----	----	----	----	----	----	----	----	------------

\$FFEBD	--	--	--	--	--	--	--	--	\$FFFBD/02
---------	----	----	----	----	----	----	----	----	------------

\$FFEBE	--	--	--	--	--	--	--	--	\$FFFBE/02
---------	----	----	----	----	----	----	----	----	------------

\$FFEBF	--	--	--	--	--	--	--	--	\$FFFBF/02
---------	----	----	----	----	----	----	----	----	------------

MOTOROLA

5-14

MC68487

5.2 RMC CONTROL REGISTERS

The following paragraphs discuss the RMC control registers. Unused bits are shown as blanks on the register diagrams; they read as zeros. The state of all registers' bits are undefined after a reset except as noted in the following paragraphs or in SECTION 3 RESET AND INITIALIZATION.

5.2.1 \$FFE00 - Memory Map

- B7, B6 MP1 and MPO. These bits are used to select pages of the register map in the folded option. They have no effect in the unfolded option.
B5 UF. Selects folded or unfolded map.
B4 M2. Selects machine 1 or machine 2.

All bits are cleared upon reset except B5.

5.2.2 \$FFE01 - Display Data Mode

- B7 BP. Selects bit-plane (1) or list modes (0).
B6-B4 LM2 through LMO. These bits select the list mode to be used.
(See Table 5-1.)

Table 5-1. List Mode Encoding

LM2	LM1	LMO	List Mode
0	0	0	0
0	0	1	1
0	1	0	2
0	1	1	3
1	0	0	4
1	0	1	5
1	1	0	Reserved
1	1	1	Reserved

B3, B2 LPR1 and LPRO. These bits define the number of video lines per character row. (See Table 5-2.)

Table 5-2. Lines-per-Row Encoding

LPR1	LPRO	Number of Lines
0	0	8
0	1	10
1	0	12
1	1	16

B1, B0 BPP1 and BPP0. These bits define the number of bits of memory to be used for each pel displayed. (See Table 5-3.)

Table 5-3. Bits-per-Pel Encoding

BPP1	BPP0	Bits per Pel
0	0	1
0	1	2
1	0	4
1	1	Reserved

5.2.3 \$FFE02 - Interrupt Control/Status

When this register is written, it enables or disables certain interrupts. When read, it gives the status of the interrupt bits. Reading this register clears any set bits.

B7 IPT. Enables or disables all RMC generated interrupts.

B6, B5 Not used.

B4 RTI. This bit is set when the real-time interrupt (RTI) input goes low.

- B3 RTO. The real-time output bit is set when the current screen X and Y coordinates match the RTO registers.
- B2 OFN. The object finish bit is set when the RMS finishes displaying a true object.
- B1 BLK. The blanking bit is set when the right-hand border is reached by the last active video line.
- B0 COL. The collision bit is set when a true object collides with another true object or with a fixed object, if both objects have collision enabled.

5.2.4 \$FFE03 - Border Color

- B7 WC. The wrap bit controls what is displayed when the visible screen is scrolled off the edges of the virtual screen.
- B6 MAPA. This bit selects one of two types of memory maps which are related to the S bus, a ROM intensive system map or a RAM intensive system map.
- B5 DV. The display-video bit is used as a general video enable. When it is reset, a border is displayed over the whole screen.
- B4-B0 BC4 through BCO. These bits define the border color by pointing to one of the CMR registers.

5.2.5 \$FFE04 - Object Available

- B7-B0 07A through 00A. The object-available register is used to indicate which true objects are not currently in use. Each bit corresponds to an object. If a bit is set, the object is available. A bit is cleared by writing to the corresponding object's X coordinate register or when a match occurs between the current scan line count and the object's Y coordinate.

5.2.6 \$FFEO5 - Paging

B3-B0

PG3 through PG0. These page bits allow the MC6809E to address the entire 1 megabyte memory space.

B4

SWAP. The swap bit is used to invert the MC6809E's A15 to flip between the bottom and top halves of the 64K byte block. This register has no effect on the M68000 Family MPUs.

5.2.7 \$FFEO6 - Page Independent Blocks (PIB)

This register is used by the MC6809E to fix two pieces of the memory map at the top and bottom of the MPU's 64K memory map regardless of the page. Three UPI and three LPI bits define the size of the memory block to be fixed in the top half and the bottom half respectively. Bits 7 and 4 (UEN, LEN) enable the PIB function for the upper and lower PIBs respectively. With a M68000 Family interface, bit 0 (VEC) selects DRAM (1) or ROM (0) for the exception vectors. (See Table 5-4.) Following a system reset, both PIBs are disabled.

Table 5-4. PIB Size Encoding

Bit 2	Bit 1	Bit 0	PIB Size
0	0	0	256 Bytes
0	0	1	512 Bytes
0	1	0	1K Bytes
0	1	1	2K Bytes
1	0	0	4K Bytes
1	0	1	8K Bytes
1	1	0	16K Bytes
1	1	1	32K Bytes

5.2.8 \$FFE07 - Vertical Scroll

This register is used with the vertical offset register to do smooth vertical scrolling while in list mode. It has no effect in bit-plane mode.

B3-B0 VSC3 through VSC0. These bits select which scan line of the character row is to be used to start the character row at the top of the visible screen.

B4 DHP. The double-high preset bit is used with vertical scrolling when the double-high attribute is selected.

5.2.9 \$FFE08 - Horizontal Scroll

This register is used with the horizontal offset register to do smooth horizontal scrolling. It is used in both bit-plane and list modes.

B3-B0 HSC3 through HSC0. These bits select the number of pixels to be scrolled. In HRES 6 and 7, all four bits are used. In other modes, HSC3 is not used.

B4 DWP. The double-wide preset bit is used to signal the RMS to begin displaying the second half of the double-wide character if that attribute is selected.

5.2.10 \$FFE0A, \$FFE0B - DRC Image Table Start Address

This register selects a 1K byte block on a 1K boundary out of the 1 megabyte memory space as the start address of the DRC image table.

5.2.11 \$FFEOC, \$FFEOD - True Object Image Table Start Address

This register functions the same way as \$FFEOA, \$FFEOB except that it defines the start of the true object image table.

5.2.12 \$FFEOF, \$FFEOF - Fixed Object Image Table Start Address

This register functions the same way as \$FFEOA, \$FFEOB except that it defines the start address of the fixed object image table.

5.2.13 \$FFE10-\$FE17 - Collision Status

Each object has its own register. These register bits are used to report collisions between true objects. Fixed object collision reporting is reported by the Nth bit (N = 0 through 8) in each register.

5.2.14 \$FFE18 - Collision Enable

The bits in this register enable or disable object collision.

5.2.15 \$FFE1C-\$FE1F - Real-Time Output

Two registers hold the X and Y coordinates which are to be compared with the position of the electron beam. When a match occurs, the RTO bit is set in the interrupt status register and can cause an interrupt if enabled.

5.2.16 \$FFE20-\$FE23 - Real-Time Input

These two registers are used in conjunction with the RTI input. When a falling edge occurs on RTI, the CRT's beam X and Y coordinates are loaded into these registers.

5.2.17 \$FE24 - Memory Organization

This register defines the type and organization of DRAM memory.

B7-B4 MTYP3 through MTYPO. The memory type bits are defined in Table 5-5.

Table 5-5. Memory Type Encoding

MTYP3	MTYP2	MTYP1	MTYPO	DRAM Type
0	0	0	0	16Kx1, 8 Bits Wide
0	0	0	1	16Kx4, 8 Bits Wide
0	0	1	0	64Kx1, 8 Bits Wide
0	1	0	0	256Kx1, 8 Bits Wide
1	0	0	0	16Kx1, 16 Bits Wide
1	0	0	1	16Kx4, 16 Bits Wide
1	0	1	0	64Kx1, 16 Bits Wide
1	1	0	0	256Kx1, 16 Bits Wide
All Other Combinations				Not Defined

B3, B2 DB1 and DBO. These bits define the number of DRAM banks (see Table 3-6).

Table 5-6. DRAM Bank Encoding

DB1	DB0	Number of Banks
0	0	1
0	1	2
1	0	Reserved
1	1	4

5.2.16 \$FFE20-\$FE23 - Real-Time Input

These two registers are used in conjunction with the RTI input. When a falling edge occurs on RTI, the CRT's beam X and Y coordinates are loaded into these registers.

5.2.17 \$FE24 - Memory Organization

This register defines the type and organization of DRAM memory.

B7-B4 MTYP3 through MTYPO. The memory type bits are defined in Table 5-5.

Table 5-5. Memory Type Encoding

MTYP3	MTYP2	MTYP1	MTYPO	DRAM Type
0	0	0	0	16Kx1, 8 Bits Wide
0	0	0	1	16Kx4, 8 Bits Wide
0	0	1	0	64Kx1, 8 Bits Wide
0	1	0	0	256Kx1, 8 Bits Wide
1	0	0	0	16Kx1, 16 Bits Wide
1	0	0	1	16Kx4, 16 Bits Wide
1	0	1	0	64Kx1, 16 Bits Wide
1	1	0	0	256Kx1, 16 Bits Wide
All Other Combinations				Not Defined

B3, B2 DB1 and DBO. These bits define the number of DRAM banks (see Table 5-6).

Table 5-6. DRAM Bank Encoding

DB1	DB0	Number of Banks
0	0	1
0	1	2
1	0	Reserved
1	1	4

5.2.19 \$FFE26 - Sync Mode

B3-B1 VIS2 through VISO. These bits determine how the SYNC pin is used. (See Table 5-8.)

Table 5-8. Sync Encoding

VIS2	VIS1	VISO	Direction	Signal
0	0	0	Output	Vertical Sync
0	0	1	Output	Horizontal Sync
0	1	x	Output	Composite Sync
1	x	x	Input	Field Sync

B0 GS. This bit is used to select whether the RMC's G output pin will have only video information, or video information and sync information.

5.2.20 \$FFE28-\$FE2B - Virtual Screen Start Address

The virtual screen start address is a 4-byte register that holds the beginning address of the virtual screen memory.

5.2.21 \$FFE2C-\$FE2F - Vertical Offset

The vertical offset register is a 4-byte register that (along with the horizontal offset register) places the displayed screen within the virtual screen. If the sign bit (B19) is on, then the vertical offset register should contain a 32-bit twos complement number whose units are bytes. If the sign bit (B19) is off, then a positive number is input.

5.2.18 \$FFE25 - Video Operation

- B7 IS. Selects interlace (1) or non-interlace (0) sync video.
- B6 ID. Selects interlaced (1) or non-interlaced (0) data.
- B5-B3 VRES2 through VRES0. Select the vertical resolution between 64 and 250 lines/field. (See Table 5-7.)
- B2-B0 HRES2-HRES0. Select the horizontal resolution between 64 and 640 pixels per line. (See Table 5-7.)

Table 5-7. Horizontal and Vertical Resolution Combinations

VRES2-0	RES	HRES2-0							
		RES							
		000	001	010	011	100	101	110	111
		64	128	256	256	320	*	512	640
000	64	2	2				R		
001	96		2				E		
010	192		2	1,2	1	1	S		
011	200			1	1	1	E	1	1
100	210			1	1	1	R	1	1
101	240 (PAL)			1	1	1	V	1	1
111	250 (PAL)			1	1	1	E	1	1
							D	1	1

1 = Machine 1

2 = Machine 2

5.2.22 \$FFE30-\$FE33 - Horizontal Offset

The horizontal offset register is a 4-byte register which is used to position the left edge of the displayed screen within the virtual screen. If the sign bit (B19) is off, then a positive number is input. If the sign bit (B19) is on, then the horizontal offset register should contain a twos complement number.

5.2.23 \$FE34-\$FE37 - Virtual Screen Size

This is a 4-byte register that sets the virtual screen size in bytes. The size must be at least the size of the visible screen and a maximum of 512K bytes.

5.2.24 \$FE38-\$FE3B - Virtual Screen Width

This 4-byte register holds the virtual screen's line width in bytes. The lines must be an integer number of memory cycles wide, and they must be at least as wide as the visible screen.

$$\text{Bit Plane LL} = (\text{Pels/Line}) * (\text{Bits/Pel}) * (1 \text{ Byte}/8 \text{ Bits})$$

$$\text{List Plane LL} = (\text{Characters/Line}) * (\text{Bytes/Character})$$

5.2.25 \$FE40-\$FE7F - Color-Mapping RAM (CMR)

There are 32 different CMR registers which are identical in function and format except CMR00. Each CMR register occupies two bytes. The MPU can write the CMR at any time; however, if it is accessed during active video and the RMS is trying to access it at the same time, a glitch may appear on the screen. Writing the CMR during retrace should eliminate that. The VEN bit controls the video enable output pin (VIDEN) when the particular CMR is

accessed by the video data. The R, G, and B bits are used as inputs to D/A converters that drive the R, G, and B output pins. Bits 15 and 14 of CMR00 are remapped from the paging register to change pages while in the folded map.

5.2.26 \$FFE90-\$FFEA9 - True Object Position

Each of the eight true objects has an X coordinate and a Y coordinate register which are used to position the object on the screen. These registers are 2-bytes each.

The X coordinate's B/R bit selects between the bit-plane mode and the run-length object display format. The X coordinate's XZ1 and XZ0 set the horizontal zoom factor. The Y coordinate's OEN is the object-enable bit. The YZ1 and YZ0 select the vertical zoom factor. (See Table 5-9.)

Table 5-9. Horizontal and Vertical Zoom Encoding

*Z1	*Z0	Zoom Factor
0	0	1 X
0	1	2 X
1	0	4 X
1	1	8 X

* = X for Horizontal

* = Y for Vertical

5.2.27 \$FFE80-\$FFE87 - True Object Names

There is one name register for each true object. The 8-bit name is used to calculate the image table entry.

SECTION 6

DISPLAY MODES

The RMS can operate in one of two display modes, bit-plane or list mode. These modes are described in the following paragraphs.

6.1 BIT-PLANE MODE

The bit-plane mode is selected by setting bit 7 of control register \$FFE01. In this mode, each pel on the screen is treated as a separate entity and is described by video memory. Each pel can be described by one, two, or four bits which allows 2, 4, or 16 colors per pel respectively. In HRES 6 and 7, four bits per pel cannot be used because of memory access limitations.

Each memory byte is broken into two, four, or eight pieces depending on the bit/pel number. For example, for two bits per pel, the byte is divided into four pieces. The two most-significant bits (bits 6 and 7) are combined with three leading zeros to create a 5-bit CMR address which defines the pel color. The next two bits define the next pel and so on. The amount of memory needed to display one full screen is calculated as follows:

$$\text{Memory (bytes)} = (\text{horizontal resolution} * \text{vertical resolution} * \text{bits/pel})/8$$

6.2 LIST MODE

In list mode, characters can be 8 or 16 pels wide and 8, 10, 12, or 16 pels high depending on the list mode selected and the number of lines per row programmed in control register \$FFE01. This screen is filled with rows of characters which can be 32, 40, or 80 characters wide depending on the

horizontal resolution. The number of character rows per screen is a function of lines per row (character) and vertical resolution.

The RMC has six list modes which can use four basic character types:

Alphanumerics,

Mosaics,

Dynamically Redefinable Characters (DRC), and

Fixed Objects.

The list modes differ in the number of bytes used to represent a character, the types of characters allowed, and the types of attributes each character may have. These list modes use the memory in two ways: screen memory and image tables. Depending on the list mode, the screen memory contains one, two, or three bytes for each character which is used to define the character type, its location in memory, and its attributes. Image tables are used to give a pel-by-pel description of the characters to be displayed. The RMC contains internal ROM image tables for alphanumerics and mosaics, and the user can define the image tables for DRC's and fixed objects in external RAM.

The amount of memory needed to display one full screen is list mode dependent. The bit descriptions of each mode follows.

NOTE

Some bits are always ones or zeros. These bits are used by the RMS to determine the type of character. The following table shows which HRES modes may be used with each of the list modes.

List Mode	HRES Mode
0	2, 3, 4
1	2, 3, 4
2	6, 7
3	6, 7
4	2, 3, 4
5	6, 7

6.2.1 List Mode 0

There are seven character codes to define an alphanumeric character, but only 96 of the 128 characters are valid (see Table 6-1). The other 32 characters are used as DRC's. The DRC's are treated as alphanumeric characters, except their pattern data is stored in DRAM.

This mode uses default foreground and background CMRs of CMROF and CMROO respectively.

Table 6-1. List Mode 0 Characters

Bit	Alphanumeric	DRC	Mosaic
7	Always 0	Always 0	Always 1
6	Character Code 6 (MSB)	Always 0	Character Code 6 (MSB)
5	Character Code 5	Always 0	Separation
4	Character Code 4	Character Code 4 (MSB)	Character Code 4
3	Character Code 3	Character Code 3	Character Code 3
2	Character Code 2	Character Code 2	Character Code 2
1	Character Code 1	Character Code 1	Character Code 1
0	Character Code 0 (LSB)	Character Code 0 (LSB)	Character Code 0 (LSB)

6.2.2 List Mode 1

The lower 32 alphanumeric characters are coded as DRC's. Fixed objects A and B differ in that the most-significant bits of the two bytes are used to complete the specification of some of the attributes. Bit 7 of the first byte is the most-significant CMR offset bit (CMR 4), color collision bit (color collision 1), and priority bit (priority 2). Bit 7 of the second byte is the most-significant bit of the character code. Therefore, fixed objects A and B are in separate halves of the fixed object list and they have different attributes.

Fixed objects A and B support 64 characters each (see Tables 6-2 and 6-3). The total number of characters available in list mode 1 is 128 fixed objects, 96 alphanumerics, 64 mosaics, and 32 DRC's.

Table 6-2. List Mode 1 Alphanumerics, DRCs, and Mosaics

Bit	Alphanumeric	DRC	Mosaic
Byte 1	7 Always 0	Always 0	Always 1
	6 Character Code 6 (MSB)	Always 0	Character Code 6 (MSB)
	5 Character Code 5	Always 0	Separation
	4 Character Code 4	Character Code 4 (MSB)	Character Code 4
	3 Character Code 3	Character Code 3	Character Code 3
	2 Character Code 2	Character Code 2	Character Code 2
	1 Character Code 1	Character Code 1	Character Code 1
	0 Character Code 0 (LSB)	Character Code 0 (LSB)	Character Code 0 (LSB)
Byte 2	7 Always 0	Always 0	Always 1
	6 Flash 1	Flash 1	Flash 1
	5 Foreground 2 (MSB)	Foreground 2 (MSB)	Foreground 2 (MSB)
	4 Foreground 1	Foreground 1	Foreground 1
	3 Foreground 0 (LSB)	Foreground 0 (LSB)	Foreground 0 (LSB)
	2 Background 2 (MSB)	Background 2 (MSB)	Background 2 (MSB)
	1 Background 1	Background 1	Background 1
	0 Background 0 (LSB)	Background 0 (LSB)	Background 0 (LSB)

Table 6-3. List Mode 1 Fixed Objects

Bit	Fixed A	Fixed B
Byte 1	7 Always 1	Always 0
	6 CMR Offset 3	CMR Offset 3
	5 CMR Offset 2	CMR Offset 2
	4 CMR Offset 1	CMR Offset 1
	3 Color Collision (MSB)	Color Collision (MSB)
	2 Priority 1	Priority 1
	1 Priority 0	Priority 0
	0 Collision Enable	Collision Enable
Byte 2	7 Always 0	Always 1
	6 Flash 1	Flash 1
	5 Character Code 5	Character Code 5
	4 Character Code 4	Character Code 4
	3 Character Code 3	Character Code 3
	2 Character Code 2	Character Code 2
	1 Character Code 1	Character Code 1
	0 Character Code 0 (LSB)	Character Code 0 (LSB)

6.2.3 List Mode 2

List mode 2 must be used with one of the high resolution modes (HRES modes 6 or 7). It displays either two 8-pel wide ASCII characters, or one 16-pel wide redefinable character during each memory cycle (40 memory cycles per character row). In this mode, the lower 32 characters in the internal character ROM are the international characters. Refer to Tables 6-4 and 6-5.

Table 6-4. List Mode 2 Alphanumerics and Mosaics

Bit	Alphanumeric	Mosaic
Byte 1	7 Always 0	Always 1
	6 Character Code 6-1 (MSB)	Character Code 6-1 (MSB)
	5 Character Code 5-1	Separation 1
	4 Character Code 4-1	Character Code 4-1
	3 Character Code 3-1	Character Code 3-1
	2 Character Code 2-1	Character Code 2-1
	1 Character Code 1-1	Character Code 1-1
	0 Character Code 0-1 (LSB)	Character Code 0-1 (LSB)
Byte 2	7 Always 0	Always 1
	6 Character Code 6-2 (MSB)	Character Code 6-2 (MSB)
	5 Character Code 5-2	Separation 2
	4 Character Code 4-2	Character Code 4-2
	3 Character Code 3-2	Character Code 3-2
	2 Character Code 2-2	Character Code 2-2
	1 Character Code 1-2	Character Code 1-2
	0 Character Code 0-2 (LSB)	Character Code 0-2 (LSB)

Table 6-5. List Mode 2 DRCs and Fixed Objects

Bit	Fixed	DRC
Byte 1	7 Always 1	Always 0
	6 Character Code 13 (MSB)	Character Code 13 (MSB)
	5 Character Code 12	Character Code 12
	4 Character Code 11	Character Code 11
	3 Character Code 10	Character Code 10
	2 Character Code 9	Character Code 9
	1 Character Code 8	Character Code 8
	0 Character Code 7	Character Code 7
Byte 2	7 Always 0	Always 1
	6 Character Code 6	Character Code 6
	5 Character Code 5	Character Code 5
	4 Character Code 4	Character Code 4
	3 Character Code 3	Character Code 3
	2 Character Code 2	Character Code 2
	1 Character Code 1	Character Code 1
	0 Character Code 0 (LSB)	Character Code 0 (LSB)

6.2.4 List Mode 3

Mode 3 supports 16,384 DRC's and 512 fixed objects. It is a games-oriented mode that uses horizontal resolution modes 6 and 7. Refer to Table 6-6.

Table 6-6. List Mode 3 Characters

Bit	DRC	Fixed
Byte 1	7 Always 0	Always 1
	6 Character Code 13 (MSB)	Color Collision 1
	5 Character Code 12	Color Collision 0
	4 Character Code 11	Priority 2
	3 Character Code 10	Priority 1
	2 Character Code 9	Priority 0
	1 Character Code 8	Character Code 8 (MSB)
	0 Character Code 7	Character Code 7
Byte 2	7 Always 1	Always 0
	6 Character Code 6	Character Code 6
	5 Character Code 5	Character Code 5
	4 Character Code 4	Character Code 4
	3 Character Code 3	Character Code 3
	2 Character Code 2	Character Code 2
	1 Character Code 1	Character Code 1
	0 Character Code 0 (LSB)	Character Code 0 (LSB)
Byte 3	7 2X W	Collision Enable
	6 CMR Offset 4	CMR Offset 4
	5 CMR Offset 3	CMR Offset 3
	4 CMR Offset 2	CMR Offset 2
	3 CMR Offset 1	CMR Offset 1
	2 2X H	Shading
	1 Flash 1	Flash 1
	0 Flash 0	Color/Resolution

6.2.5 List Mode 4

List mode 4 is a three-byte-per-character mode that mixes several types of characters and provides all the RMS' attributes that apply to games. It uses the low horizontal resolution modes (HRES2-4). Refer to Tables 6-7 and 6-8.

Table 6-7. List Mode 4 Alphanumerics and Mosaics

Bit	Alphanumeric	Mosaic
Byte 1	7 Always 0	Always 1
	6 Character Code 6 (MSB)	Character Code 6 (MSB)
	5 Character Code 5	Separation
	4 Character Code 4	Character Code 4
	3 Character Code 3	Character Code 3
	2 Character Code 2	Character Code 2
	1 Character Code 1	Character Code 1
	0 Character Code 0 (LSB)	Character Code 0 (LSB)
Byte 2	7 Always 0	Always 1
	6 2X W	2X W
	5 Spare	Spare
	4 Invert	Invert
	3 Foreground 3 (MSB)	Foreground 3 (MSB)
	2 Foreground 2	Foreground 2
	1 Foreground 1	Foreground 1
	0 Foreground 0	Foreground 0
Byte 3	7 Underline	Mosaic 4
	6 Background 3 (MSB)	Background 3 (MSB)
	5 Background 2	Background 2
	4 Background 1	Background 1
	3 Background 0	Background 0
	2 2xH	2xH
	1 Flash 1 (MSB)	Flash 1 (MSB)
	0 Flash 0 (LSB)	Flash 0 (LSB)

Table 6-8. List Mode 4 DRCs and Fixed Objects

Bit	DRC (Alphanumeric)	Fixed	DRC
Byte 1	7 Always 0	Always 1	Always 0
	6 Always 0	Color Collision 1	Color/Resolution
	5 Always 0	Color Collision 0	Invert
	4 Character Code 4	Priority 2	2xW
	3 Character Code 3	Priority 1	Spare
	2 Character Code 2	Priority 0	Spare
	1 Character Code 1	Character Code 8 (MSB)	Character Code 8 (MSB)
	0 Character Code 0 (LSB)	Character Code 7	Character Code 7
Byte 2	7 Always 0	Always 0	Always 1
	6 2X W	Character Code 6	Character Code 6
	5 Spare	Character Code 5	Character Code 5
	4 Invert	Character Code 4	Character Code 4
	3 Foreground 3 (MSB)	Character Code 3	Character Code 3
	2 Foreground 2	Character Code 2	Character Code 2
	1 Foreground 1	Character Code 1	Character Code 1
	0 Foreground 0	Character Code 0 (LSB)	Character Code 0 (LSB)
Byte 3	7 Underline	Collision Enable	Underline
	6 Background 3 (MSB)	CMR Offset 4	CMR Offset 4
	5 Background 2	CMR Offset 3	CMR Offset 3
	4 Background 1	CMR Offset 2	CMR Offset 2
	3 Background 0	CMR Offset 1	CMR Offset 1
	2 2xH	Shading	2xH
	1 Flash 1 (MSB)	Flash 1 (MSB)	Flash 1 (MSB)
	0 Flash 0 (LSB)	Flash 0 (LSB)	Flash 0 (LSB)

6.2.6 List Mode 5

List mode 5 is designed to be used with HRES modes 6 or 7. It displays either two 8-pel wide alphanumeric (or mosaic) characters or one 16-pel wide redefinable character in each memory cycle. The DRC's are one bit per pel regardless of the display data mode register setting. The second least-significant bit is ORed with CMR1 to obtain a four-color availability in a DRC. For alphanumerics and mosaics, the least-significant bits of the third byte (bits 0-3) apply to the character in the first byte, and the most-significant bits (bits 4-7) apply to the character in the second byte. In this mode, the lower 32 characters in the internal character ROM are the international character set. Refer to Tables 6-9 and 6-10.

Table 6-9. List Mode 5 Alphanumeric and Mosaics

Bit	Alphanumeric	Mosaic
Byte 1	7 Always 0	Always 1
	6 Character Code 6-1 (MSB)	Character Code 6-1 (MSB)
	5 Character Code 5-1	Separation 1
	4 Character Code 4-1	Character Code 4-1
	3 Character Code 3-1	Character Code 3-1
	2 Character Code 2-1	Character Code 2-1
	1 Character Code 1-1	Character Code 1-1
	0 Character Code 0-1 (LSB)	Character Code 0-1 (LSB)
Byte 2	7 Always 0	Always 1
	6 Character Code 6-2 (MSB)	Character Code 6-2 (MSB)
	5 Character Code 5-2	Separation 2
	4 Character Code 4-2	Character Code 4-2
	3 Character Code 3-2	Character Code 3-2
	2 Character Code 2-2	Character Code 2-2
	1 Character Code 1-2	Character Code 1-2
	0 Character Code 0-2 (LSB)	Character Code 0-2 (LSB)
Byte 3	7 Underline (Character 2)	Mosaic 4 (Character 2)
	6 Invert	Invert
	5 Flash 1	Flash 1
	4 CMR Offset 4 (MSB)	CMR Offset (MSB)
	3 Underline (Character 1)	Mosaic 4 (Character 1)
	2 Invert	Invert
	1 Flash 1	Flash 1
	0 CMR Offset 4 (MSB)	CMR Offset (MSB)

Table 6-10. List Mode 5 DRCs and Fixed Objects

Bit	DRC	Fixed
Byte 1	7 Always 0	Always 1
	6 Character Code 13 (MSB)	Character Code 13 (MSB)
	5 Character Code 12	Character Code 12
	4 Character Code 11	Character Code 11
	3 Character Code 10	Character Code 10
	2 Character Code 9	Character Code 9
	1 Character Code 8	Character Code 8
	0 Character Code 7	Character Code 7
Byte 2	7 Always 1	Always 0
	6 Character Code 6	Character Code 6
	5 Character Code 5	Character Code 5
	4 Character Code 4	Character Code 4
	3 Character Code 3	Character Code 3
	2 Character Code 2	Character Code 2
	1 Character Code 1	Character Code 1
	0 Character Code 0 (LSB)	Character Code 0 (LSB)
Byte 3	7 Underline	Underline
	6 Invert	Invert
	5 CMR Offset 3	CMR Offset 3
	4 CMR Offset 2	CMR Offset 2
	3 CMR Offset 1	CMR Offset 1
	2 2xH	2xH
	1 Flash 1 (MSB)	Flash 1 (MSB)
	0 Flash 0 (LSB)	Flash 0 (LSB)

6.3 CHARACTER TYPES

Attributes by character and list mode are given in Figure 6-1 and are defined in the following paragraphs.

Attributes	List Mode															Fixed Objects							
	Alphanumerics					Mosaics					DRCs												
	0	1	2	4	5	0	1	2	4	5	0	1	2	3	4A	4	5	1	2	3	4	5	
Character Codes	96	96	128	96	128	64	64	64	16/64	16/64	32	32	16K	16K	32	512	16K	128	16K	512	512	16	
Flash Rates	1			3	1		1		3	1		1		3	3	3	3	3	1	1	3	3	
Foreground and Background Colors																							
CMR Offset						2				2					16		16	8	16		16	16	8
Priority																			8	8	8	8	
Color Collision																			4	4	4	4	
Invert			x	x			x	x					x	x	x							x	
Underline		x	x										x	x	x							x	
2xH	x					x					x	x	x	x								x	
2xW	x					x					x	x	x	x								x	
Separation			x	x	x	x	x	x					x									x	
Color/Res													x									x	
Shading																			x	x		x	
Collision Enable																		x	x	x			
Mosaic 4/6							x	x															

Figure 6-1. Attributes by Character and List Mode

6.3.1 Alphanumeric Characters

The RMC has 96 ASCII and 32 international characters in an internal ROM (Figure 6-2). The characters are 5×8 pels and are positioned in an $8 \times n$ ($n = 8, 10, 12, 16$) pel block as shown in Figure 6-3. Some lower case characters are displayed with single line descenders in eight lines per row; two lines in 10, 12, or 16 lines per row. Alphanumeric characters are stored as one bit per pel. A one represents the foreground color and a zero represents the background color.

6.3.2 Mosaic Characters

The RMC has two types of mosaics: mosaic 4 and mosaic 6 (Figure 6-4). In mosaic 4 the character space is divided into four equal size blocks and is described by four bits of a byte. Mosaic 6 is similar to mosaic 4 except that the character block is divided into six blocks. Mosaic 6 characters have a separation attribute to separate the sub-blocks from each other (Figure 6-5).

6.3.3 Redefinable Characters

These characters' pel description is defined by the user and stored in image tables. One, two, or four bits per pel can be used, allowing the character block to have multiple colors. DRC's may have attributes like those of alphanumeric characters. The DRC's are 8 or 16 pels wide depending on the list mode selected. They must have the number of rows of pels as specified by the lines-per-row bits (\$FFE01). The memory space where these DRCs are written is defined by the DRC start address control register.

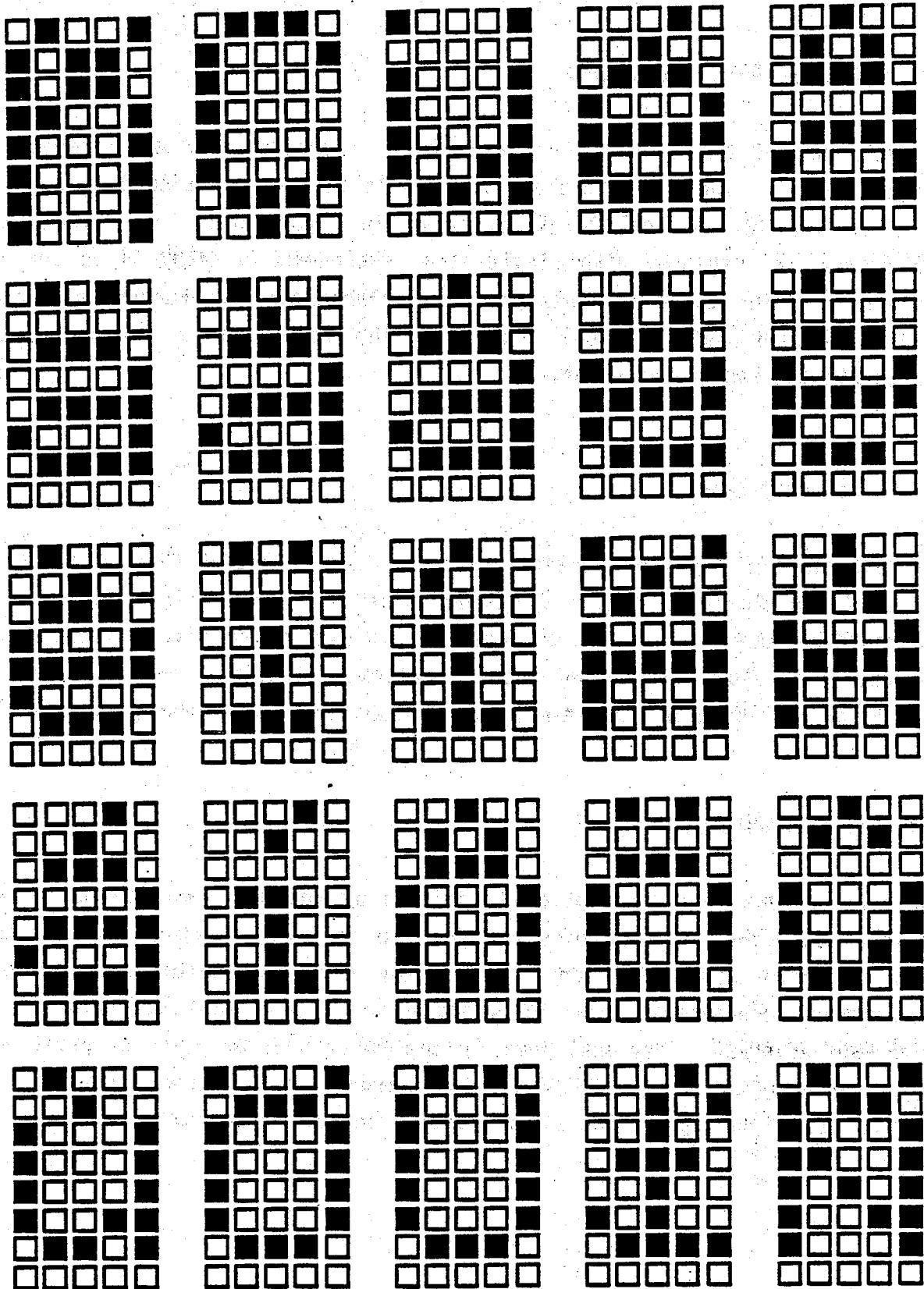


Figure 6-2. ASCII Character Format (Sheet 1 of 6)

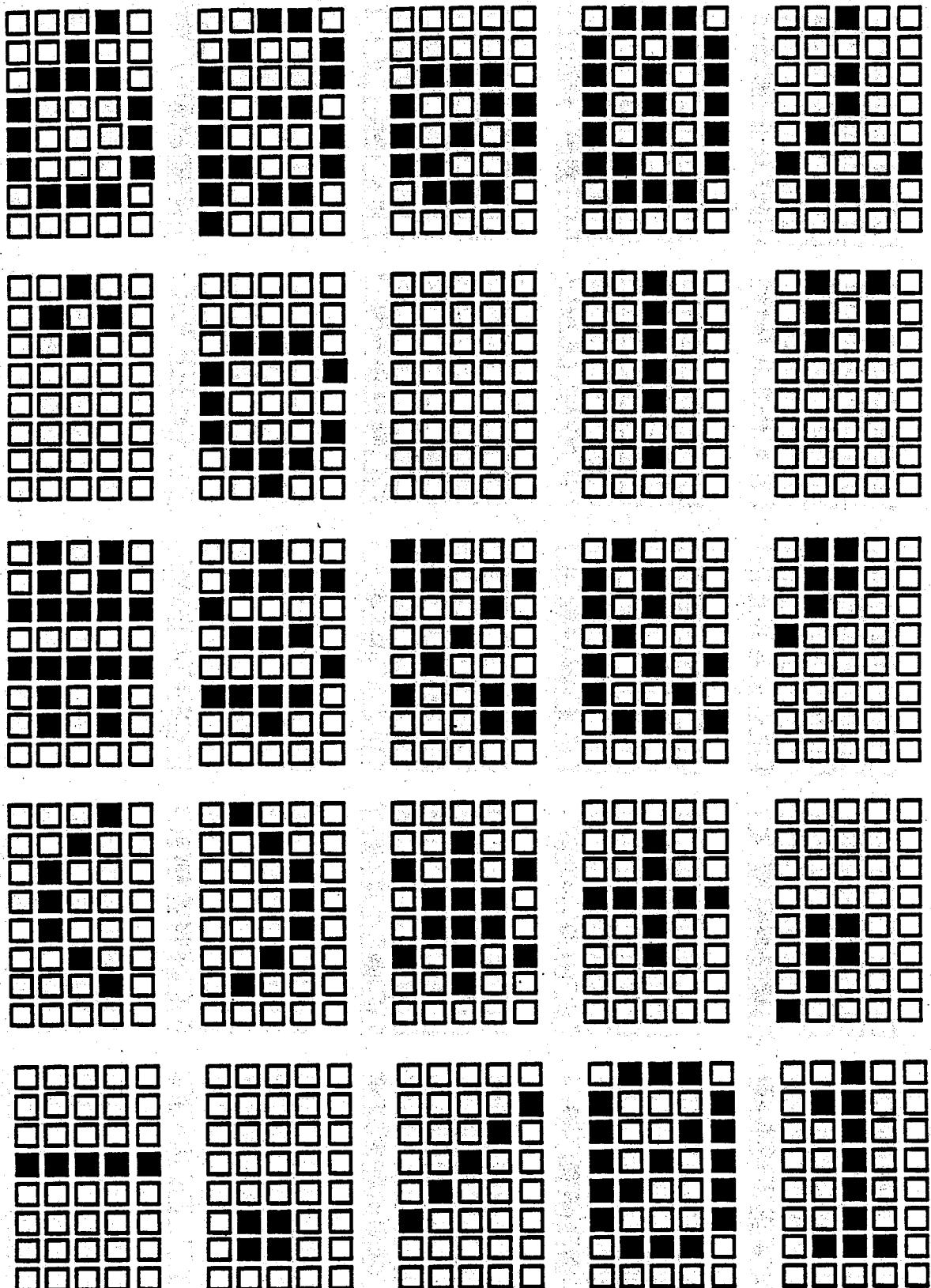


Figure 6-2. ASCII Character Format (Sheet 2 of 6)

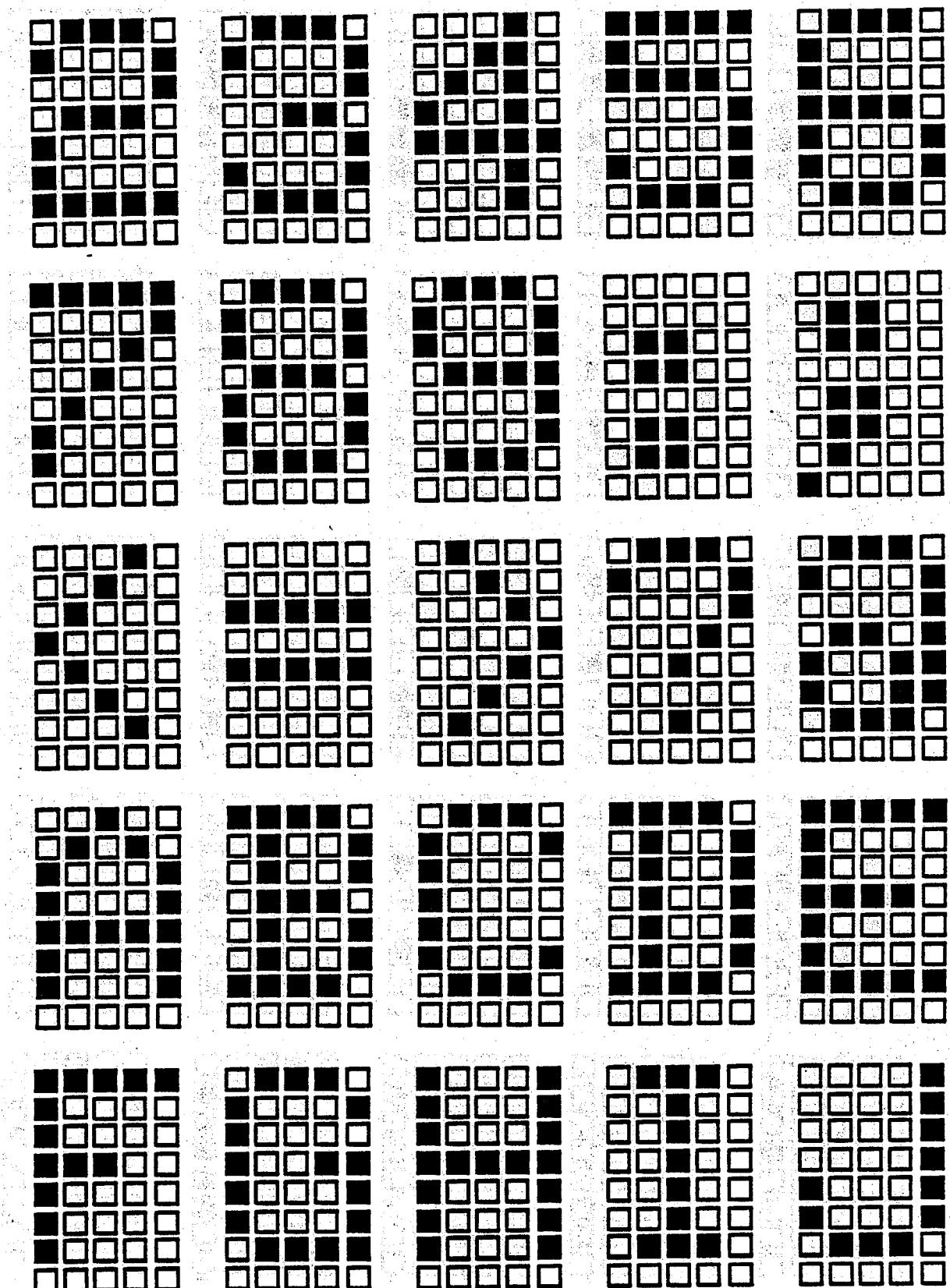


Figure 6-2. ASCII Character Format (Sheet 3 of 6)

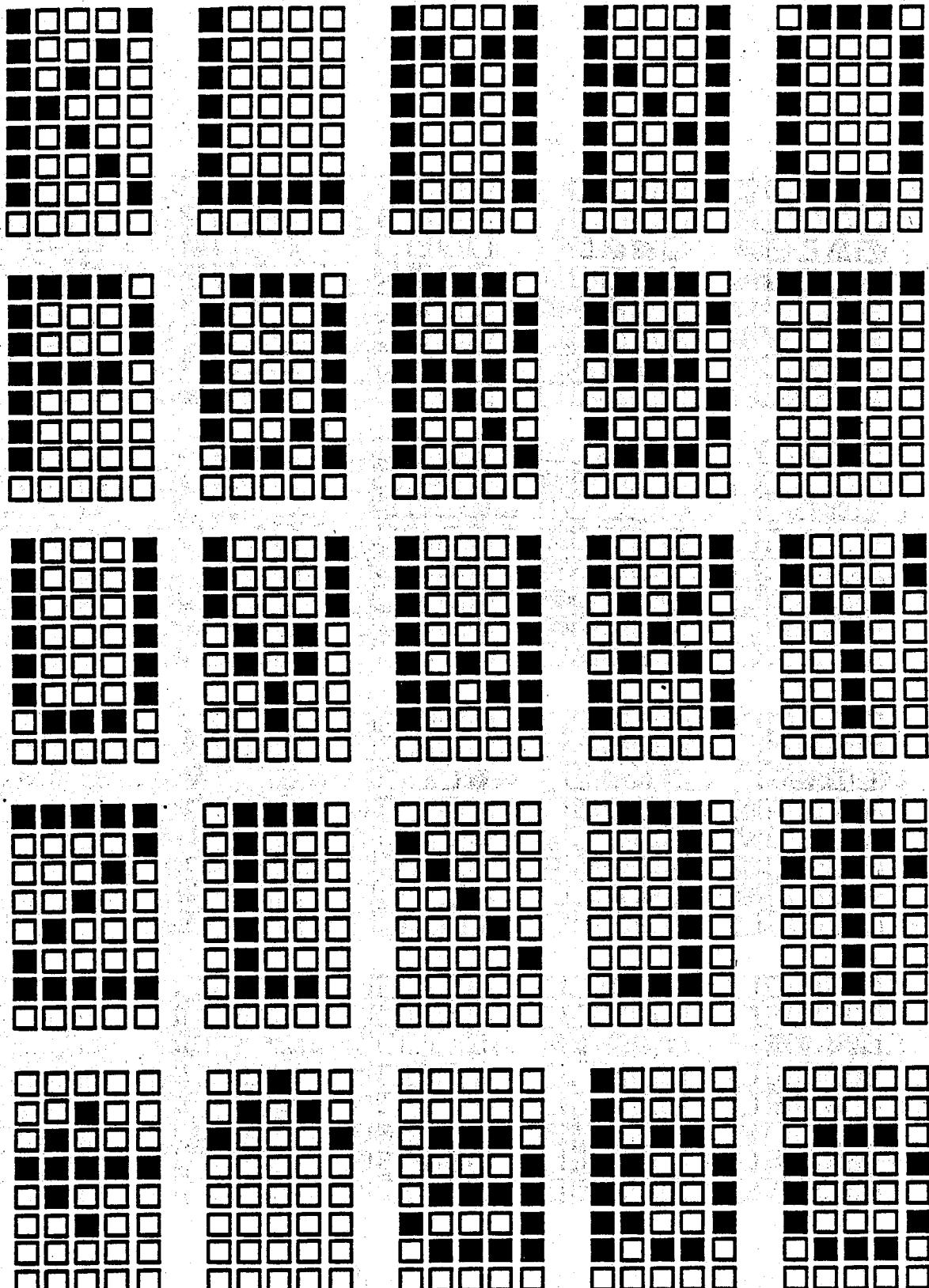


Figure 6-2. ASCII Character Format (Sheet 4 of 6)

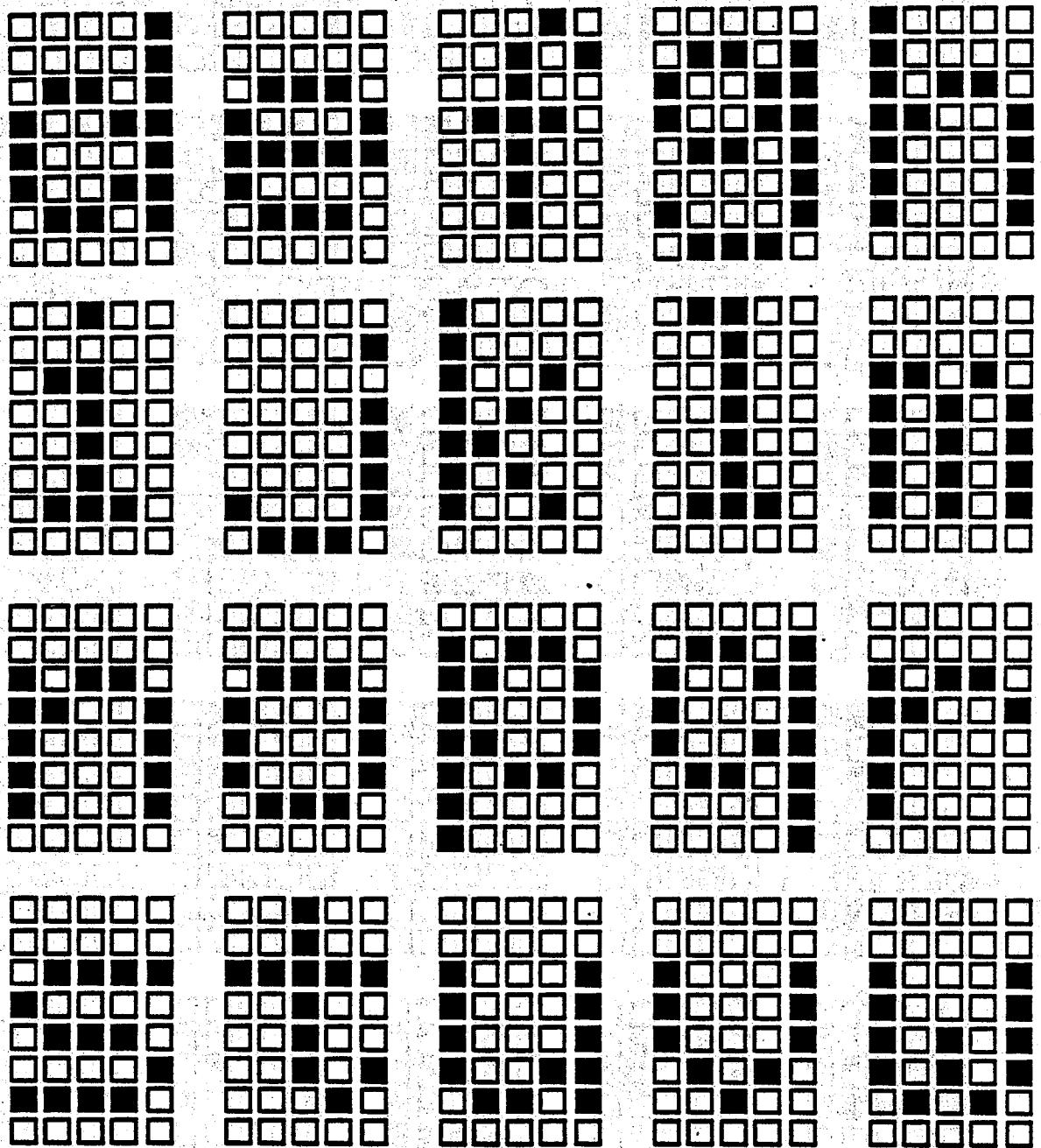


Figure 6-2. ASCII Character Format (Sheet 5 of 6)

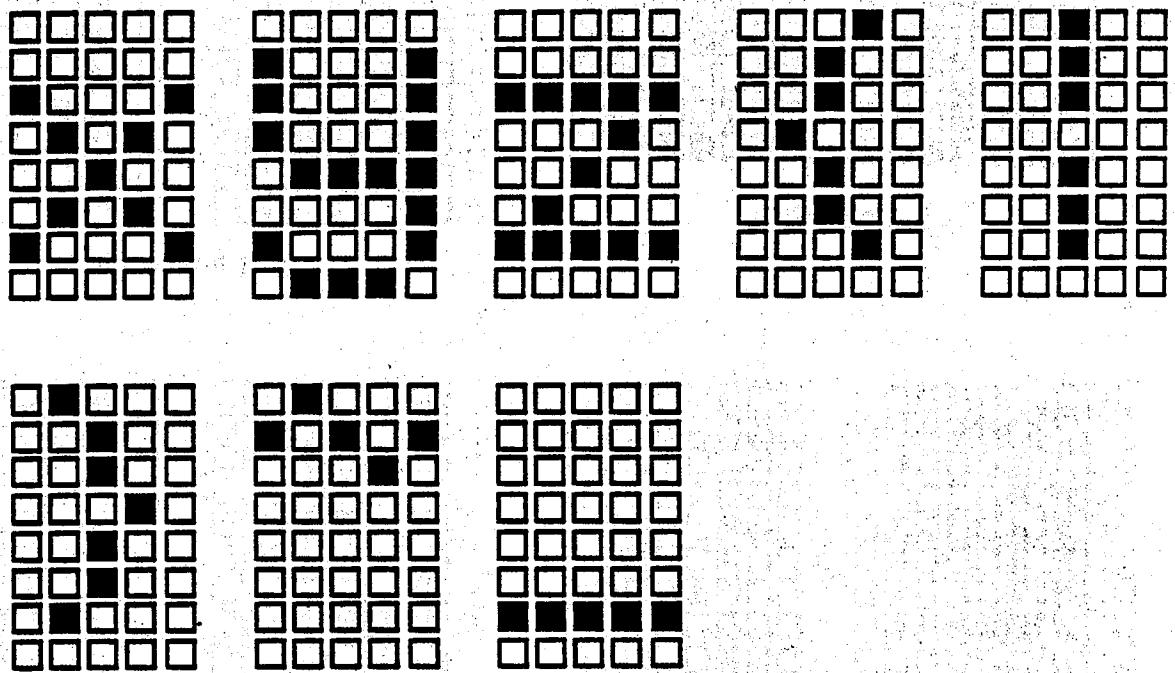
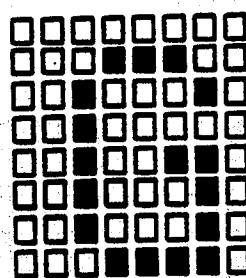
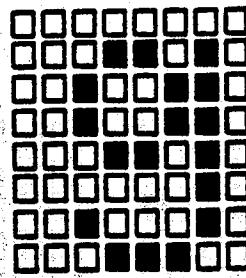


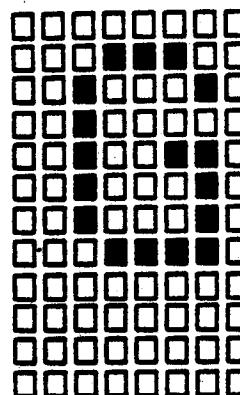
Figure 6-2. ASCII Character Format (Sheet 6 of 6)



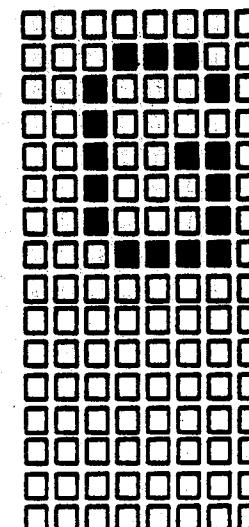
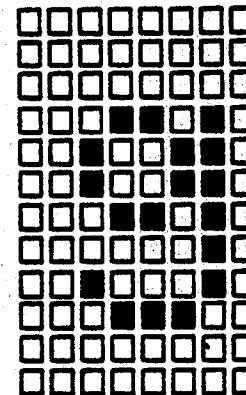
8 LINES/ROW



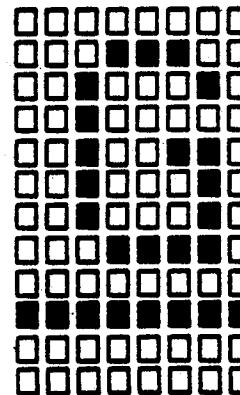
10 LINES/ROW



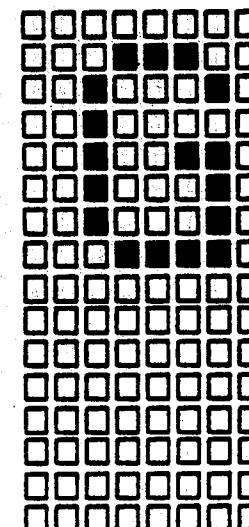
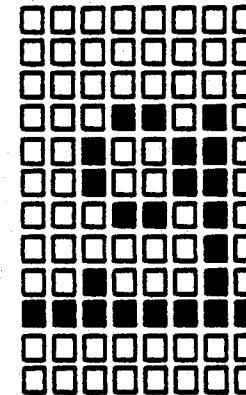
12 LINES/ROW



12 LINES/ROW



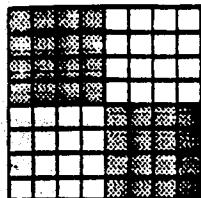
12 LINES/ROW WITH UNDERLINE



16 LINES/ROW

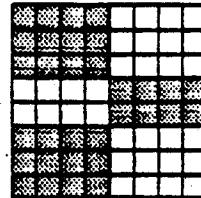
Figure 6-3. Character Position vs Lines per Character Row

BLOCK 0



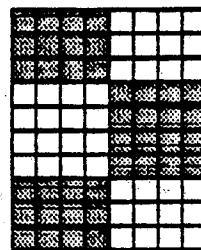
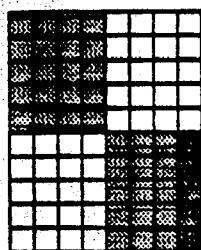
BLOCK 2

1 0
2 3
3 4

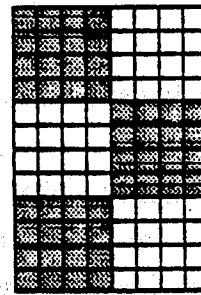
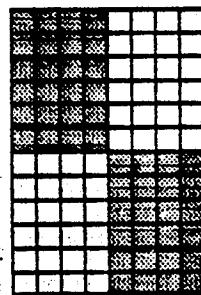


BLOCK 1
BLOCK 3
BLOCK 6

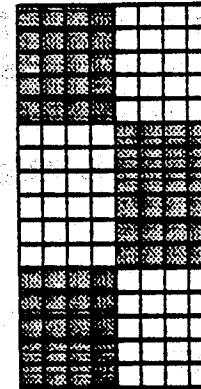
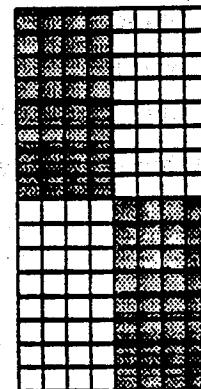
8 LINES PER CHARACTER ROW



10 LINES PER CHARACTER ROW



12 LINES PER CHARACTER ROW



16 LINES PER CHARACTER ROW

MOSAIC 4

MOSAIC 6

Figure 6-4. Mosaics 4 and 6

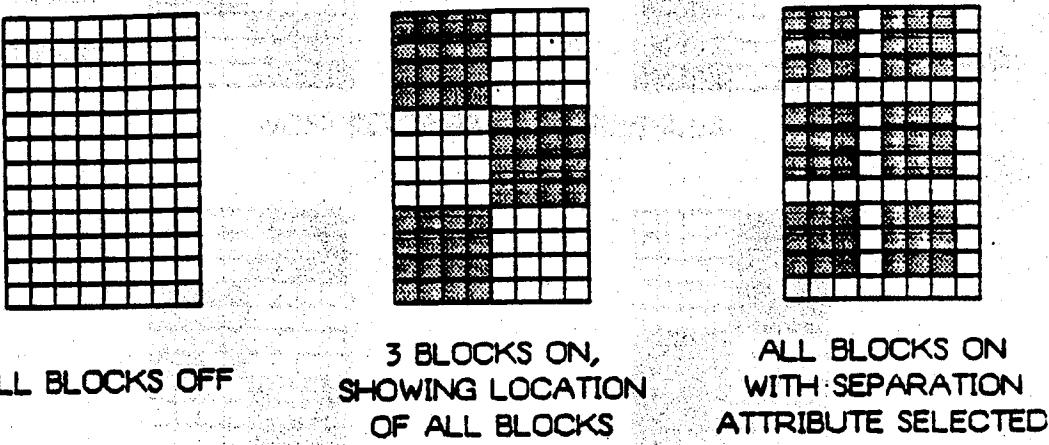


Figure 6-5. Mosaic Separation

6.3.4 Dynamically Redefinable Characters

Dynamically redefinable characters are available in either a character-oriented type or a graphics-oriented type. Their image tables must contain the number of bits per pel specified in the display data mode register. The difference between the two types of DRC's is that only the least-significant bit of each pel value is used for character type DRC's, while graphic type DRC's use all bits to select pel colors. Graphic-oriented DRC's are available in list modes 2, 3, 4, and 5. Alphanumeric type DRC's can be used in list modes 0, 1, and 4. Attributes available for the DRCs are flash, foreground and background colors, double height and width, CMR offset, invert, underline, and color/resolution.

6.3.5 Fixed Objects

Fixed objects are redefinable characters with additional true-object interactive attributes: collision enable, color collision, priority, and shading.

They may also have the DRC's appearance altering attributes as described in 6.3.4 Dynamically Redefinable Characters. The fixed object image tables must always contain the number of bits per pel in the data display mode register.

6.3.6 Attributes

The display screen list is comprised of one, two, or three bytes, depending on the list mode selected, for each character. Within these bytes are flags that allow the individual characters to be modified with appearance altering attributes. The attributes are applied to the data once it is in the RMS, so they do not alter any of the image table contents. There are no attributes that can be set for more than one character time.

6.3.7 Foreground and Background Color

Foreground and background colors are selectable in list modes 1 and 4 for ROM based alphanumerics and mosaics, and for the first 32 characters of the DRC's when they are used with alphanumeric attributes. When a DRC is used as an alphanumeric, the least-significant bit of each pel in the image table selects between foreground (LSB = 1) and background color (LSB = 0). The DRC image table must conform to the bits per pel selected in the data display mode register even though only the least-significant bit is used.

In list modes 0 and 2, the ROM based alphanumerics and mosaics have no other attributes. Only foreground and background colors.

6.3.8 Color Mapping Offset

DRC's, fixed objects, and alpha/mosaics in some list modes are allowed to have a color offset that can be inclusive ORed with the pel data from the image table to get a new CMR address. CMR offsets are allowed to be one, three, or four bits. Table 6-11 shows where they are used in the color mapping address space.

Table 6-11. CMR Offsets in Color Mapping

Address Space

Number of CMR Offset Bits	CMR Address					Comment
	A	A	A	A	A	
4	X	X	X	X	0	A0 = 0
3	0	X	X	X	0	A4, A0 = 0
1	X	0	0	0	0	A3-A0 = 0

X = Color Mapping Offset Bit

6.3.9 Flash

The flash attribute causes the foreground color to become the background color at periodic rates. The rates are programmable at 1, 2, or 4 Hertz. Alphanumerics, DRCs, mosaics, and fixed objects are allowed to flash.

6.3.10 Invert

Alphanumeric, mosaic, and alpha-type DRC's can have their foreground and background colors reversed. If the character has multiple bits per pel, all of the pel bits are inverted. A character that has both invert and CMR offset bits will have its pel value inverted, then the CMR offset will be applied.

6.3.11 Underline

Underline applies to alphanumeric and some redefinable characters in list modes 4 and 5. It occupies the tenth row from the top to the character (row 9). Invert and flash attributes work on underline in the same way as the character itself. No underline attribute is displayed if the character row size is eight scan lines.

6.3.12 Double High

Double-high characters are stretched to twice their normal height, while their width remains the same. They occupy all of two contiguous character rows. The same character code and attributes must be written into both of the normal size character locations that the double-high character occupies in the display list.

6.3.13 Double Wide

The double-wide attribute cause an eight pel wide character to be displayed with twice its normal width, but with normal height. The character is stretched to the right so that next character is covered by the extra width. The character code and attributes of the second character is ignored by the RMC.

6.3.14 Color/Resolution

Some redefinable characters in list modes 3 and 4 allow trading some resolution for an increased color range when one or two bits per pel resolution is selected. When this bit is set, the RMS uses the image table pel data from two pels to get one color for both pels. With one bit per pel selected, two colors are available in the normal mode, but with the COL/RES flag set, each two pels now can take on one of four colors. In the two bits per pel mode, four colors are expanded to sixteen colors. The following is an example of this usage:

EXAMPLE:

List mode 4 at two bits per pel, a DRC's first line of video data in the image table is:

First byte -- 1 0 1 0 1 1 0 1

Second byte -- 0 0 0 1 0 1 0 1

Colors displayed if COL/RES flag NOT selected:

CMR	2	2	3	1	0	1	1	1		
Pel:	1	0	1	0	1	0	0	1	0	1

Colors displayed if COL/RES flag IS selected:

CMR	A	D	1	5						
Pel:	1	0	1	0	1	1	0	1	0	1

In both cases, eight pixels are displayed, but with COL/RES flag set pixels are grouped into pairs to form the pels.

6.3.15 Separation

The separation attribute is only used on mosaic characters. It leaves a one pel background border on the bottom and right side of each mosaic block. This bit is active low.

6.3.16 Priority

Priority is used to determine which item is in front (visible) when two objects (fixed and true) occupy the same place on the displayed screen. The fixed objects in list modes 1, 3, and 4 can have any one of eight priorities. These priorities are interleaved with the eight true object priorities. When the priority of fixed and true objects are the same, the fixed object is visible. Priorities are assigned with the higher priority being closest to the front of the screen. Alphanumerics and DRCs and mosaics all have the lowest priority.

6.3.17 COLLISION ENABLE

Fixed objects have a one bit code that allow them to be used in collision reporting with true objects. The collisions are reported to the MPU by the true object. Collisions occur when a true object is moved to a fixed object's X/Y location on the screen and they both have collision enabled. See 6.4 TRUE OBJECTS for more details.

6.3.18 Color Collision

Color collision, a 2-bit code, is used with fixed objects to determine which part of the fixed object is allowed to cause a collision. The value of the fixed object's pel data, before CMR offset is combined, must be greater than the color collision value for a collision to be detected.

This same test is used to determine priority. If the pel data is greater than the color collision value, then that pel's priority is determined by the character's priority attribute. If it is less than or equal to the color collision code, then its priority is the same as alphanumerics.

6.3.19 Shading

Shading is a fixed object attribute that affects true objects with higher priority when they move in front of the fixed object. The color collision test described earlier is used for shading as well, if the shading attribute is set. If the fixed object's pel value is greater than the two color collision bits, the most-significant bit of the true object's pel value is inverted. If the fixed object's pel value is less than or equal to the two color collision bits, nothing is done to the true object's pel value. The effect of this attribute is to change the true object's color as it passes over the fixed object.

6.4 TRUE OBJECTS

True objects are RMS hardware intensive objects that are designed to be easily positioned on the displayed screen and interact with other fixed and true objects. Eight identical sets of registers allow simultaneous and independent operation of eight objects. True objects can be reused in real time, so more than eight objects can appear on the displayed screen. Each object may be individually enabled or disabled.

True objects are described by image tables similar to the list mode image tables and they are positioned on the screen using X/Y registers. Each true object has a collision reporting register used to inform the MPU that it has overlapped another true object or a fixed object. The true object hardware is independent of the display mode, so true objects can be used in any list mode and bit-plane mode.

6.4.1 Object Position

The position on the screen of the upper left-hand corner of the object is defined by X and Y coordinates. The X/Y grid is larger than the visible screen allowing an object to be positioned off the screen in any direction. The X and Y position registers are 10-bit registers which correspond to the horizontal and vertical counters of the displayed screen. These counters count at a pel clock rate. This allows object movement at a pel resolution rate in any horizontal or vertical operating mode. X and Y positioning of true objects is also independent of scrolling. True object collisions are reported even if they occur off the visible screen.

6.4.2 Names

True objects are given names which define their position in the object image table. Each name is eight bits in length, thus allowing a maximum of 256 different object patterns. An object is allocated 128 bytes of DRAM for pattern storage. The address for the first byte of object \$00 is stored in the true object start address register. All other object patterns follow in numerical order.

Objects are used by writing a name byte in one of eight hardware object registers, having them enabled and writing their position registers.

6.4.3 Pattern Data

Each true object can be described in one of two ways: bit-plane encoded or run-length encoded. Bit-plane encoding has 14 individually colored pels per video line, and run-length color can be one, three, five, or seven pels long. Run-length objects have flexibility in size, while bit-plane objects have more colors. Run-length or bit-plane object selection is done by using a bit in the object's X coordinate register.

The 128 bytes used to define an object are divided into four byte packets. Each packet describes one video line. Objects may be defined in less than 128 bytes, but this does not save memory since all of the bytes are allocated to the object. Less than 128 bytes can be used, thus freeing the object hardware to display another object as soon as the last visible line is displayed.

6.4.3.1 RUN LENGTH. In run-length encoding, each nibble describes a line segment. The first two bits define the length: 00 = one pel, 01 = three pels, 10 = five pels, and 11 = seven pels. The other two bits define the color for the segment. See Table 6-12.

Table 6-12. Run-Length Encoding

	B7	B6	B5	B4	B3	B2	B1	B0
1st Byte	LL	CMR4	CMR3	CMR2	1R1	1R0	1C1	1C0
2nd Byte	2R1	2R0	2C1	2C0	3R1	3R0	3C1	3C0
3rd Byte	4R1	4R0	4C1	4C0	5R1	5R0	5C1	5C0
4th Byte	6R1	6R0	6C1	6C0	7R1	7R0	7C1	7C0

LL = Last Line (Active Low)

6.4.3.2 BIT-PLANE ENCODING. In bit-plane encoding, the 3-1/2 bytes of video data are treated as 14 2-bit colors, each 2-bit color describing one pel. See Table 6-13.

Table 6-13. Bit-Plane Encoding

	B7	B6	B5	B4	B3	B2	B1	B0
1st Byte	LL	CMR4	CMR3	CMR2	1P1	1P0	2P1	2P0
2nd Byte	3P1	3P0	4P1	4P0	5P1	5P0	6P1	6P0
3rd Byte	7P1	7P0	8P1	8P0	9P1	9P0	10P1	10P0
4th Byte	11P1	11P0	12P1	12P0	13P1	13P0	14P1	14P0

LL = Last Line (Active Low)

6.4.4 Last-Line Flag

Each entry in the true object image table must contain a 4-byte packet with the last-line flag reset. Systems using interlaced data must have two such lines. The RMS is designed so that, once the scan line matching the true object's Y coordinate is reached, it starts fetching and displaying 4-byte packets until it finds a packet with its most-significant byte clear or until vertical retrace. If interlaced data is being used, two independent versions of that process are performed. Each time a last-line flag is encountered, the object's object available flag is set, so for interlaced data, it is set twice. If the last-line bit is set in all 32 of the 4-byte packets, the object is repeated, one version directly under the previous one, until vertical retrace.

6.4.5 Zoom Factors

Each true object can be individually zoomed horizontally and vertically or both without changing DRAM. Two bits in each object's X and Y position registers select the display size: 00 = normal size, 01 = 2x zoom, 10 = 4x zoom, and 11 = 8x zoom. The origin of the expansion is the object's X/Y register value, which is the upper left-hand corner of the object. As a result, horizontal expansion is to the right and vertical expansion is down.

6.4.6 Priority

When two objects occupy the same display space, priority is used to determine which object will be visible. The priority of the true objects are defined in hardware and cannot be changed. Object seven has the highest priority and object zero has the lowest priority. Fixed object priorities are interleaved with true object priorities. As a result, when fixed objects and true objects have the same priority, the true object is hidden by the fixed object.

Alphanumerics and bit plane have a lower priority than the lowest priority object, and the border has the highest priority. Note that two true objects, or a true object and a fixed object, can collide regardless of their priority level.

6.4.7 Collision

The RMS control register map contains eight bytes of registers for collision reporting. There is one register for each true object. The bits within a true object's register indicate which other true object it has collided with.

Before an object can have a reportable collision, its collision enable bit and the collision enable bit of the colliding object must be set. Transparent pixels are not considered when checking for collisions. For a true object to collide with a fixed object, the fixed object pel involved must pass the fixed object's color collision test.

SECTION 7

MEMORY ORGANIZATION

This section describes the dynamic memory that must be connected to the RMS. It lists the memory parts that can be used with the RMS and it also describes the different memory configurations that can be attached to the RMS, and their features.

7.1 OVERVIEW

The RMS interfaces directly to dynamic RAM (DRAM) which is configured in banks of eight bits. The DRAM is used for video memory and program storage. The MPU can access the screen memory at any time due to time division multiplexing. During a read, the data is held until the MPU expects to receive it. In the meantime, the RMS display process is accessing the memory to retrieve up to four bytes which will be used to generate video display. During each memory cycle, both the MPU and the RMS get one access to DRAM.

7.2 MEMORY REQUIREMENTS

The RMS supports different types of DRAM which must have an access time of 150 nanoseconds or less. These types are: 16K x 1, 64K x 1, 16K x 4, or 256K x 1.

The dynamic memory can be organized in one, two, or four banks. This selection is done through the memory organization register. These banks are always eight bits wide. For a 16-bit MPU a minimum of two banks are required with bank zero tied to the most-significant byte and bank one tied to the least-significant byte (see Figure 7-1). The following table summarizes the possible memory organizations.

Banks	Amount
1	16K to 256K
2	32K to 512K
4	64K to 1Meg

For a MC6809E interface with two banks of DRAM, the even address accesses are done to bank zero and the odd ones are done to bank one. If four banks are used, accesses to the lower half of memory are done to banks zero and one as described earlier, and accesses to the upper half are done to banks two and three.

7.3 SYSTEM MEMORY MAP

The RMS lets the user specify if he wants his memory map to be ROM or RAM intensive by using the MAPA bit. The memory map also changes from machine 1 to machine 2. The following paragraphs apply to machine 1, the machine 2 memory map will be discussed in SECTION 8 MACHINE 2.

The size and location of ROM, DRAM, and I/O within the 1 megabyte address space is controlled by the S bus. In machine 1, there are four system memory maps for each microprocessor.

MC68000

RMC

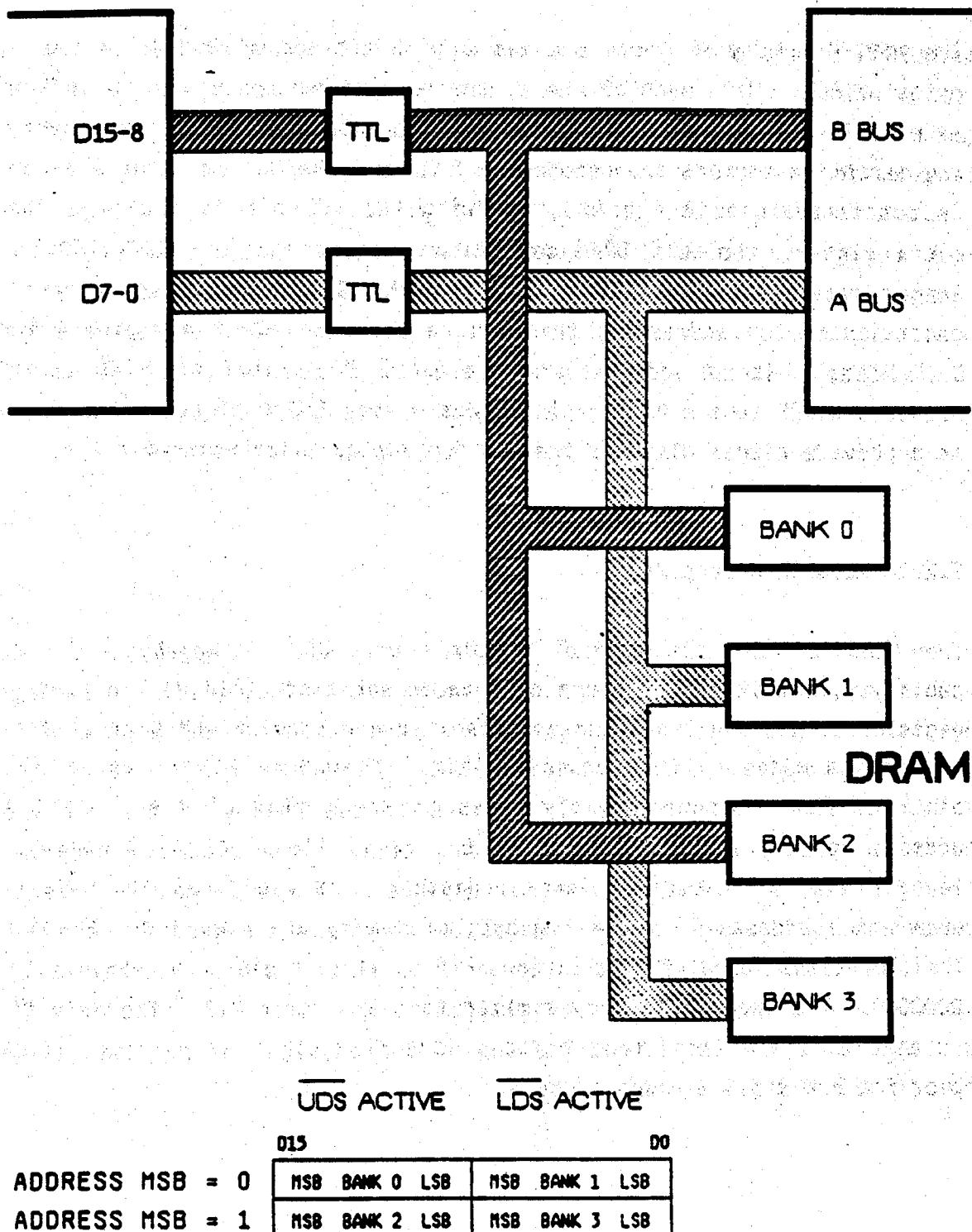


Figure 7-1. MC68000 DRAM Bank Organization

7.3.1 M68000 Family Map

The M68000 Family MPUs can use a map with 60K bytes of ROM or one with 252K bytes of ROM. With each of these, the exception vectors can be in ROM or DRAM (see Table 7-1). The MAPA bit selects the amount of ROM. After system reset, the exception vectors are decoded in ROM, and the bottom 1K byte block of DRAM is not accessible to the RMS. Setting the VEC bit in the page independent block register selects DRAM exception vectors and the ROM vectors are no longer accessible. When VEC is clear, the S bus generates a zero for the exception vector addresses; this can be used to select a separate ROM. When VEC is set, these addresses generate an S bus value of seven. The RMS provides DTACK to the MPU for all decodes except I/O blocks D and E. The user must provide either DTACK or the M6800 peripheral interface signals.

7.3.2 MC6809E Memory Map

The MC6809E can use machine 1 memory map with 1 megabyte of addressing capability, but it must use the page independent block (PIB) and paging control registers. By using those two registers it can control which part of the total 1 megabyte makes up the accessible 64K. The lower PIB is used to select a block of DRAM that permanently resides at the bottom of the MPU's 64K byte address space. The MPU determines the size of the block by programming the lower PIB bits in the PIB control register. The user does not have control over which addresses in the 1 megabyte of memory are used for the lower PIB. The locations used for the lower PIB always begin at physical address \$00000. The upper PIB is very similar to the lower PIB. The only difference is that they are taken from the top of the physical memory and reside in the top of the MPU's 64K address space.

Table 7-1. M68000 Family Memory Map

Chip Select	MAPA	VEC	Start	End	S Bus	Name	Size
0	0 -	0	00000 - 003FF	0	Exception Vectors	1K	
			00400 - BFFFF	7	DRAM	767K	
			C0000 - CFFFF	1	ROM 9	64K	
			D0000 - DFFFF	2	ROM A	64K	
			E0000 - EFFFF	3	ROM B	64K	
			F0000 - FFFFF	4	ROM C	60K	
			FF000 - FF7FF	5	I/O D	2K	
			FF800 - FFBFF	6	I/O E	1K	
			FFC00 - FFDFE	7	Reserved	256	
			FFE00 - FFE8F	7	RMS Registers	192	
			FFEC0 - FFFFF	7	Reserved	576	
0	0	1	00000 - BFFFF	7	DRAM	768K	
			C0000 - CFFFF	1	ROM 9	64K	
			D0000 - DFFFF	2	ROM A	64K	
			E0000 - EFFFF	3	ROM B	64K	
			F0000 - FFFFF	4	ROM C	60K	
			FF000 - FF7FF	5	I/O D	2K	
			FF800 - FFBFF	6	I/O E	1K	
			FFC00 - FFDFE	7	Reserved	256	
			FFE00 - FFE8F	7	RMS Registers	192	
			FFEC0 - FFFFF	7	Reserved	576	
0	1	0	00000 - 003FF	0	Exception Vectors	1K	
			00400 - EFFFF	7	DRAM	959K	
			F0000 - FEFFF	4	ROM C	60K	
			FF000 - FF7FF	5	I/O D	2K	
			FF800 - FFBFF	6	I/O E	1K	
			FFC00 - FFDFE	7	Reserved	256	
			FFE00 - FFE8F	7	RMS Registers	192	
			FFEC0 - FFFFF	7	Reserved	576	
0	1	1	00000 - EFFFF	7	DRAM	960K	
			F0000 - FEFFF	4	ROM C	60K	
			FF000 - FF7FF	5	I/O D	2K	
			FF800 - FFBFF	6	I/O E	1K	
			FFC00 - FFDFE	7	Reserved	256	
			FFE00 - FFE8F	7	RMS Registers	192	
			FFEC0 - FFFFF	7	Reserved	576	
1	X	X	RMS Not Selected				

Following reset, the user should make certain that at least the upper PIB is set up before changing the paging register.

NOTE

If the paging register is changed before the upper PIB is set, the control registers may disappear from the memory map and it will be impossible to get them back.

In machine 1 there are four memory maps available to the MC6809E. Refer to Table 7-2.

Table 7-2. MC6809 Memory Map

MAPA	UNFOLD	Start	End	S Bus	Name	Size
0	0	00000 - F7FFF	7	DRAM	992K	
		F8000 - F9FFF	1	ROM 0	8K	
		FA000 - FBFFF	2	ROM 1	8K	
		FC000 - FFEFF	3	ROM 2	16K - 256	
		FFF00 - FFFF1F	4	I/O 0	32	
		FFF20 - FFFF3F	5	I/O 1	32	
		FFF40 - FFFF5F	6	I/O 2	32	
		FFF60 - FFFF7F	7	Reserved	32	
		FFF80 - FFFF8F	7	RMS Registers	64	
		FFF90 - FFFFDF	7	Reserved	32	
		FFFEO - FFFFF	2	ROM 1 (Vectors)	32	
0	1	00000 - F7FFF	7	DRAM	992K	
		F8000 - F9FFF	1	ROM 0	8K	
		FA000 - FBFFF	2	ROM 1	8K	
		FC000 - FFDFFF	3	ROM 2	16K - 512	
		FFE00 - FFEFBF	7	RMS Registers	192	
		FFEC0 - FFEFFF	7	Reserved	64	
		FFF00 - FFFF1F	4	I/O 0	32	
		FFF20 - FFFF3F	5	I/O 1	32	
		FFF40 - FFFF5F	6	I/O 2	32	
		FFF60 - FFFFDF	7	Reserved	128	
		FFFEO - FFFFF	2	ROM 1 (Vectors)	32	
1	0	00000 - FFEFFF	7	DRAM	1M - 256	
		FFF00 - FFFF1F	4	I/O 0	32	
		FFF20 - FFFF3F	5	I/O 1	32	
		FFF40 - FFFF5F	6	I/O 2	32	
		FFF60 - FFFF7F	7	Reserved	32	
		FFF80 - FFFFBF	7	RMS Registers	64	
		FFF90 - FFFFDF	7	Reserved	32	
		FFFEO - FFFFF	2	ROM 1 (Vectors)	32	
1	1	00000 - FFDFFF	7	DRAM	1M - 512	
		FFE00 - FFEFBF	7	RMS Registers	192	
		FFEC0 - FFEFFF	7	Reserved	64	
		FFF00 - FFFF1F	4	I/O 0	32	
		FFF20 - FFFF3F	5	I/O 1	32	
		FFF40 - FFFF5F	6	I/O 2	32	
		FFF60 - FFFF7F	7	Reserved	128	
		FFFEO - FFFFF	2	ROM 1 (Vectors)	32	

===== MOTOROLA

SECTION 8

MACHINE 2

Machine 2 allows software compatibility with the MC6883/MC6847 combination when the RMS is used with an MC6809E MPU. The RMS needs initialization software different than the MC6883/MC6847 to run existing software using the RMS. Once the RMS registers are initialized, MC6883-type control bits and a PIA-simulating register are used to control the RMS.

Machine 2 mode is invoked by setting the M2 bit in the memory map register. The MC6883 control bits work differently than the normal RMS controls. Each control bit occupies two bytes in memory. Writing to the odd byte sets the control bit, and writing to the even byte clears the bit. Reading those locations has no effect. Table 8-1 shows the MC6883 control bits followed by a discussion of the bits.

Address location \$FF22 simulates a PIA that controls the MC6847 register. The graphic mode bits GM1 and GMO provide a superset of the MC6883-MC6847 display modes. The CSS bit is the same as the color set select bit on the MC6847.

B7	B6	B5	B4	B3	B2	B1	B0	
G/A	--	GM1	GMO	CSS	--	--	--	\$FFF22

Table 8-1. MC6883 Control Bits

Bit	Set Address	Reset Address
TY	\$FFDF	\$FFDE
MJ	NOT USED BY RMS	
MO	NOT USED BY RMS	
RO	NOT USED BY RMS	
R1	NOT USED BY RMS	
P1	\$FFD5	\$FFD4
F6	\$FFD3	\$FFD2
F5	\$FFD1	\$FFD0
F4	\$FFCF	\$FFCE
F3	\$FFCD	\$FFCC
F2	\$FFCB	\$FFCA
F1	\$FFC9	\$FFC8
F0	\$FFC7	\$FFC6
V2	\$FFC5	\$FFC4
V1	\$FFC3	\$FFC2
VO	\$FFC1	\$FFC0

TY: This bit selects between 2 system memory maps which are decoded by the RMS and are available on the S bus.

M1, MO: These bits which were originally used to define the type of DRAM are not needed anymore because the user initializes the memory type in the memory organization control register.

R1, RO: These bits have no effect because the MPU cannot be put in the high speed mode.

P1: It is a paging bit for use with memory map 0. It is used in place of A15 for addresses from \$0000 through \$7FFF. This allows the ROM/RAM map to contain two 32K pages of RAM in the lower half of the address range.

F6-F0: These bits allow the display to be shifted from its base location at \$0000. The address of the top left corner of the display is:

$$\text{DISPLAY START} = \$0000 + (512 * F)$$

where F is F6 (MSB) through F0 (LSB)

V1, VO: These bits are part of the bits used to determine the display mode of the RMS as described in SECTION 6 DISPLAY MODES.

8.1 MACHINE 2 INITIALIZATION

At power up or reset, the RMS goes into the unfolded map, machine 1 state. The RMS registers should be initialized as described in the following paragraphs.

8.1.1 Registers Page 00

For page 00, the registers should be initialized as follows:

<u>Register</u>	<u>Value</u>	<u>Comments</u>
Memory Map	\$10	Machine 2 Folded Map Page 0
Display Data Mode	\$08	
Interrupt Status	\$00	
Border Color	\$10	CMR10, Video Disabled
Object Available	N/A	Read Only
Paging	\$0F	Page F
Page Independent Block	\$00	
Vertical Scroll	\$00	
Horizontal Scroll	\$00	
DRC Start Address	\$0000	
True Object Start Address	\$0000	
Fixed Object Start Address	\$0000	
Collision Status	N/A	Read Only
Collision Enable	\$00	
Real Time Output	\$00000000	
Real Time Input	\$00000000	
Memory Organization	See Register \$FFE24	
Video Operation	\$12	256 x 192
Sync Mode	\$04	Composite Sync Output
Virtual Screen Start	\$00000000	

<u>Register</u>	<u>Value</u>	<u>Comments</u>
Vertical Offset	\$00000000	
Horizontal Offset	\$00000000	
Virtual Screen Size	\$00001800	6K Bytes
Virtual Screen Width	\$00000020	32 Bytes

8.1.2 Registers Page 01

Write a \$50 to the memory map register to get to page 01. For page 01, the registers should be initialized as follows:

<u>Register</u>	<u>Value</u>	<u>Comments</u>
CMR00	\$40F0	Green - Stay in Page 01
CMR01	\$0FF0	Yellow
CMR02	\$000F	Blue
CMR03	\$0F00	Red
CMR04	\$0FFF	Buff
CMR05	\$0OFF	Cyan
CMR06	\$0FOF	Magenta
CMR07	\$0F80	Orange
CMR08	\$0000	Black
CMR09-0F	\$0000	
CMR10	\$00F0	Green - Alpha/Semigraph Border Color
CMR11	\$0F80	Orange - Alpha/Semigraph Border Color
CMR12	\$00F0	Green - Graphics Border Color
CMR13	\$0FFF	Buff - Graphics Border Color
CMR14-1F	\$0000	

8.1.3 Registers Page 10

Write a \$80 in CMR00 to get to page 10. For page 10, the registers should be initialized as follows:

<u>Register</u>	<u>Value</u>	<u>Comments</u>
True Object Position, 0-7	\$00000000	Disables All True Objects
True Object Names, 0-7	\$00	

When all initialization is done, write \$28 to the border color register to enable video. Machine 2 registers must be initialized.

8.2 MACHINE 2 DISPLAY MODES

The RMS offers a variety of display modes, as shown in Table 8-2. The color set select bit can be used with every display mode to choose between two sets of colors.

Table 8-2. Display Mode Selection

V2	V1	V0	GM1	GMO	G/A	Display Mode
0	0	0	0	0	0	Alpha-1/Semigraphics
0	0	0	0	1	0	Alpha-2/Semigraphics
0	0	0	1	0	0	Alpha-3/Semigraphics
0	0	0	1	1	0	Alpha-4/Semigraphics
0	0	1	x	0	1	Color Graphics 1
0	0	1	x	1	1	Resolution Graphics 1
0	1	0	x	0	1	Color Graphics 2
0	1	1	x	1	1	Resolution Graphics 2
1	0	0	x	0	1	Color Graphics 3
1	0	1	x	1	1	Resolution Graphics 3
1	1	0	x	0	1	Color Graphics 6
1	1	0	x	1	1	Resolution Graphics 6

8.2.1 Alpha/Semographics

The four types of alpha/semographics are supersets of the MC6847's alphanumeric internal and semographics four display modes. Each type can display 16 character rows with 32 characters per row.

8.2.2 Semographics

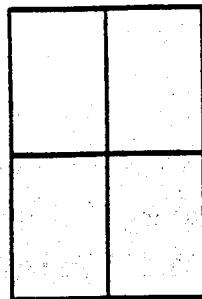
Semographics are selected by clearing the V bits and setting bit seven of the display data register. The C bits select the color (Table 8-3) and the L bits define 1 of 16 character blocks (Table 8-4). If the L code is 0000, then the color is black regardless of what the C bits are. Figure 8-1 shows the 16 patterns. The data is interpreted as follows:

Table 8-3. Semigraphic Colors

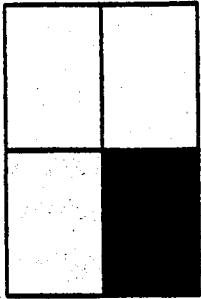
Lx	C2	C1	C0	Color	CMR Address
0	x	x	x	Black	08
1	0	0	0	Green	00
1	0	0	1	Yellow	01
1	0	1	0	Blue	02
1	0	1	1	Red	03
1	1	0	0	Buff	04
1	1	0	1	Cyan	05
1	1	1	0	Magenta	06
1	1	1	1	Orange	07

Table 8-4. Semigraphic Block

B7	B6	B5	B4	B3	B2	B1	B0
1	C2	C1	C0	L3	L2	L1	L0



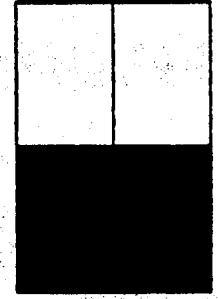
0



1

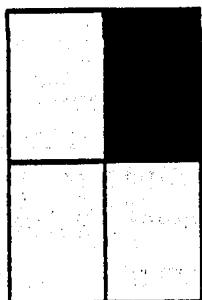


2



3

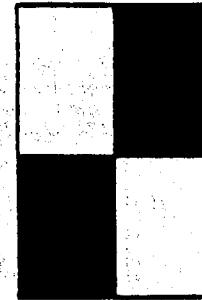
HEX VALUE OF BLOCK'S LEAST SIGNIFICANT DIGIT



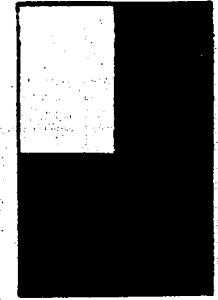
4



5

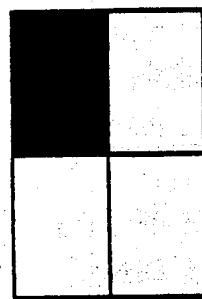


6

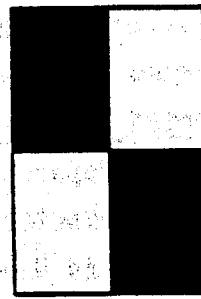


7

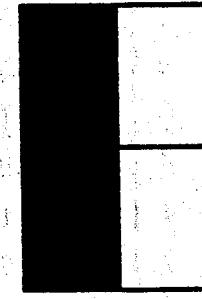
HEX VALUE OF BLOCK'S LEAST SIGNIFICANT DIGIT



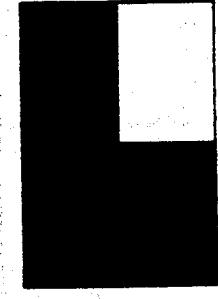
8



9

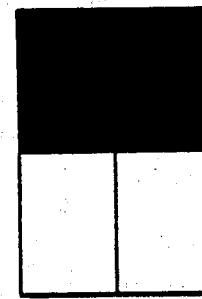


A

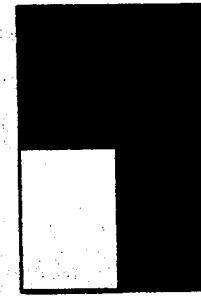


B

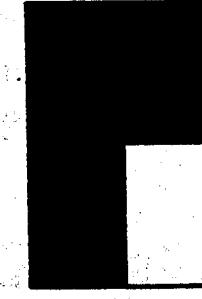
HEX VALUE OF BLOCK'S LEAST SIGNIFICANT DIGIT



C



D



E



F

HEX VALUE OF BLOCK'S LEAST SIGNIFICANT DIGIT

SELECTED COLOR SHOWN DARK

BACKGROUND COLOR SHOWN WHITE

Figure 8-1. Semigraphic Block Patterns

===== MOTOROLA

8.2.3 Alphanumerics

The four alphanumeric sets allow all uppercase, upper and lower case, and inverted versions of these. They are selected by GM1 and GMO as shown in Table 8-5. The color set select bit can be used to select the foreground and background colors. The machine 2 alphanumeric format is given in Figure 8-2.

Table 8-5. Alphanumeric Character Selection

Alpha	GM1	GMO	B7	B6	B5	Characters
One	0		0	0	0	Non-Inverted Uppercase Alphabet
			0	0	1	Non-Inverted Punctuation and Numbers
			0	1	0	Inverted Uppercase Alphabet
			0	1	1	Inverted Punctuation and Numbers
Two	0	1	0	0	0	Inverted Lowercase Alphabet
			0	0	1	Non-Inverted Punctuation and Numbers
			0	1	0	Inverted Uppercase Alphabet
			0	1	1	Inverted Punctuation and Numbers
Three	1	0	0	0	0	Inverted Uppercase Alphabet
			0	0	1	Inverted Punctuation and Numbers
			0	1	0	Non-Inverted Uppercase Alphabet
			0	1	1	Non-Inverted Punctuation and Numbers
Four	1	1	0	0	0	Non-Inverted Lowercase Alphabet
			0	0	1	Inverted Punctuation and Numbers
			0	1	0	Non-Inverted Uppercase Alphabet
			0	1	1	Non-Inverted Punctuation and Numbers

Table 8-6. Alphanumeric Colors

CSS	GM1	Foreground Color	Background Color
0	0	Black (CMR08)	Green (CMR00)
0	1	Green (CMR00)	Black (CMR08)
1	0	Black (CMR08)	Orange (CMR07)
1	1	Orange (CMR07)	Black (CMR08)

Alph Num D7-D0		DD-D4 In HEX															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
4	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0

0A BCDEF GHIJKLMNOPQRSTUVWXYZ[]^<=^? !%"#%&,()#+,-./0123456789:;,<=> ? ^ abcd e fghi jklmno pqr s tuvwxyz {}~` -

0A BCDEF GHIJKLMNOPQRSTUVWXYZ[]^<=^? !%"#%&,()#+,-./0123456789:;,<=> ? ^ abcd e fghi jklmno pqr s tuvwxyz {}~` -

0A BCDEF GHIJKLMNOPQRSTUVWXYZ[]^<=^? !%"#%&,()#+,-./0123456789:;,<=> ? ^ abcd e fghi jklmno pqr s tuvwxyz {}~` -

0A BCDEF GHIJKLMNOPQRSTUVWXYZ[]^<=^? !%"#%&,()#+,-./0123456789:;,<=> ? ^ abcd e fghi jklmno pqr s tuvwxyz {}~` -

NOTE : BLACK REPRESENTS THE FOREGROUND COLOR
WHITE REPRESENTS THE BACKGROUND COLOR
• D7 IS INTERNALLY CONNECTED TO A15. IF D7=1 THEN MOSAIC 4 CHARACTERS ARE SELECTED

Figure 8-2. Machine 2 Alphanumeric Format

The GMO bit selects upper case only when clear or upper case and lower case when set. The fonts are identical to machine 1.

The alphanumeric data is used to define the character type (B7, B6, B5) and the code for 1 of the 32 possible characters on a row in Figure 6-2 (B4, B3, B2, B1, B0).

In alpha one and three, the characters can be inverted on a character-by-character or a screen basis, but lower-case characters are not available. In alpha two and four, the characters can be inverted only on a screen basis, but lower-case characters are available.

8.3 GRAPHICS MODE

The eight graphic modes are similar to machine 1's bit-plane mode. They use from 1024 to 6144 bytes to hold the data. There are four resolution modes and four color modes. Refer to Tables 8-7 and 8-8.

Table 8-7. Color-Mode Colors

CSS	C1	C0	Color	Border
0	0	0	Green (CMR00)	Green (CMR12)
	0	1	Yellow (CMR01)	
	1	0	Blue (CMR02)	
	1	1	Red (CMR03)	
1	0	0	Buff (CMR04)	Buff (MCR13)
	0	1	Cyan (CMR05)	
	1	0	Magenta (CMR06)	
	1	1	Orange (CMR07)	

Table 8-8. Resolution-Mode Colors

CSS	Lx	Color	Border
0	0	Black (CMR08)	Green (CMR12)
	1	Green (CMR00)	
1	0	Black (CMR08)	Buff (CMR13)
	1	Buff (CMR04)	

The data byte for these modes is interpreted as follows:

	B7	B6	B5	B4	B3	B2	B1	B0
Color mode data	3C1	3C0	2C1	2C0	1C1	1C0	0C1	0C0
Resolution mode data	L7	L6	L5	L4	L3	L2	L1	L0

Table 8-9 lists the differences between these modes. C is color graphics and R is resolution graphics.

Table 8-9. Display Memory Requirements

	C1	R1	C2	R2	C3	R3	C6	R6
Picture Element Height	3	3	3	2	2	1	1	1
Picture Element Width	4	2	2	2	2	2	2	1
Elements per Row	64	128	128	128	128	128	128	256
Number of Rows	64	64	64	96	96	192	192	192
Memory (Bytes)	1024	1024	2048	1536	3072	3072	6144	6144

SECTION 9

ELECTRICAL SPECIFICATIONS

This section contains the electrical specifications and associated timing information for the RMC.

9.1 MAXIMUM RATINGS

Rating	Symbol	Value	Unit
Supply Voltage	V _{CC}	-0.3 to 7.0	V
All Input Voltages	V _{in}	GND -0.5 to V _{DD} +0.5	V
Current Drain per Pin Excluding V _{DD} and V _{SS}	I _{in}	10	mA
Operating Temperature	T _A	0 to 70	°C
Storage Temperature	T _{stg}	-55 to +150	°C

9.2 THERMAL CHARACTERISTICS

Characteristic	Symbol	Value	Symbol	Value	Unit
Thermal Resistance Plastic Ceramic	θ _{JA}	40	θ _{JC}	25*	°C/W

*Estimated

This device contains circuitry to protect the inputs against damage due to high static voltages or electric fields; however, it is advised that normal precautions be taken to avoid application of any voltage higher than maximum rated voltages to this high-impedance circuit. For proper operation it is recommended that V_{in} and V_{out} be constrained to the range GND ≤ (V_{in} or V_{out}) ≤ V_{CC}. Reliability of operation is enhanced if unused inputs are connected to appropriate logic voltage level (e.g., either GND or V_{CC}).

9.3 POWER CONSIDERATIONS

The average chip-junction temperature, T_J , in °C can be obtained from:

$$T_J = T_A + (P_D \cdot \theta_{JA}) \quad (1)$$

Where:

T_A = Ambient Temperature, °C

θ_{JA} = Package Thermal Resistance, Junction-to-Ambient, °C/W

$P_D = P_{INT} + P_{I/O}$

$P_{INT} = I_{CC} \times V_{CC}$, Watts – Chip Internal Power

$P_{I/O}$ = Power Dissipation on Input and Output Pins – User Determined

For most applications $P_{I/O} < P_{INT}$ and can be neglected.

An approximate relationship between P_D and T_J (if $P_{I/O}$ is neglected) is:

$$P_D = K \cdot (T_J + 273^\circ\text{C}) \quad (2)$$

Solving equations 1 and 2 for K gives:

$$K = T_D \cdot (T_A + 273^\circ\text{C}) + \theta_{JA} \cdot P_D^2 \quad (3)$$

Where K is a constant pertaining to the particular part. K can be determined from equation 3 by measuring P_D (at equilibrium) for a known T_A . Using this value of K the values of P_D and T_J can be obtained by solving equations 1 and 2 iteratively for any value of T_A .

The total thermal resistance of a package (θ_{JA}) can be separated into two components, θ_{JC} and θ_{CA} , representing the barrier to heat flow from the semiconductor junction to the package (case) surface (θ_{JC}) and from the case to the outside ambient (θ_{CA}). These terms are related by the equation:

$$\theta_{JA} = \theta_{JC} + \theta_{CA} \quad (4)$$

θ_{JC} is device related and cannot be influenced by the user. However, θ_{CA} is user dependent and can be minimized by such thermal management techniques as heat sinks, ambient air cooling and thermal convection. Thus good thermal management on the part of the user can significantly reduce θ_{CA} so that $\theta_{JA} = \theta_{JC}$. Substitution of θ_{JC} for θ_{JA} in equation 1 will result in a lower semiconductor junction temperature.

Values for thermal resistance presented in this data sheet, unless estimated, were derived using the procedure described in Motorola Reliability Report 7843, "Thermal Resistance Measurement Method for MC68XX Microcomponent Devices", and are provided for design purposes only. Thermal measurements are complex and dependent on procedure and setup. User derived values for thermal resistance may differ.

9.4 DC ELECTRICAL CHARACTERISTICS (GND = 0, TA = 0° C to 70°C)

Parameter	Symbol	Min	Typ	Max	Unit
Supply Voltage	V _{CC}	4.75	5	5.25	V
Input Leakage Current V _{CC} = Max, V _{in} = Max	I _{in}	-	-	2.5	uA
Input High Voltage (All Inputs Except X0-X9) V _{CC} = Min	V _{IH}	2.4	-	V _{CC}	V
Input Low Voltage V _{CC} = Min	V _{IL}	GND-0.3	-	0.8	V
Output High Voltage (All Outputs) I _{OH} = 400 uA	V _{OH}	2.4	-	-	V
Output Low Voltage I _{OL} = 5.3 mA I _{OL} = 1.6 mA	V _{OL}	-	-	0.5	V
R, G, B: Output High Voltage (R _{Load} = 100K ohm) Output Low Voltage (C _{Load} = 20 pF) Blanking Voltage Sync Voltage (G Output Only)	V _{OH} V _{OL} V _{blank} V _{sync}	- 1.0 -	2.5 1.5 1.5 1.07	3.0	V
R, G, B dc Offset Voltage R _{Load} = 100K ohm. C _{Load} = 20 pF	V _{RGB}	-	-	0.01	V
Differential - Non-Linearity, % of Full Scale Between any Output	-	-	TBD	-	%
High-Impedance Leakage Current V _{CC} = Max, V _{in} = 0.5 V to 2.4 V	I _{TSI}	-20	-	20	uA
Drain Current, V _{CC} = 5.0 V	I _{CC}	-	TBD	-	mA
Power Dissipation, V _{CC} = 5.0 V	P _D	-	TBD	-	W

9.5 DRAM TIMING (See Figures 9-1 through 9-4)

No.	Parameter	Symbol	Min	Max	Unit
1	Random Read or Write Cycle Time	t_{RC}	335	-	ns
2	Access Time from Row Address Strobe	t_{RAC}	168	-	ns
3	Access Time from Column Address Strobe	t_{CAC}	84	-	ns
4	Output Buffer and Turn-Off Delay Time	t_{OFF}	t_{CAC}	-	ns
5	Row Address Strobe Precharge Time	t_{RP}	140	-	ns
6	Row Address Strobe Pulse Width	t_{RAS}	196	-	ns
7	Column Address Strobe Pulse Width	t_{CAS}	112	-	ns
8	Row to Column Strobe Lead Time	t_{RCD}	84	-	ns
9	Row Address Setup Time	t_{ASR}	0	-	ns
10	Row Address Hold Time	t_{RAH}	28	-	ns
11	Column Address Setup Time	t_{ASC}	56	-	ns
12	Column Address Hold Time	t_{CAH}	56	-	ns
13	Column Address Hold Time Referenced to RAS	t_{AR}	140	-	ns
14	Transition Time (Rise and Fall)	t_r, t_f	3	-	ns
15	Read Command Setup Time	t_{RCS}	196	-	ns
16	Read Command Hold Time	t_{RCH}	168	-	ns
17	Read Command Hold Time Referenced to RAS	t_{RRH}	168	-	ns
18	Write Command Hold Time	t_{WCH}	140	-	ns
19	Write Command Hold Time Referenced to RAS	t_{WCR}	223	-	ns
20	Write Command Pulse Width	t_{WP}	196	-	ns
21	Write Command to Row Strobe Lead Time	t_{RWL}	168	-	ns
22	Write Command to Column Strobe Lead Time	t_{CWL}	168	-	ns
23	Data In Setup Time	t_{DS}	17	-	ns

9.5 DRAM TIMING (Continued)

No.	Parameter	Symbol	Min	Max	Unit
24	Data In Hold-Time	t_{DH}	140	-	ns
25	Data In Hold Time Referenced to RAS	t_{DHR}	223	-	ns
26	Column to Row Strobe Precharge Time	t_{CRP}	140	-	ns
27	RAS Hold Time	t_{RSH}	112	-	ns
28	Refresh Period	t_{RFSH}	1.64	-	ms
29	Write Command Setup Time	t_{WCS}	56	-	ns
30	CAS Hold Time	t_{CSH}	196	-	ns
31	CAS Precharge, Non-Page Mode	t_{CPN}	140	-	ns
32	Page Mode Cycle Time	t_{PC}	223	-	ns
33	CAS Precharge Time (Page-Mode Cycle Only)	t_{CP}	112	-	ns
34	RAS Pulse Width (Page-Mode Cycle Only)	t_{RPM}	531	-	ns

9.6 AC CHARACTERISTICS (NTSC) (See Figures 9-5 through 9-7)

No.	Characteristic	Symbol	Min	Typ	Max	Unit
1	Clock Rise Time	t_{Cr}	0	-	10	ns
2	Clock Fall Time	t_{Cf}	0	-	10	ns
3	VTCLK Period	t_{VT}	-	140	-	ns
4	VTCLK High	t_{VTH}	42	-	70	ns
5	VTCLK Low	t_{VTL}	70	-	98	ns
-	VTCLK Duty Cycle	$VTDC$	30/70	-	50/50	%
6	MTCLK Period	t_{MT}	-	112	-	ns
7	MTCLK High	t_{MTH}	50	-	62	ns
8	MTCLK Low*	t_{MTL}	50	-	62	ns
-	MTCLK Duty Cycle	$MTDC$	40/60	-	60/40	%
9	VTCLK High to MTCLK High**	t_{VTHMTH}	-	-	± 5	ns
10	VTCLK High to PCLK High**	t_{VTHPH}	-	-	± 5	ns
11	HRES 0, 1, 2 PCLK Period	tp_2	-	140	-	ns
12	HRES 0, 1, 2 PCLK High	t_{PH2}	-	70	-	ns
-	HRES 0, 1, 2 PCLK Duty Cycle	$PDC2$	40/60	-	60/40	%
13	HRES 3 PCLK Period	tp_3	-	168	-	ns
14	HRES 3 PCLK High	t_{PH3}	-	84	-	ns
-	HRES 3 PCLK Duty Cycle	$PDC3$	40/60	-	60/40	%
15	HRES 4 PCLK Period	tp_4	-	126	-	ns
16	HRES 4 PCLK High	t_{PH4}	-	56	-	ns
-	HRES 4 PCLK Duty Cycle	$PDC4$	40/60	-	60/40	%
17	HRES 6 PCLK Period	tp_6	-	84	-	ns
18	HRES 6 PCLK High	t_{PH6}	-	42	-	ns

9.6 AC CHARACTERISTICS (NTSC)(Continued)

No.	Characteristic	Symbol	Min	Typ	Max	Unit
	HRES 6 PCLK Duty Cycle	tPDC6	40/60	-	60/40	%
19	HRES 7 PCLK Period	tp7	-	70	-	ns
20	HRES 7 PCLK High	tPH7	-	28	-	ns
	HRES 7 PCLK Duty Cycle	tPDC7	40/60	-	50/50	%
21	X Bus Valid to ADSEL Low	tXBASL	45	-	-	ns
22	X Bus Valid to ADSEL High	tXBASH	45	-	-	ns
23	X Bus Hold Time	tXBXH	5	-	-	ns
24	MTCLK High to X Bus Valid	tMTHXB	30	-	60	ns
25	MTCLK High to X Bus Three-State	tMTHXBT	30	-	50	ns
26	B Bus Valid to DBEN High (09 Write)	tBBDEH	15	-	-	ns
27	DBEN High to B Bus Valid (Read MEM/RMC)	tDEHBB	-	-	60	ns
28	DBEN Low to B Bus Invalid (Hold Time)	tDELBB	0	-	60	ns
29	B Bus Valid to CASTB High (Disp Setup)	tBBCBH	25	-	-	ns
30	DBEN High to A Bus Valid (09 Write)	tDEHAB	50	-	-	ns
31	DBEN Low to A Bus Valid (Hold Time)	tDELAB	0	-	10	ns
32	A Bus Valid to CASTB High (Disp Setup)	tABC BH	25	-	-	ns
33	MTCLK Low to ADSEL Low	tMTLASL	22	-	34	ns
34	ADSEL Width Low	tASL	100	112	125	ns
35	MTCLK Low to DBEN High (Write)	tMTLDE	-	-	35	ns

9.6 AC CHARACTERISTICS (NTSC)(Concluded)

No.	Characteristic	Symbol	Min	Typ	Max	Unit
36	MTCLK High to DBEN High (Read)	tMTHDEH	-	-	35	ns
37	DS High to DBEN Low (6800X Read)	tDSHDEL	-	-	75	ns
38	E Clock Low to DBEN Low (MC6809E Read)	tELDEL	-	-	75	ns
39	MTCLK High to DBEN Low (Write)	tMTHDEL	-	-	35	ns
40	MTCLK Low to CASTB Low	tMTLCBL	-	-	35	ns
41	CAS High to CASTB High**	tCSHCB	-	-	+/- 5	ns
42	R/W Valid Before ADSEL Low	tRWASL	80	-	-	ns
43	ADSEL Low to R/W Invalid (Hold Time)	tASLRW	280	-	-	ns
44	RTI Hold Time Low	tRTIL	100	-	-	ns
45	MTCLK High to REN Low	tMTREL	-	-	40	ns
46	MTCLK High to REN High	tMTREH	-	-	40	ns
47	MTCLK High to VIDEN Valid	tMTVE	-	-	35	ns
48	Field SYNC Input Width Low Non-Interlace Interlace (NTSC) Interlace (PAL)	tFSL	-	-	3 3 2.5	H***
49	R,G,B Settling Time (20 pF Load)	tRGBS	-	-	35	ns
50	G Rise and Fall to Sync Level (1.07 V)	tGr,tGf	-	-	100	ns

*Will be stretched every memory cycle in some HRES modes.

**Coincident rising edges.

***Equals one horizontal line time.

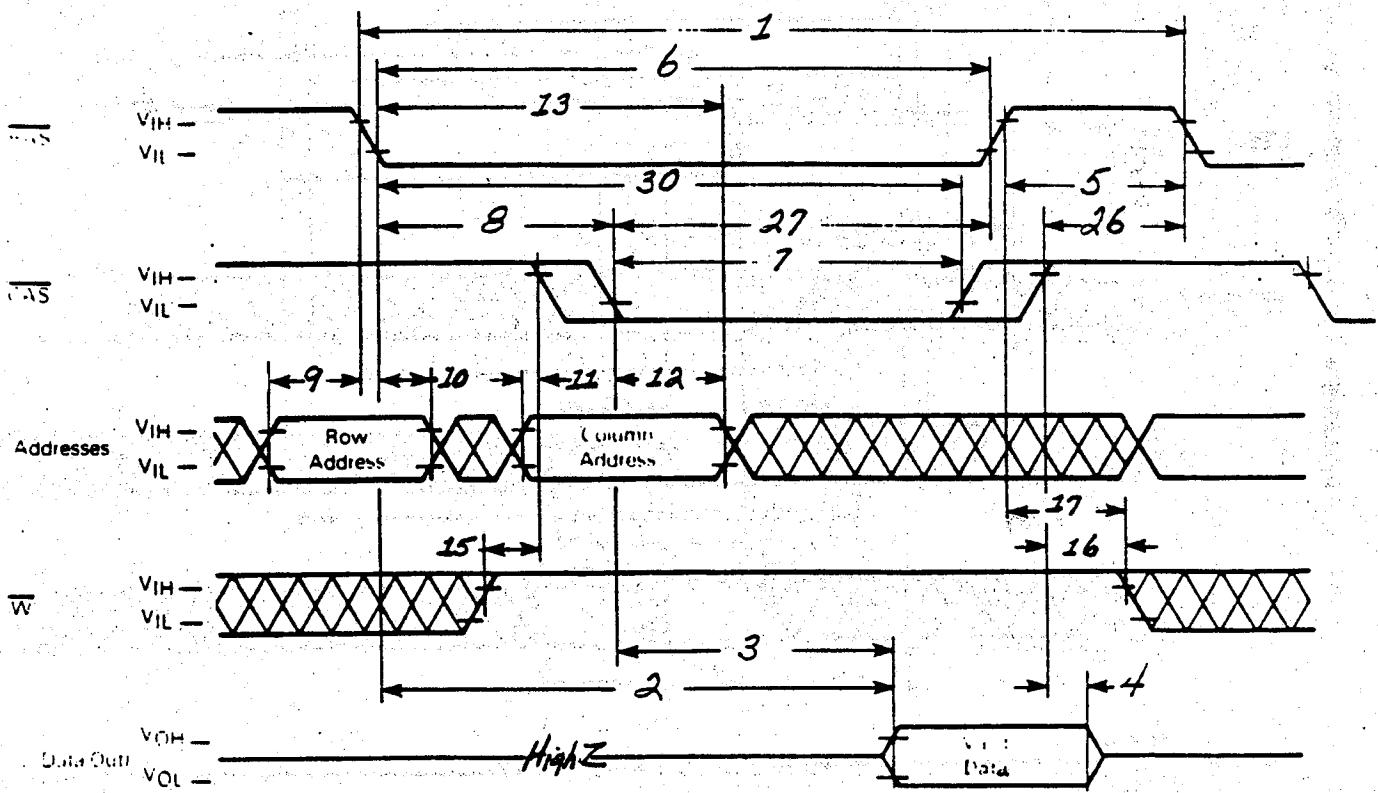


Figure 9-1. Read Cycle Timing Diagram

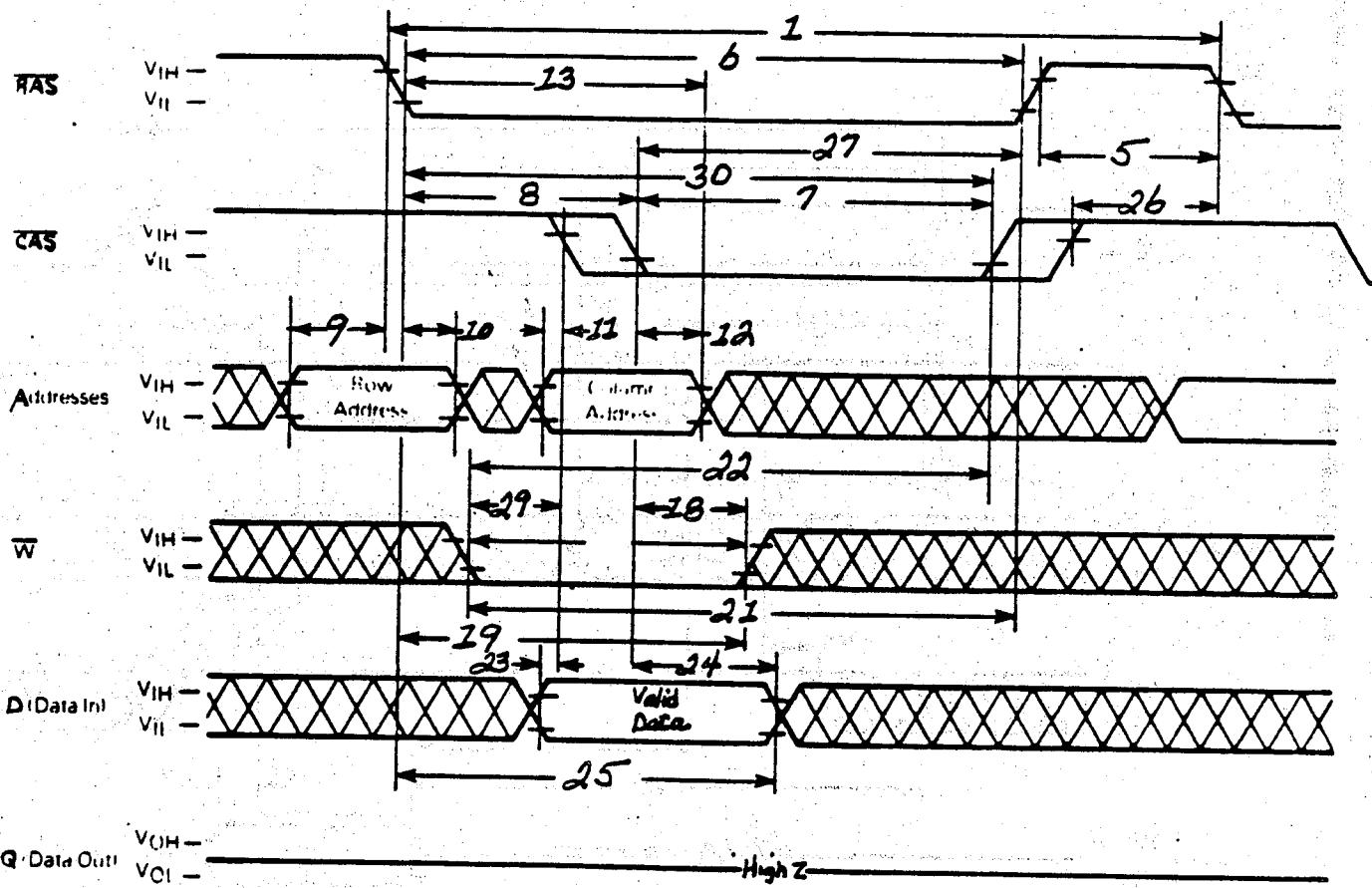


Figure 9-2. Write Cycle Timing Diagram

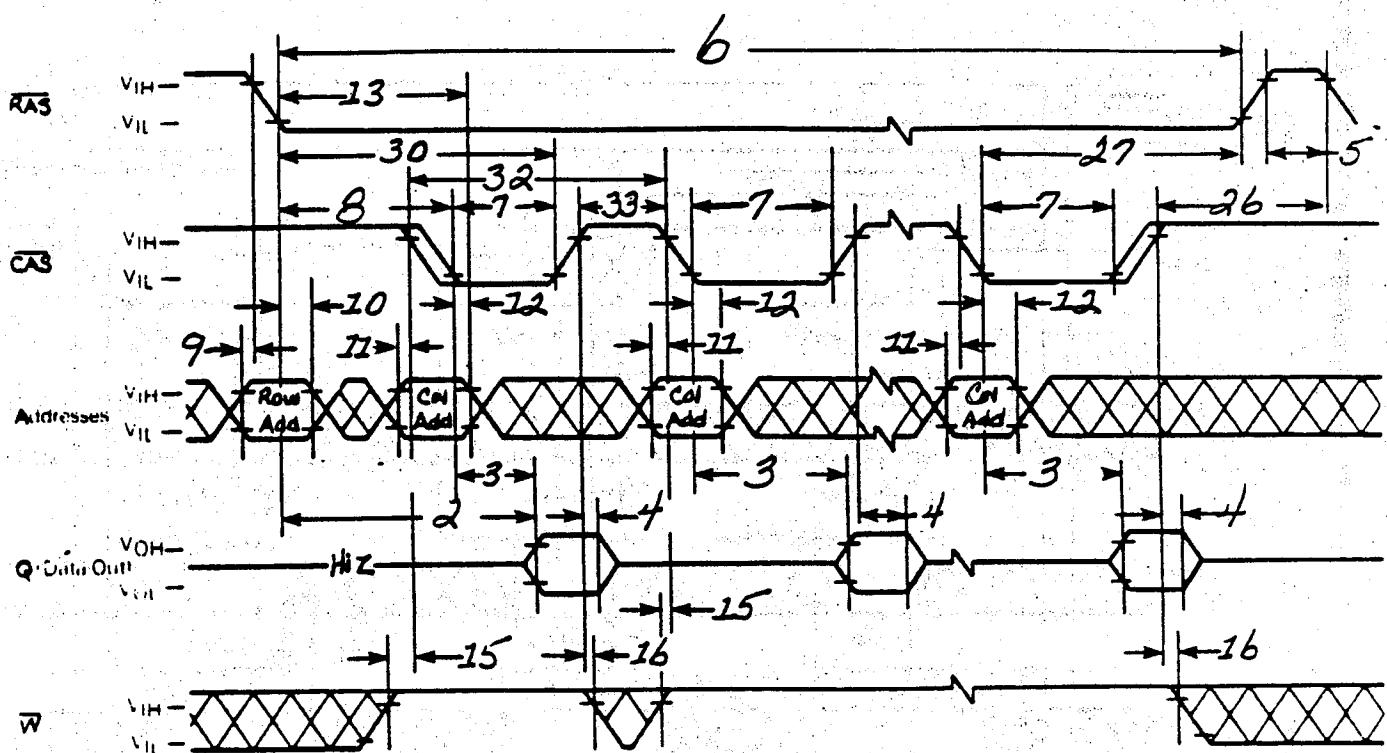


Figure 9-3. Page Mode Read Cycle Timing Diagram

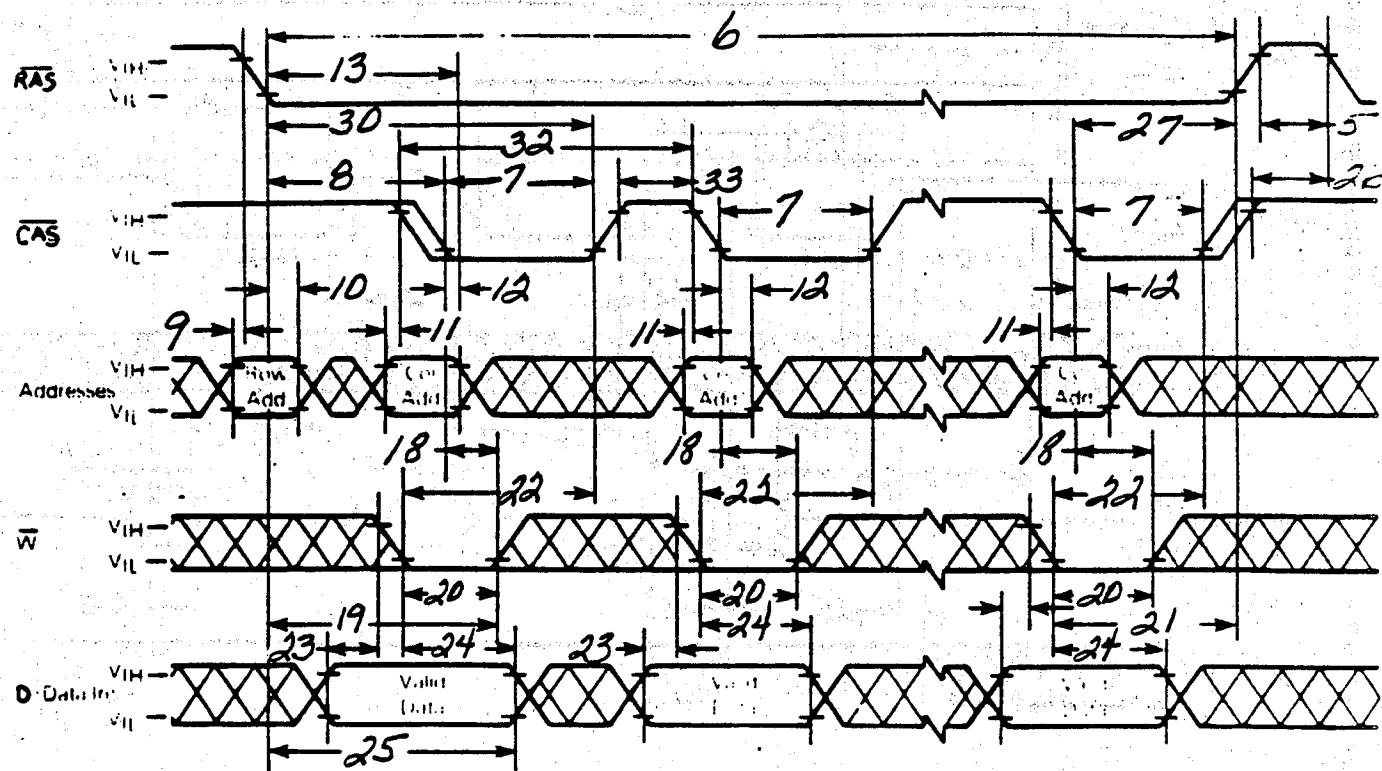


Figure 9-4. Page Mode Write Cycle Timing Diagram

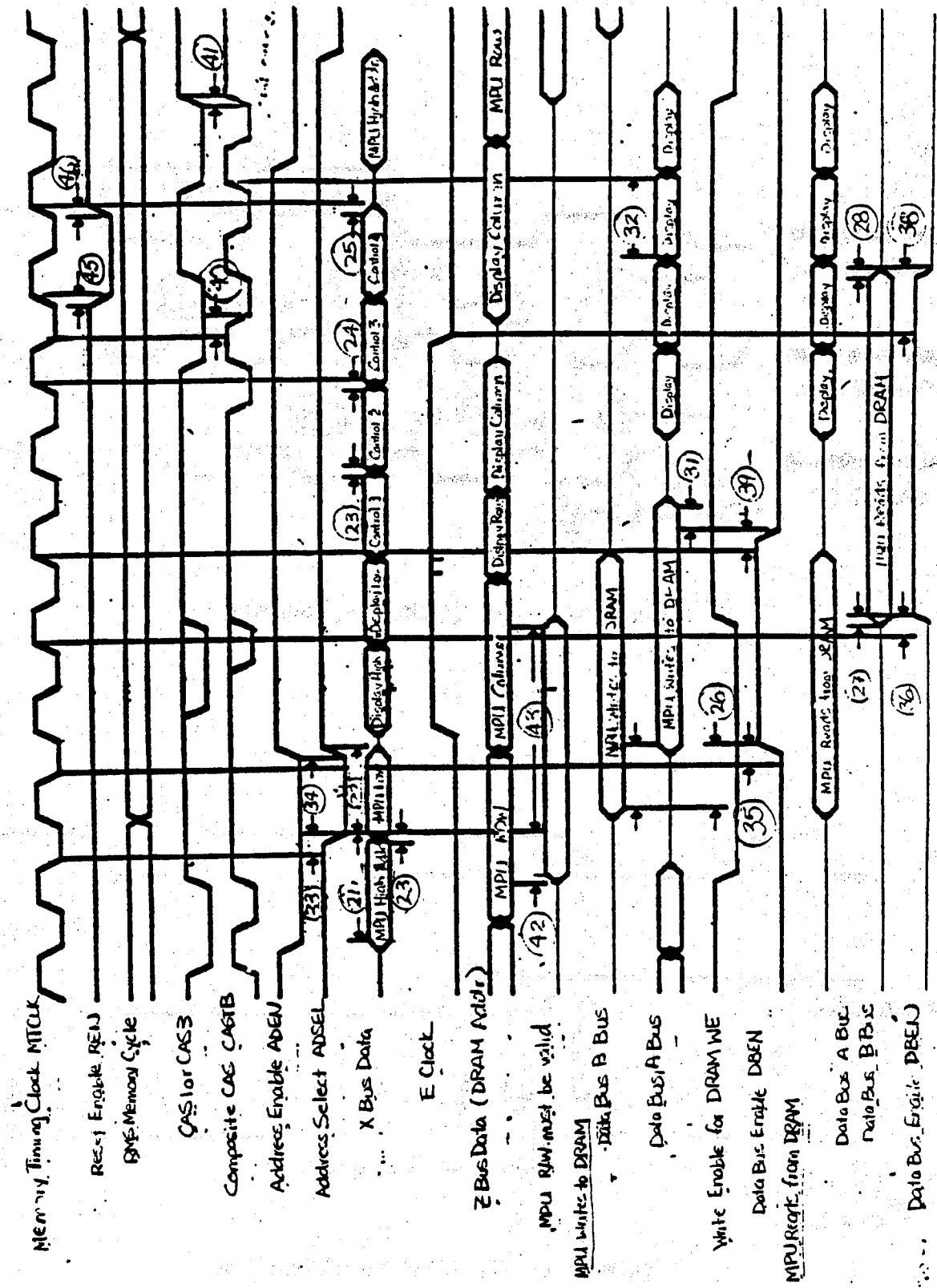


Figure 9-5. Raster Memory System Timing Diagram
(HRES4 with MC6809E MPU)

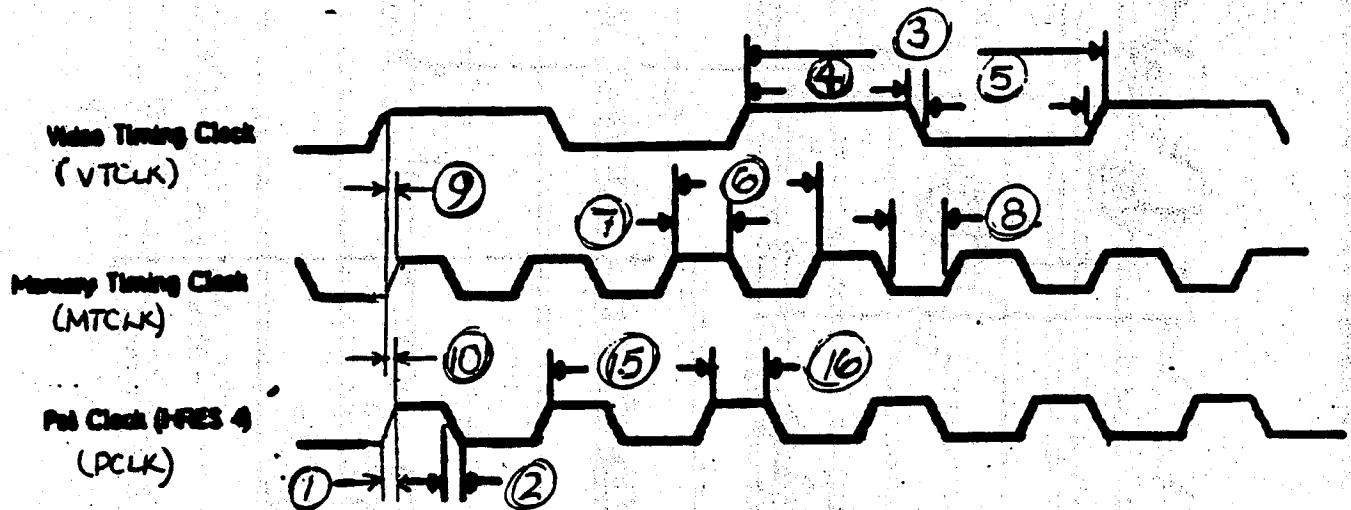


Figure 9-6. RMC Clock Relationships

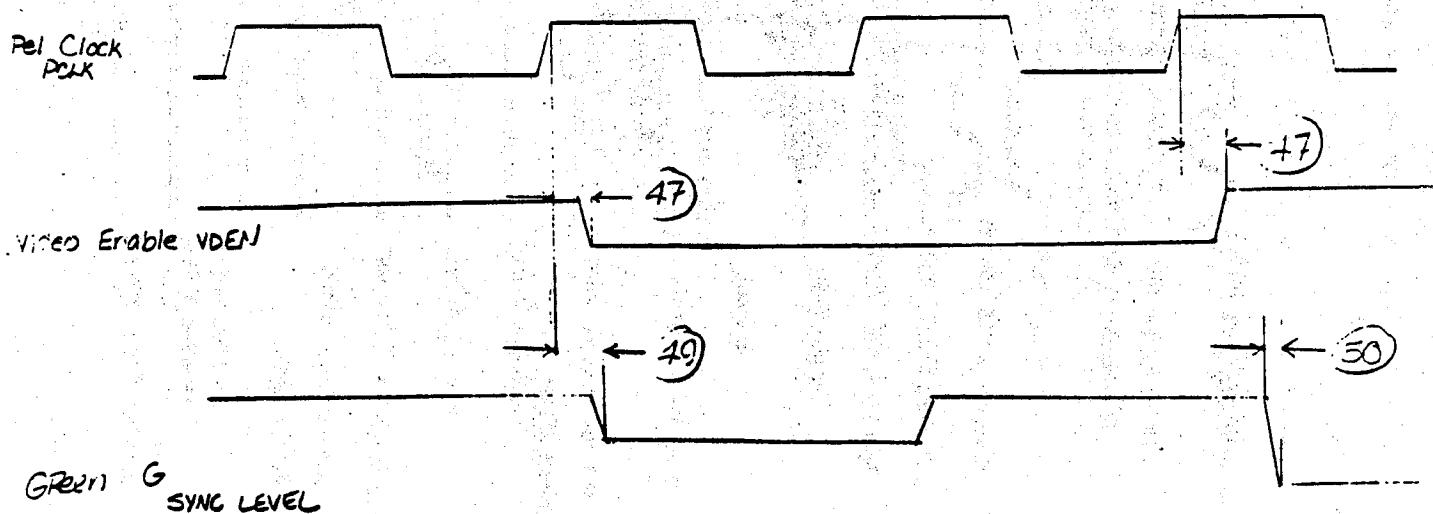


Figure 9-7. PEL Clock Relationships

9.7 525-LINE TIMING

525 line timing is available as either 525 lines per frame in full interlace mode or 262 lines per field in noninterlace mode. The following table is based on a 35.79545 MHz master oscillator.

Lines per Frame	Interlace 525 Noninterlace 524
Field Frequency	Interlace 59.94 Hz Noninterlace 60.19 Hz
Line Frequency	Interlace 15,734.26 Hz Noninterlace 15,768.92 Hz
Line Period	Interlace 63.5555 microseconds Noninterlace 62.4159 microseconds
Line Blanking Interval	Interlace 11.873 microseconds Noninterlace 11.733 microseconds
Front Porch	1.816 microseconds
Sync Pulse Duration	4.749 microseconds
Back Porch	5.308 microseconds
Field Blanking Period	21 H
Duration of First Series of Equalizing Pulses	3 H
Duration of Synchronizing Pulses	3 H
Duration of Second Series of Equalizing Pulses	3 H
Duration of Equalizing Pulses	2.375 microseconds
Duration of Field Sync Pulses	27.029 microseconds
Interval Between Field Sync Pulses	4.749 microseconds

NOTE: H equals one horizontal line time.

9.8 625-LINE TIMING

625-line timing is available either as 625 lines per interlace frame or as 312 lines per noninterlace field. The following table is based on a 35.46895 MHz master oscillator.

Lines per Frame	Interlace 625
Field Frequency	Noninterlace 624
Line Frequency	50.00 Hz
Line Period	15,625.08 Hz
Line Blanking Interval	63.999 microseconds
Front Porch	11.982 microseconds
Sync Pulse Duration	1.833 microseconds
Back Porch	4.793 microseconds
Field Blanking Interval	5.215 microseconds
Duration of First Series of Equalizing Pulses	27 H
Duration of Synchronizing Pulses	2.5 H
Duration of Second Series of Equalizing Pulses	2.5 H
Duration of Equalizing Pulse	2.396 microseconds
Duration of Field Sync Pulse	27.207 microseconds
Interval Between Field Sync Pulses	4,793 microseconds

NOTE: H equals one horizontal line time.

SECTION 10

ORDERING INFORMATION AND MECHANICAL DATA

This section contains information to be used as a guide when ordering the MC68487, the pin assignment, and package dimension diagram.

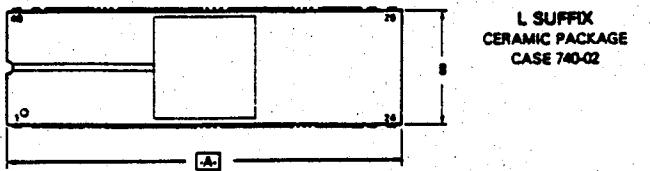
10.1 ORDERING INFORMATION

<u>Package Type</u>	<u>Temperature</u>	<u>Order Number</u>
Plastic P Suffix	0°C to 70°C	MC68487P
Ceramic L Suffix	0°C to 70°C	MC68487L

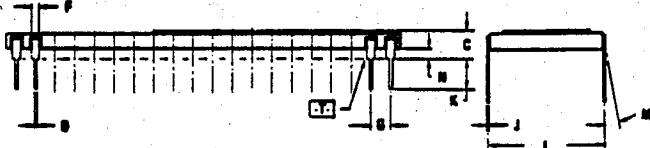
10.2 PIN ASSIGNMENT

B4	1	48	A5
A4	2	47	B5
B3	3	46	A6
A3	4	45	B6
B2	5	44	A7
A2	6	43	B7
<u>B1</u>	7	42	<u>CASTB</u>
A1	8	41	DBEN
B0	9	40	R/W
A0	10	39	T2
<u>INT</u>	11	38	T1
<u>V_{CC}</u>	12	37	GND
GND	13	36	<u>V_{CC}</u>
B	14	35	<u>HSYNC</u>
R	15	34	SYNC
G	16	33	PCLK
<u>VIDEN</u>	17	32	MTCLK
<u>RTI</u>	18	31	VTCLK
REN	19	30	X0
ADSEL	20	29	X1
X9	21	28	X2
X8	22	27	X3
X7	23	26	X4
X6	24	25	X5

10.3 PACKAGE DIMENSIONS



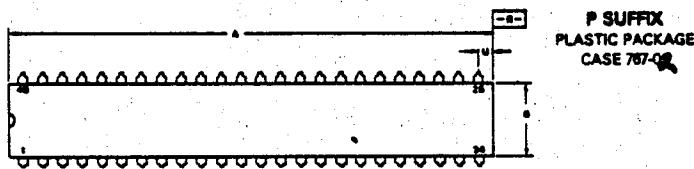
L SUFFIX
CERAMIC PACKAGE
CASE 740-02



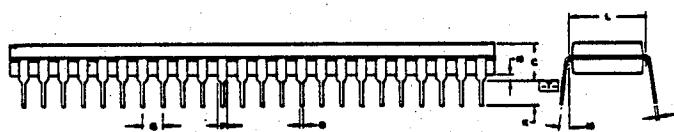
	MILLIMETERS		INCHES	
DIM	MIN	MAX	MIN	MAX
A	66.38	61.57	2.378	2.424
B	14.63	15.34	0.575	0.604
C	3.05	4.32	0.120	0.160
D	0.381	0.330	0.015	0.021
F	0.762	1.307	0.030	0.050
G	2.54 0.552	4.10 0.552		
J	0.203	0.330	0.008	0.015
K	2.54	4.10	0.100	0.160
L	14.59	15.35	0.575	0.610
M	0.00	100	0.000	100
N	1.016	1.524	0.040	0.060

NOTES:

1. DIMENSION **A** IS DATUM.
2. POSITIONAL TOLERANCE FOR LEADS:
 ± 0.25 (0.010) $\text{G}(\text{T})$ **A**
3. **C** IS SEATING PLANE.
4. DIMENSION "L" TO CENTER OF LEADS WHEN FORMED PARALLEL.
5. DIMENSIONING AND TOLERANCING PER ASME Y14.5, 1973.



P SUFFIX
PLASTIC PACKAGE
CASE 767-02



	MILLIMETERS		INCHES	
DIM	MIN	MAX	MIN	MAX
A	81.07	81.77	2.420	2.432
B	13.93	14.93	0.548	0.582
C	4.03	5.08	0.158	0.200
D	0.58	0.58	0.018	0.020
F	1.22	1.34	0.048	0.053
G	2.54 0.552	4.10 0.552		
J	0.28	0.28	0.010	0.012
K	3.23	3.37	0.127	0.133
L	15.24 0.552	16.00 0.552		
M	0.00	100	0.000	100
N	0.64	0.68	0.025	0.035
U	1.78 0.552	2.070 0.552		

NOTES:

1. **A** IS END OF PACKAGE DATUM PLANE.
2. **C** IS BOTH A DATUM AND SEATING PLANE.
2. POSITIONAL TOLERANCE FOR LEADS 24 & 25:
 ± 0.31 (0.012) $\text{G}(\text{T})$
- POSITIONAL TOLERANCE FOR LEAD PATTERN:
 ± 0.25 (0.010) $\text{G}(\text{T})$
3. DIM B DOES NOT INCLUDE MOLD FLASH.
4. DIM L IS TO CENTER OF LEADS WHEN FORMED PARALLEL.
5. DIMENSIONING AND TOLERANCING PER ASME Y14.5, 1973.

MOTOROLA

10-4

MC68487