012
Pete Gerrard

COMPANY CONFIDENTIAL

DEVICE NUMBER C020577				DRAFTED BY Mike Stevens	DATE 7/22/83	ATARI, INCORPORATED 275 GIBRALTAR DRIVE SUNNYVALE, CA 94086
REVISIONS				SYSTEMS ENGINEERING	8/11/83	
DOC. REV.	DEVICE REV.	COMMENTS	DATE	LOGIC ENGINEERING	8/11/83	ATARI Semiconductor Group
A	N/C	See Page 2		C AND D ENGINEERING	8/11/83	DEVICE NAME CGIA (NTSC)
				TEST ENGINEERING	8/11/83	
				ORIGINATING DIVISION		DOCUMENT NUMBER D020577
				DIVISION APPROVAL		PAGE 1 OF 90 PAGES

REVISIONS

Description

This document was revised to REV A in order to reflect a change to the device pin-out from the preliminary CGIA specification. The device revision was not changed because the CGIA had not been released to ATARI vendors.

Table of Contents

Section Description — Page Number

Table of Contents

Table of Contents

Table of Illustrations

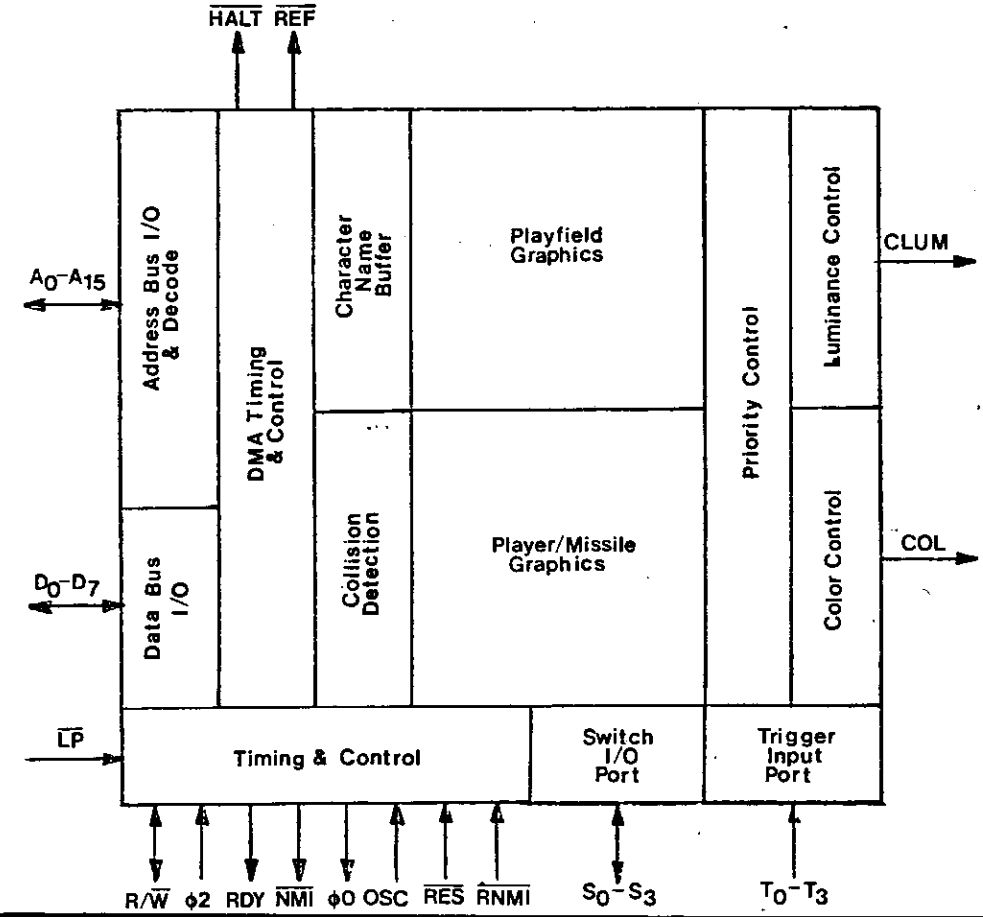
General Description

The CGIA is an NMOS custom circuit designed to generate sophisticated playfield and player/missile graphics for display on an NTSC television system. The CGIA converts graphics data stored in system memory into composite video and chrominance information required by a television R.F. modulator. The CGIA also provides four-bits of input from the joystick controllers and four-bits of general purpose I/O.

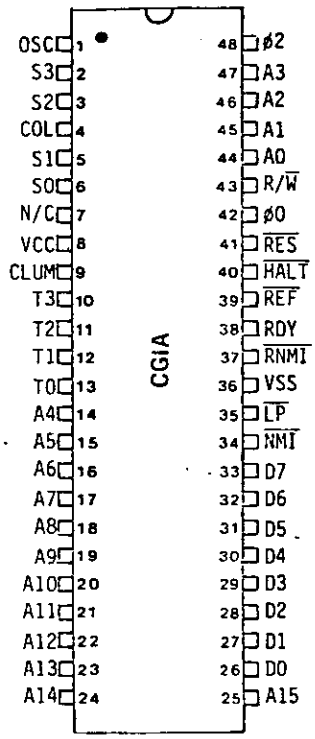
Features

- Single-Chip Version of the ANTIC and GTIA Chips
- Software Compatible with ANTIC and GTIA
- All Address Decode for ANTIC and GTIA On-Chip
- Composite Video Output
- Enhanced Color Frequency Generator
- All Graphics Functions On a Single Chip
- Increased System Reliability

Block Diagram



Pin Assignment



PIN DESCRIPTION

Pin Name	Type	Pin No.	Function
OSC	I	1	Master clock input--Typically 3.579 MHz.
S_0-S_3	I	2-3,5-6	Switch I/O lines which are used for general purpose I/O functions.
COL	O	4	Color frequency output signal--This output contains all of the video chroma information. In order to display different colors, the output frequency is shifted in time (phase angle) with respect to the "color burst" reference frequency.
N/C	*	7	Pin not internally connected.
V_{CC}	I	8	Positive voltage power supply-- Typically +5.0 volts.
CLUM	O	9	Analog video output signal--This output contains all of the video luminance and sync information (composite monochrome signal).
T_0-T_3	I	10-13	Data input lines which are used to input data from the joystick controllers. These inputs can be latched by setting a bit in the GRACTL register.
A_0-A_4 A_5-A_7 A_8-A_{15}	I/O O I/O	14,44-47 15-17 18-25	Address bus lines which are used to interface to the microprocessor and to system memory. As inputs, the address lines are used to select one of the internal registers of the CGIA. As outputs, the address lines are used to address graphics data stored in system memory and to output RAM refresh addresses.
D_0-D_7	I/O	26-33	Data I/O lines which are used to transfer data to and from the CGIA.
$\overline{NMI}$	O	34	Active low output signal which is used to generate a microprocessor non-maskable interrupt. A true condition is generated when there is a display list interrupt, vertical blank occurs, or the $\overline{RNMI}$ input goes low. Display list interrupts and vertical blank interrupts can be masked by clearing bits in the NMIEN register. The $\overline{RNMI}$ interrupt cannot be disabled.

PIN DESCRIPTION (cont'd)

Pin Name	Type	Pin No.	Function
$\overline{LP}$	I	35	Active low input which is used to interface the CGIA with an external light pen. When the $\overline{LP}$ input makes a high to low transition, the current VCOUNT value is stored in the PENV register. The horizontal position value is stored in the PENH register.
V_{SS}	I	36	Power supply ground.
$\overline{RNMI}$	I	37	Active low input which is used to generate a non-maskable interrupt ($\overline{NMI}$ output goes low). The $\overline{RNMI}$ interrupt cannot be disabled.
RDY	O	38	Output signal which is used to enable and disable the microprocessor. When the RDY line is low, the processor is put into an idle state. When the RDY line is high, the processor is active and enabled to execute program instructions. The RDY line is set low by writing to the WSYNC address, and will continue to stay low until horizontal blank occurs.
$\overline{REF}$	O	39	Active low output which is used to indicate that the address currently being output on the address lines is to be used for RAM refresh.
$\overline{HALT}$	O	40	Active low output control signal which is used to halt the microprocessor during graphics data DMA or RAM refresh.
$\overline{RES}$	I	41	Active low input which is used to reset the CGIA. When the $\overline{RES}$ line is low, bits D_7 and D_6 of the NMIEN register and bits D_2-D_5 of the DMACTL register are cleared. The CLUM output will be forced into the blank level for as long as the $\overline{RES}$ line is low. When the $\overline{RES}$ line makes the transition from low to high, the CLUM and COL lines will output background color until the display list is enabled.

PIN DESCRIPTION (cont'd)

Pin Name	Type	Pin No.	Function
$\phi 0$	O	42	Phase zero clock output for the 6502 microprocessor. Phase zero is generated from the OSC clock input and is equal to OSC/2 (typically 1.79 MHz).
R/$\overline{W}$	I/O	43	Input/output control signal which controls the direction of data transfers between the CGIA, microprocessor, and system memory. As an input, the R/$\overline{W}$ line controls data transfers between the CGIA and the microprocessor. When high, data is transferred from the CGIA to the microprocessor. When low, data is transferred from the microprocessor to CGIA. As an output, the R/$\overline{W}$ line is used to transfer data from system memory to the CGIA (R/$\overline{W}$ is always high).
$\phi 2$	I	48	Phase two microprocessor clock from the 6502 MPU. Phase two is used to synchronize data transfers between the CGIA and microprocessor.

Key to Pin Types:

I=Input
O=Output
I/O=Input/Output
*=Undefined Pin Type (pin not used)

INTRODUCTION

The CGIA is a custom NMOS LSI device that is capable of accessing graphics data stored in system memory and generating composite video and chroma signals necessary for interfacing to an NTSC video standard television. The CGIA is a single-chip version of the ANTIC (Graphics Processor) and GTIA (Television Interface Adapter) chips. The CGIA is completely software compatible with the ANTIC and GTIA chips. The CGIA requires fewer external components since the address decode for the GTIA has been added on-chip as well as the video D/A for generating composite video (luminance and sync signals). The color frequency generator with external color adjustment has been replaced with an enhanced self-correcting color frequency generator, thereby eliminating the need for external color adjustment.

CGIA FUNCTIONAL OVERVIEW

The CGIA is a simple microprocessor. It has a basic instruction set which it uses to generate the graphics on the television screen. Instructions are grouped together in a "display list," similar to a microprocessor program. The display list tells the CGIA how the screen is formatted and how to fetch the graphics data from memory. The CGIA has a program counter register (Display List Pointer) which is used to fetch program instructions. Display list instructions are either single or triple byte instructions.

Graphics are divided into two basic categories: playfield graphics and player/missile graphics. The display list has direct control over all playfield graphics. The playfield is generally used for inanimate objects, however, playfield animation is possible. Player/missile graphics are generally used to provide animation for objects that move on the screen.

Playfield Graphics (an overview)--Playfield graphics modes are divided into character-map graphics and bit-map graphics. There are six character-map modes and eight bit-map modes.

Character-map graphics use a group of pre-defined memory bits to create individual picture elements (pixels) which form a character pattern or "stamp." The character can be recreated or "stamped" on the screen many times without having to reproduce the graphics data in memory every time the character is used. The obvious advantage of character graphics is that it requires less memory to fill a screen. The disadvantage is that the number of characters are limited, therefore the number of different bit patterns are also limited.

Bit-map graphics use bits in memory to define individual pixels on the screen. The advantages and disadvantages of bit-map graphics are exactly the opposite of those for character-map graphics. The advantage is that there is total control of all the pixels on the screen. The disadvantage is that memory bits must be used to define every pixel on the screen therefore, more memory is required for bit-map graphics.

Player/Missile Graphics (an overview)--Player/missile graphics are not controlled by the display list. Players and missiles are defined by individual graphics registers. The graphics information for players and missiles will be the same for every television scan line unless the graphics registers are altered. Player/missile graphics data can be stored in memory and fetch every scan line if player/missile DMA (Direct Memory Access) is enabled. Each player has its own missile and both derive their color and luminance from the same color/lum register. Each player and missile has its own horizontal position register and can be moved in the horizontal direction by simply changing its horizontal position register. This allows players and missiles to be moved horizontally on the screen very easily. Vertical positioning is more difficult because the graphics data (if using DMA) must be moved in memory.

Graphics Mixing--In order to display both the playfield graphics and player/missile graphics on the same screen, both sets of graphics information must be combined. This is accomplished by establishing priority among the different graphics objects. Player/missile graphics are added to the playfield graphics when the horizontal position register for a player or missile equals the count value of the horizontal counter. If the player/missile object has priority over the playfield object, then the overlapping player/missile pixels will be displayed. If the playfield object has priority over the player/missile object, then the overlapping playfield pixels will be displayed. Priority between objects is determined by setting bits in the priority control register (PRIOR).

Object Collisions--When objects overlap, the CGIA records this occurrence in the form of collision detection bits. There are collision detection bits for detecting player to playfield collisions, missile to playfield collisions, player to missile collisions, and player to player collisions. There are however, no collision detection bits for playfield to playfield collisions since this condition cannot exist. The CGIA has no provision for missile to missile collisions.

Input/Output Functions--The CGIA has a four-bit input port for inputting data from the joystick controller trigger buttons and a four-bit general purpose I/O port. Data inputs on the trigger port can be latched by setting a bit in the graphics control register. The trigger inputs will remain latched until the latch enable bit in the graphics control register is reset. The switch I/O port is capable of data input and output. Switch lines are programmed to be either inputs or outputs by writing to the switch I/O output register.

1.0) Television Frame Timing

In order to understand how the CGIA works, it is necessary to understand how a television works. An electron beam is generated at the rear of the television tube and shoots toward the television screen in the front. Along the way, it passes through a set of horizontal and vertical coils which, if energized, can deflect the beam. In this way the beam can be made to strike any point on the television screen. The electronics in the television set cause the beam to sweep across the television screen in a regular fashion. The beam's intensity can also be controlled. If the beam is intense, the spot on the screen will glow brightly. If the beam is less intense, the spot will glow dimly or not at all.

The beam starts at the top-left corner of the screen and traces horizontally across the screen. As it sweeps across the screen, its intensity paints an image on the screen. When it reaches the right edge of the screen, it is turned off and brought back to the left side of the screen and down just a little. The beam is turned on again and starts back across the screen. In NTSC systems, this process is repeated for 262 sweeps across the screen. These 262 lines fill the screen from top to bottom to make a complete field. At the bottom of the screen (after the 262nd line), the beam is turned off and returned to the top-left corner of the screen. This process happens 60 times every second. Since the picture is drawn so fast, the eye does not notice that the television picture is being drawn one line at a time. All televisions use a technique called "interlacing." This technique increases the vertical resolution by drawing half of the television picture in the first 262 lines and the second half of the picture in the second 262 lines. The second half of the picture is moved down by one half of a scan line so the two picture halves are not displayed on top of each other. The CGIA does not do interlacing so the picture is the same for every field.

Terminology--A single trace of the beam across the screen is referred to as a "horizontal scan line." A horizontal scan line is the fundamental unit of measurement of vertical distance on the screen. The height of an image is stated by specifying the number of horizontal scan lines it spans. The period during which the beam returns from the right edge of the screen to the left side is referred to as "horizontal blank." The period during which the beam returns from the bottom-right edge of the screen to the top-left is referred to as "vertical blank." The entire process of drawing a screen takes 16,666 microseconds. The vertical blank period is about 1,400 microseconds. The horizontal blank period is about 11.16 microseconds. A single horizontal scan line takes approximately 64 microseconds (this includes the horizontal blank time).

1.0) Television Frame Timing (cont'd)

Most television sets are designed with "overscan." This means that the television picture is spread out so the picture edges are off the edge of the
any unsightly borders
ust be somewha
his overscan can vary
resolution to use is

or clock." The width
wide it is. There
an line. Only 160 of
rizontal overscan. It
dividual half clocks.
e picture elements. A
ed to as a "pixel."
ing the CGIA is 320
lustrates the screen

television tube. This guarantees that there will be no be
in the television picture. For this reason the picture m
smaller than the television can theoretically display. T
from television to television. A good limit of vertical
192 horizontal scan lines.

The standard unit of horizontal distance is the "col
of an image is specified by stating how many color clocks
are a total of 228 color clocks in a single horizontal sc
these are actually visible due to horizontal blank and ho
is possible with the CGIA to go even finer and control in
This gives double the horizontal resolution or 320 visibl
picture element, either vertical or horizontal, is referr
The maximum visible resolution of a television picture us
pixels horizontally by 192 pixels vertically. Figure 1 il
format as generated by the CGIA.

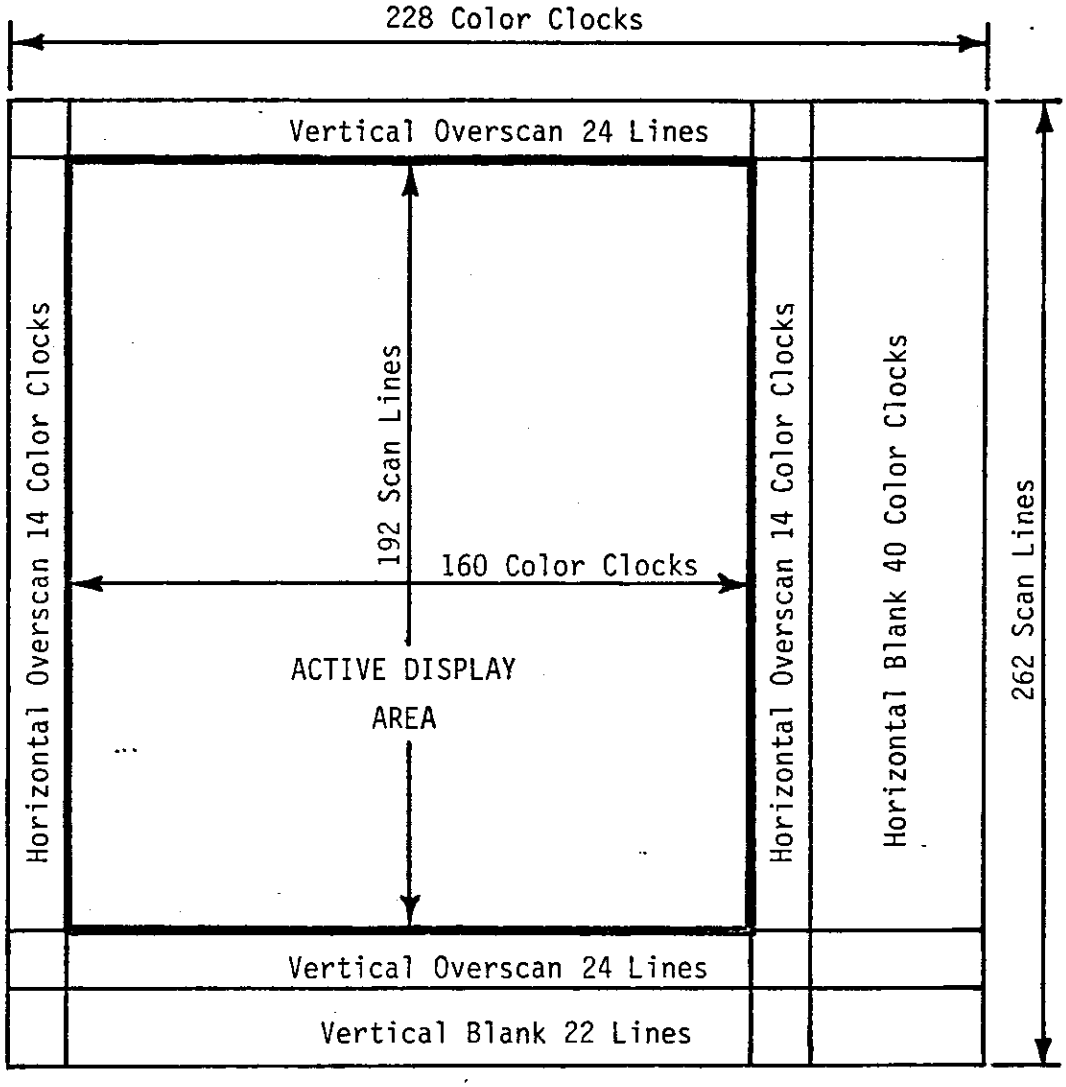


Figure 1) Television Screen Format

2.0) THE DISPLAY LIST

The display list is a sequence of display instructions stored in memory. These instructions are either one byte or three bytes long. The display list can be considered a display program and the display list pointer that fetches these instructions can be thought of as a display program counter. (6 bit base register plus 10 bit counter).

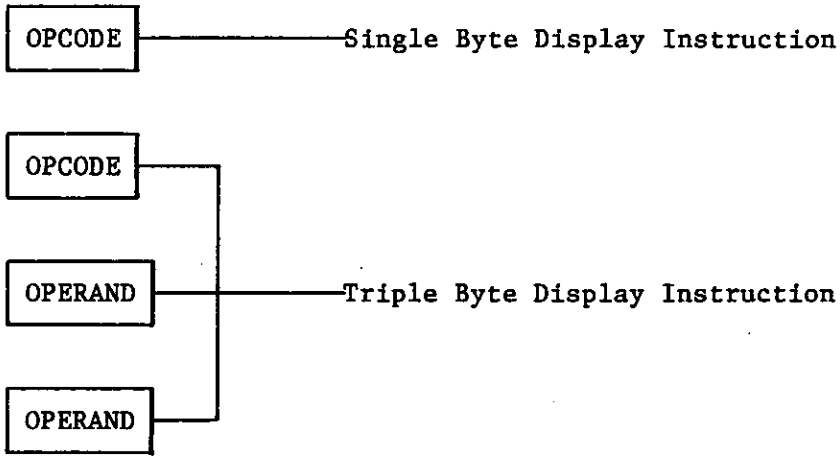
The display list counter can be initialized by writing to DLISTH and DLISTL. Once initialized, this counter value is used to address the display list, fetch the instruction, display one to sixteen lines of data on the TV screen, increment the display list pointer, fetch the next display instruction, and so on automatically without microprocessor control. DLISTL and DLISTH should be altered only during vertical blank or when DMA is disabled (see DMACTL).

Each instruction defines the type (character map or bit map) and the resolution (size of bits on the screen) and the location of data in memory to be displayed for a group (1 to 16) lines. Each group of lines is called a display block or mode line.

Note: The top 6 bits of the display list pointer are latches only and have no count capability, therefore the display list cannot cross a 1K byte memory boundary unless a jump instruction is used.

2.1) Display Instruction Format

Each instruction consists of either an opcode only, or of an opcode followed by two bytes of operand.



The opcode is always fetched first and placed in the Display Instruction Register. This opcode defines the type of instruction (1 or 3 bytes) and will cause two more bytes to be fetched if needed. If fetched, these next two bytes will be placed in the memory scan counter, or in the display list counter.

2.1) Display Instruction Format (cont'd):

Display Instruction Register (IR)-- This register is loaded with the opcode of the current display list instruction. It cannot be accessed directly by the programmer. There are three basic types of display list instructions: blank, jump, and display.

Blank
(1-byte)

D6	D5	D4	0	0	0	0

This instruction is used to create 1 to 8 blank lines on the display (background color used).

D6-D4 0-7 = 1-8 blank lines
D3-D0 0 = Blank

Jump
(3-bytes)

D6	X	X	0	0	0	1

This instruction is used to reload the Display List Counter. The next two bytes specify the address to be loaded (LSB first).

D6 0 = Jump (creates one blank line on display)
1 = Jump and wait until end of next vertical blank
D3-D0 1 = Jump

Display
(1 or 3 bytes)

D7	D6	D5	D4	D3	D2	D1	D0

This instruction specifies the type of display for the next display block.

D7 1 = Display list instruction interrupt
D6 0 = 1 byte instruction
1 = 3 byte instruction (reload Memory Scan Counter using address in next two bytes, LSB first)
D5 1 = Vertical scroll enable
D4 1 = Horizontal scroll enable
D3-D0 2-F = Graphics mode select (bit map or character map)

Bit D7 of a display list instruction can be set to create a display list interrupt, if bit D7 of the NMIEN register is set. Display list interrupts can be used to change the color registers or graphics during the middle of the TV display. The type of interrupt is determined by checking the NMIST register. The NMIRES register clears the NMIST register.

Bits D5 and D4 of a display list instruction are used to enable vertical and horizontal scrolling. The amount of scrolling depends on the values in the VSCROL and HSCROL registers. Figure 2 lists the CGIA display list opcodes.

HSCROL		XX		XX		XX		XX		XX		XX		XX		XX	Horizontal Scroll
VSCROL			XX	XX			XX	XX			XX	XX			XX	XX	Vertical Scroll
LD MEM SCAN					XX	XX	XX	XX					XX	XX	XX	XX	Load MSC (3 Byte)
INST INTERRUPT									XX	XX	XX	XX	XX	XX	XX	XX	Display List Interrupt
BLK 1	00								80								Blank 1 Scan Line
" 2	10								90								Blank 2 Scan Lines
" 3-7	**								##								Blank 3-7 Scan Lines
" 8	70								F0								Blank 8 Scan Lines
JMP	01								81								Jump (3 Byte)
JVB	41								C1								Jump and Wait for Vert. Blank (3 Byte)
CHR (40,2,8)	02	12	22	32	42	52	62	72	82	92	A2	B2	C2	D2	E2	F2	
" (40,2,10)	03	13	23	33	43	53	63	73	83	93	A3	B3	C3	D3	E3	F3	
" (40,4,8)	04	14	24	34	44	54	64	74	84	94	A4	B4	C4	D4	E4	F4	Character Map Display Instructions
" (40,4,16)	05	15	25	35	45	55	65	75	85	95	A5	B5	C5	D5	E5	F5	
" (20,5,8)	06	16	26	36	46	56	66	76	86	96	A6	B6	C6	D6	E6	F6	
" (20,5,16)	07	17	27	37	47	57	67	77	87	97	A7	B7	C7	D7	E7	F7	
BIT (40,4,8)	08	18	28	38	48	58	68	78	88	98	A8	B8	C8	D8	E8	F8	
" (80,2,4)	09	19	29	39	49	59	69	79	89	99	A9	B9	C9	D9	E9	F9	
" (80,4,4)	0A	1A	2A	3A	4A	5A	6A	7A	8A	9A	AA	BA	CA	DA	EA	FA	
" (160,2,2)	0B	1B	2B	3B	4B	5B	6B	7B	8B	9B	AB	BB	CB	DB	EB	FB	Bit Map Display Instructions
" (160,2,1)	0C	1C	2C	3C	4C	5C	6C	7C	8C	9C	AC	BC	CC	DC	EC	FC	
" (160,4,2)	0D	1D	2D	3D	4D	5D	6D	7D	8D	9D	AD	BD	CD	DD	ED	FD	
" (160,4,1)	0E	1E	2E	3E	4E	5E	6E	7E	8E	9E	AE	BE	CE	DE	EE	FE	
" (320,2,1)	0F	1F	2F	3F	4F	5F	6F	7F	8F	9F	AF	BF	CF	DF	EF	FF	

——————Number of TV Scan Lines Per Pixel
——————Number of Colors (Background and Playfield Types)
——————Number of Characters or Pixels Per Standard Mode Line

** = 20,30,40,50,60 Respectively
= A0,B0,C0,D0,E0 Respectively

Figure 2) Display List Instruction Opcodes

2.2) Memory Scan Counter

The memory scan counter is not directly accessible by the programmer. It is loaded with the value in the last 2 bytes of a 3 byte (non-jump) display list instruction.

This counter points to the location in memory where the graphics data to be directly displayed (bit map modes) is stored, or to the location of character name strings to be indirectly displayed (character map modes), on the television screen.

A single byte instruction does not reload this counter. This implies a continuation in memory of the data to be displayed from that displayed by the previous instruction. Since this counter consists of 4 bits of register and 12 of actual counter, a continuous memory block of display data can not cross a 4K byte memory boundary unless the counter is repositioned with a 3 byte Load Memory Scan counter display instruction.

2.3) Vertical and Horizontal Fine Scrolling

Playfield objects are difficult to move smoothly. Memory map playfield can be moved by rewriting sections of memory. This is extremely time consuming if large sections of the screen must be moved smoothly. Character playfield objects can be moved easily in a rough fashion by changing the memory scan counter. This results in a large position jump from one character position to another. For this reason, the CGIA provides two registers (VSCROL and HSCROL) which allow smooth horizontal or vertical motion of up to one character width horizontally and up to one character height vertically. After this much smooth motion has been done, memory is rewritten or the memory scan counter is modified and smooth motion is resumed for another character distance.

Horizontal Scrolling--Only playfield and not players and missiles are scrolled when scrolling is enabled. Horizontal scrolling is enabled by setting display instruction bit D4 to a one. When horizontal scrolling is enabled, the display is shifted to the right by the number of color clocks specified by the contents of HSCROL. More bytes of data are needed for horizontal scrolling than normal. For a narrow playfield, there should be the same number of bytes per line as for standard playfield and no scrolling. Similarly, for standard playfield, the same number of bytes are required as for the wide playfield. For wide playfield, there is no change in the number of bytes and background color is shifted into the vacant locations.

Vertical Scrolling--A zone of playfield on the screen can be scrolled upward by using VSCROL and bit D5 of the display list instruction. The display blocks at the upper and lower boundaries of the zone must have variable vertical size. In particular, the first display block within that zone must be shortened for the top, and the last display block must be shortened from the bottom (i.e. not all of the top and bottom blocks will be displayed).

2.3) Vertical and Horizontal Fine Scrolling (cont'd)

The vertical dimension of each display block is controlled by a 4-bit counter within the CGIA called the "Delta Counter" (DCTR). Without vertical scrolling, it starts at zero on the first line and counts up to a standard value which is determined by the current display instruction. (Ex: For upper and lower case text display, the end value is 9. For 5 color character displays, it is 7 or 15.)

If bit D5 of the instruction remains unchanged between consecutive display blocks, the second block is displayed in the normal fashion. If bit D5 of the instruction goes from 1 to 0 between two consecutive display blocks, the second block will start with DCTR = 0, as usual, but will count up until DCTR = VSCROL, instead of the standard value. This shortens that display block from the bottom.

To define a vertically scrolled zone, the most direct method is to set bit D5 to a one in the first display instruction for that zone and in all consecutive blocks but the last one. If the VSCROL register is not rewritten on the fly, this results in a total scrolled zone that has a constant number of lines (provided that the VSCROL value does not exceed the standard individual block size). If N is the standard block size, the top block will be N - VSCROL lines (N is greater than VSCROL), and the last block will be VSCROL + 1 lines: (N - VSCROL) + (VSCROL + 1) = N + 1. Figure 3 is an example of a scrolled zone, top block, for 8 VSCROL values for N=8.

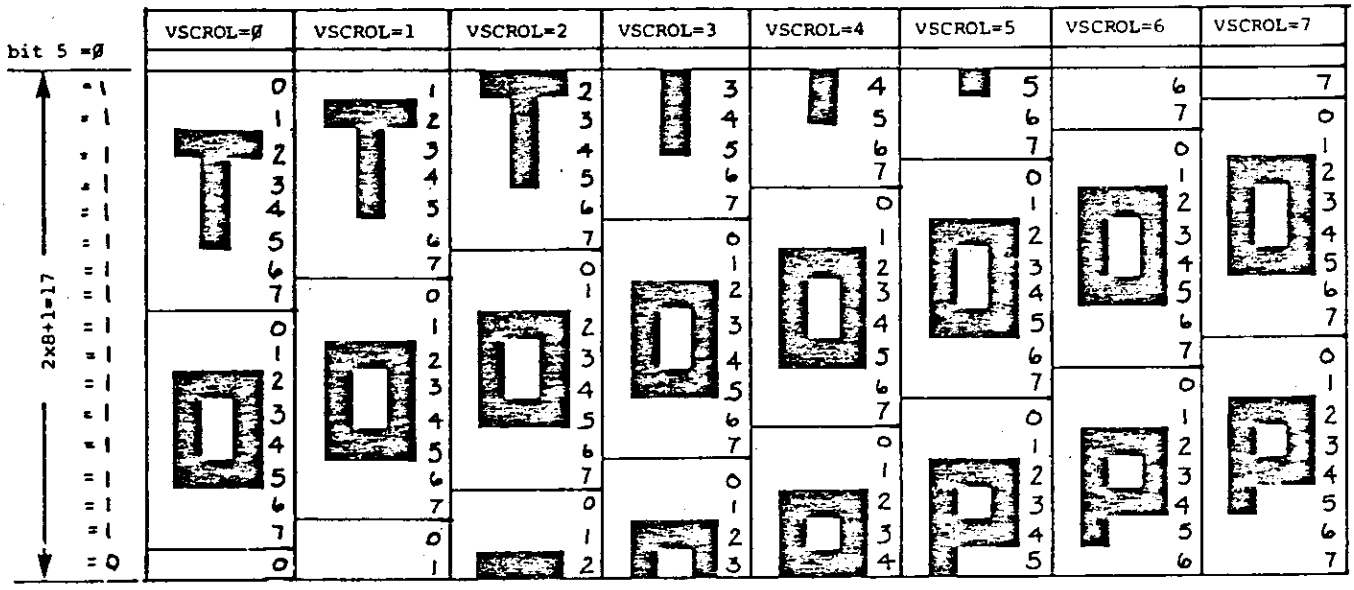


Figure 3) Vertical Scrolling Example

2.4) Simple Display List Example

The following display list example attempts to show how to create a display list for a character graphics mode 2 display screen. The DMACTL register is configured for a standard width playfield and display list instruction DMA enabled. This will allow a total of 40 characters per mode line with 24 mode lines per screen. The display list starts at address 1000 (HEX). DLISTH and DLISTL must already be set to 1000. Character names for the display are stored at address 1100 (HEX).

Graphics Mode 2 Display List Example (40 characters by 24 lines)

Address	Data	Comments
1000	70	
1001	70	Display 24 blank lines (output background color).
1002	70	
1003	42 00 11	Reload MSC with 1100 (starting address of character name data) and display one Mode 2 display line.
1006	02	
1007	02	
1008	02	
1009	02	
100A	02	
100B	02	
100C	02	
100D	02	
100E	02	
100F	02	
1010	02	
1011	02	
1012	02	Display 23 more Mode 2 display lines.
1013	02	
1014	02	
1015	02	
1016	02	
1017	02	
1018	02	
101A	02	
101B	02	
101C	02	
101D	02	
101E	41 00 10	Jump back to 1000 (start of display list)and wait for the end of vertical blank.
1100	XX	
--	--	960 bytes of character names
14BF	XX	

3.0) GRAPHICS MODES

As previously mentioned, there are two types of graphics that can be displayed on the television screen: playfield graphics and player/missile graphics. Playfield graphics are always generated by a process known as direct memory access or DMA. There are four playfields, each identified by its own color/luminance register.

Playfield is generated by two playfield DMA techniques: character map and bit map. Player/missile graphics can be generated by DMA or by software. Players and missiles are different from playfields in that they only occupy a small portion of the screen and can be positioned horizontally on the screen just by changing the contents of their horizontal position registers. The following sections describe in detail the different aspects of playfield and player/missile graphics.

3.1) Character Map Graphics Modes

In order to use the character map modes a character font has to be created. The character font consists of eight bytes of data per character which is stored in system memory. The selected display mode determines how this data is interpreted by the CGIA. The display mode also determines the total number of characters in the character set. The character set can contain either 64 or 128 characters. The most significant six or seven bits (A_{15}-A_{10},A_9) of the address of the character font data is stored in CHBASE. The remaining one or two bits and the least significant eight bits (A_7-A_0) are assumed to be zero, so the character font data must start on an acceptable page boundary.

The 40 characters/line modes (2-5) use the six most significant bits of CHBASE, which forces the character font data to start on a 1K byte memory boundary. The character font must contain 128 characters with each character defined by 8 bytes of data. This means that a total of 1024 bytes of memory are required to define the character font.

The 20 characters/line modes (6-7) use the seven most significant bits of CHBASE, which forces the character font data to start on a 512 byte memory boundary. The character font must contain 64 characters with each character defined by 8 bytes of data. This means that a total of 512 bytes of memory are required to define the character font.

Character map graphics are different from bit map graphics in many ways. The most significant difference is that the character graphics are generated by indirect methods. Each set of character font data bytes is given a character value or "name." In the 40 characters/line modes, the character values range from 00 (HEX) to 7F (HEX) for a total of 128 distinct characters. In the 20 characters/line modes, the character values range from 00 (HEX) to 3F (HEX) for a total of 64 distinct characters.

The memory scan counter (MSC) is generally used to fetch graphics information to be displayed. The contents of the memory scan counter can only be altered by using the LMS (Load Memory Scan counter) display instruction.

3.1) Character Map Modes (cont'd)

In the case of character mode graphics, the MSC is used to fetch character names which have been stored in memory. The character name in turn becomes part of the address (A_9,A_8-A_3) used to access the character graphics data which eventually appears on the TV screen. Simply stated, the character name indirectly points to the location in memory where the character's graphics font data is stored. The least significant three bits (A_2-A_0) of the character data address come from an internal line counter which counts from 0 to 7. At the beginning of a character mode line, this counter is set to zero and is incremented by one at the end of every pixel line. This is how the individual data bytes of the character's font data is accessed.

Character data is displayed on the screen in a left to right, top to bottom fashion. This means that data is displayed starting at the top-left position of the screen and continues to be displayed horizontally across the screen to the right. When the right edge of the screen is reached, a new display line is started at the left side of the screen just below the line previously displayed. This process can be compared to writing or typing on a piece of paper. Figure 5 illustrates how a display mode 2 character is created in memory and accessed for display on the TV screen.

There are six different character modes on the CGIA. These modes are similar in the way that they are used but differ in the number of characters per mode line, number of colors displayed, number of characters in the character set, and the number of bytes to define the character font. Figure 4 lists the characteristics of the six different character modes.

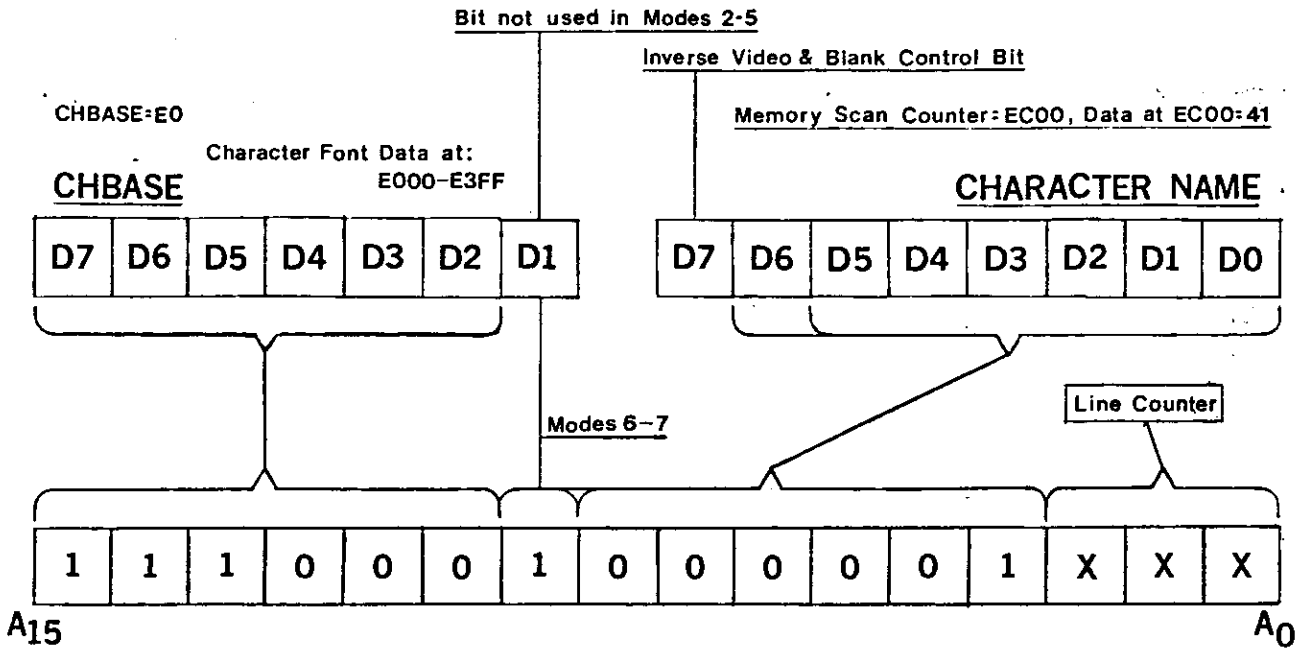
3.2) Graphics Modes 2 and 3

Character graphics modes 2 and 3 are the highest resolution character modes available on the CGIA. Mode 2 characters are 8 pixels wide by 8 pixels tall. Each pixel is equal to ½ color clock horizontally by 1 TV scan line vertically. Mode 3 characters are identical to mode 2 characters in that the character block is 8 pixels wide by 8 pixels tall. However, mode 3 characters have two additional scan lines that are used for displaying lower case alpha-numeric characters with descenders. The last fourth of the mode 3 character set (name bits D5 and D6 equal to one) is reserved for lower case characters. If the character being displayed is a lower case character, the CGIA takes the first two character data bytes and moves them to the bottom of the character, displaying two blank lines at the top of the character (see figure 6). Upper case characters have two blank lines displayed at the bottom.

In character modes 2 and 3, character name bit D7 is used to indicate whether the character is to be displayed in inverse video and/or blanked. The CHACTL register is used to enable or disable inverse video and blanking as well as vertical reflect. If bit D2 of CHACTL is set, then all characters will be displayed upside down, regardless of mode. If CHACTL bit D1 is set, then each character which has name bit D7 equal to one will be displayed in inverse video. Inverse video simply means that the off pixel assumes the color/lum of

Display Opcode (HEX)	Colors Per Mode	Char. Per Std. Line	TV Scan Lines Per Char.	Color Clocks Per Pix.	Color Bits In Name	Bit Values In Pix.	Color Reg. Selected
X2	2	40	8	½	-	0 1	COLPF2 COLPF1 (Luminance)
X3	2	40	10	½	-	0 1	COLPF2 COLPF1 (Luminance)
X4	5	40	8	1	0 0 0 0 1	0 0 0 1 1 0 1 1 1 1	COLBK COLPF0 COLPF1 COLPF2 COLPF3
X5	5	40	16	1	0 0 0 0 1	0 0 0 1 1 0 1 1 1 1	COLBK COLPF0 COLPF1 COLPF2 COLPF3
X6	5	20	8	1	--- 0 0 0 1 1 0 1 1	0 1 1 1 1	COLBK COLPF0 COLPF1 COLPF2 COLPF3
X7	5	20	16	1	--- 0 0 0 1 1 0 1 1	0 1 1 1 1	COLBK COLPF0 COLPF1 COLPF2 COLPF3

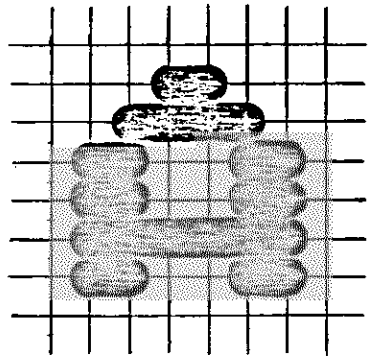
Figure 4) Character Map Display Mode Characteristics



CHARACTER DATA ADDRESS

D7	D6	D5	D4	D3	D2	D1	D0	
0	0	0	0	0	0	0	0	E208
0	0	0	1	1	0	0	0	E209
0	0	1	1	1	1	0	0	E20A
0	1	1	0	0	1	1	0	E20B
0	1	1	0	0	1	1	0	E20C
0	1	1	1	1	1	1	0	E20D
0	1	1	0	0	1	1	0	E20E
0	0	0	0	0	0	0	0	E20F

Character Data in Memory



Character Display on Screen

Figure 5) Character Data Addressing Example

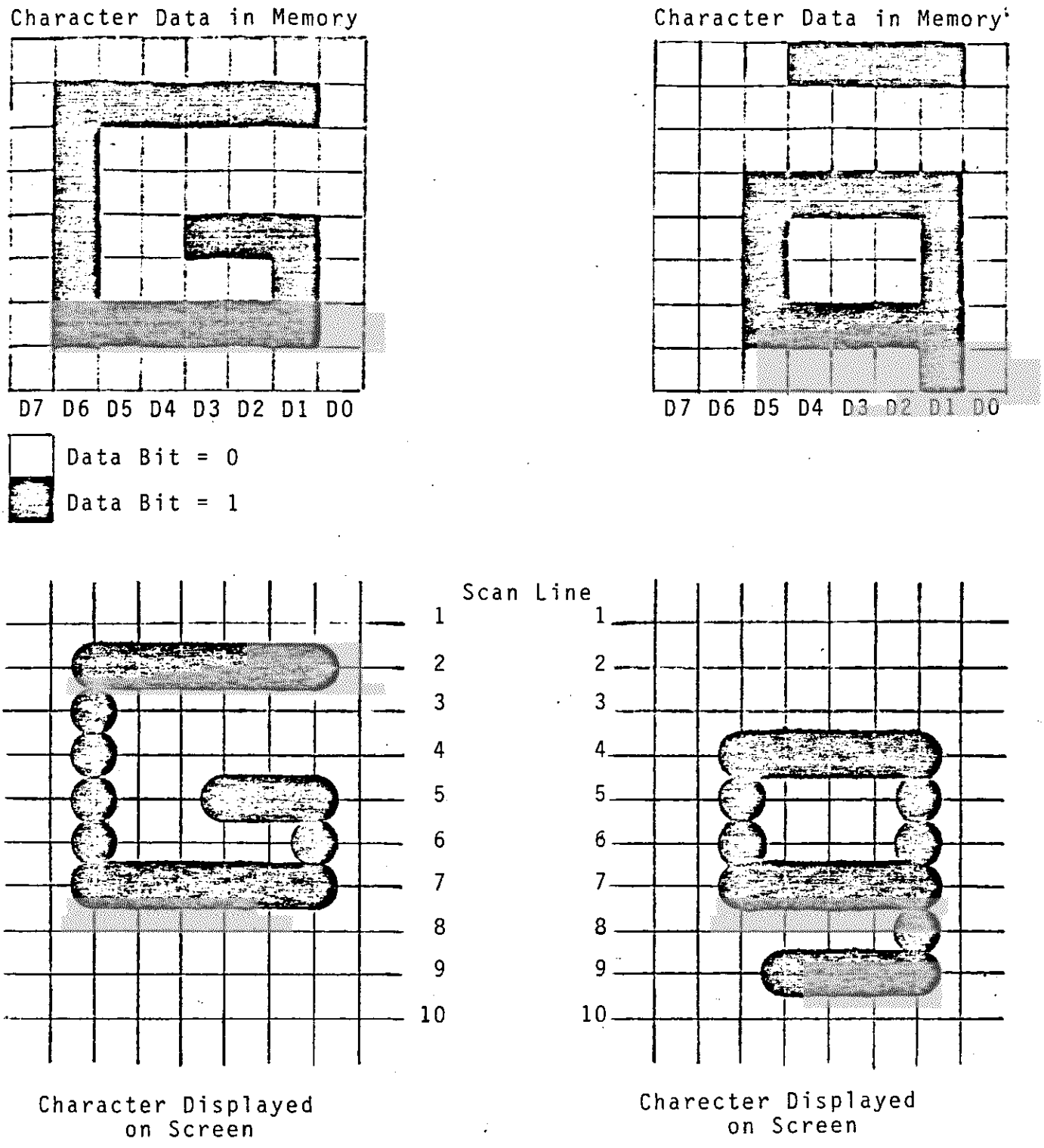


Figure 6) Graphics Mode 3 Display Example

the on pixel and the on pixel assumes the color/lum of the off pixel. Characters can also be blanked (all character block pixels turned off). This is accomplished by setting CHACTL bit D0 to one. Any character with name bit D7 set will be blanked. If both inverse video and blank are set, the character will appear as an inverse video blank character (solid square).

There are two display colors available in character modes 2 and 3. The off pixel gets its color and luminance from the playfield 2 color/lum register. The on pixel gets its color from the playfield 2 color/lum register and its luminance from the playfield 1 color/lum register. It can be seen that pixels are of the same color but differ in luminance. There are a total of 32 characters/line in a narrow width playfield, 40 characters/line in a standard width playfield, and 48 characters/line in a wide playfield. Figure 7 shows how character data bits and name bytes are organized.

2.3) Graphics Modes 4 and 5

Character graphics modes 4 and 5 are also 40 character modes but differ from modes 2 and 3 in resolution and the number of colors per character. Mode 4 and 5 characters are 4 pixels wide by 8 pixels tall, and each pixel can be one of five colors instead of one of two colors, as in modes 2 and 3. Each mode 4 pixel is equal to one color clock horizontally and one TV scan line vertically. Mode 5 pixels are equal to one color clock horizontally and two TV scan lines vertically (same data used for two scan lines).

As in all character modes, it takes 8 bytes of data to define a character. Modes 4 and 5 require two bits of data to define a pixel, hence four pixels per data byte. These two data bits select one of the color registers which gives the pixel its color. Two data bits allow only 4 colors to be selected (background plus 3 playfield colors). The character's D7 name bit allows the fourth playfield color to be used for a total of five colors. Pixel colors are selected as follows:

Character name bit D7=0

Pixel Bit Values (Binary)	Color Register Selected
0 0	COLBK
0 1	COLPF0
1 0	COLPF1
1 1	COLPF2

2.3) Graphics Modes 4 and 5 (cont'd)

Character name bit D7=1

Pixel Bit Values (Binary)		Color Register Selected
0	0	COLBK
0	1	COLPF0
1	0	COLPF1
1	1	COLPF3

Note that COLPF3 can only be selected if name bit D7 is set and the pixel bit values are 1,1. Character modes 4 and 5 are useful in generating graphics for scenes which are repetitive and can be built out of "blocks." The amount of memory used could be considerably less than would otherwise be required when using bit map graphics. Figure 7 shows how character data bits and name bytes are organized.

3.4) Graphics Modes 6 and 7

Character graphics modes 6 and 7 are 20 characters per line modes. Mode 6 and 7 characters are 8 pixels wide by 8 pixels tall. Each mode 6 pixel is equal to one color clock horizontally by one TV scan line vertically. Mode 7 pixels are also equal to one color clock horizontally but are equal two TV scan lines vertically (same data used for two scan lines).

In modes 6 and 7 the character's D7 and D6 name bits select on the the four playfield colors. For each character data bit that contains a one, the character pixel will be displayed using the selected playfield color and luminance. For each character data pixel that contains a zero, the character pixel will be displayed using the background color and luminance. This means that all of the on pixels of a character are of the same color, however, the same character can be displayed on the screen four times, with each character shown in a different color simply by changing the color bits in the character name. Character colors are selected as outlined below:

Character Data Bit Equal to Zero

Color Register Selected
COLBK

3.4) Graphics Modes 6 and 7 (cont'd)

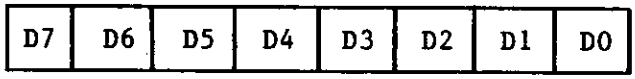
Character Data Bit Equal to One

Character Name Bits D7	D6	Color Register Selected
0	0	COLPF0
0	1	COLPF1
1	0	COLPF2
1	1	COLPF3

There are a total of 16 characters/line in a narrow width playfield, 20 characters/line in a standard width playfield, and 24 characters/line in a wide playfield. Figure 7 shows how character data bits and name bytes are organized.

Graphics Modes 2 and 3

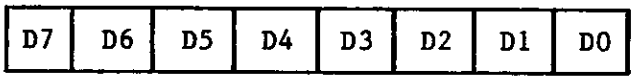
Data Byte



Left Most Pixel Right Most Pixel

D7-D0 = 1 Pixel color=COLPF2, pixel luminance=COLPF1
D7-D0 = 0 Pixel color=COLPF2, pixel luminance=COLPF2

Name Byte

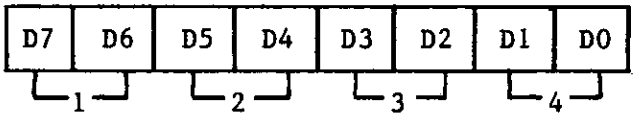


D7 = 1 Enable inverse video and character blank
D7 = 0 Disable inverse video and character blank

D6-D0 = 00-7F Character Name (address)

Graphics Modes 4 and 5

Data Byte

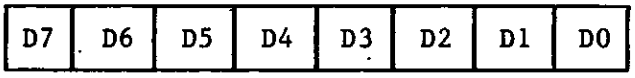


Pixel Number

Left most pixel Right most pixel

Pixel Value = 0,0 Pixel color/lum=COLBK
= 0,1 Pixel color/lum=COLPF0
= 1,0 Pixel color/lum=COLPF1
= 1,1 Pixel color/lum=COLPF2 or COLPF3

Name Byte



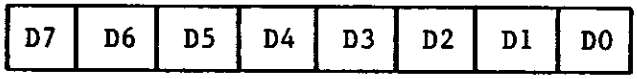
D7 = 1 Enable COLPF3 for pixel value 1,1
D7 = 0 Enable COLPF2 for pixel value 1,1

D6-D0 = 00-7F Character Name (address)

Figure 7) Character Data and Name Byte Organization

Graphics Modes 6 and 7

Data Byte

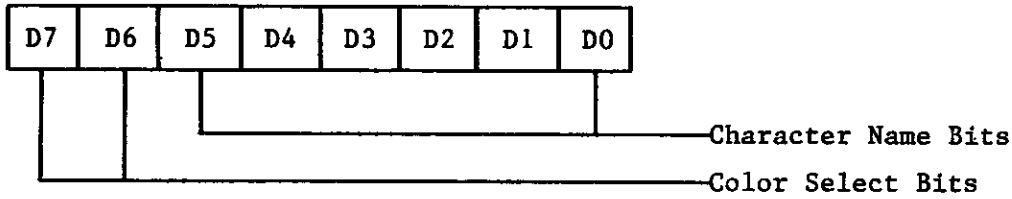


Left Most Pixel Right Most Pixel

D7-D0 = 1 Pixel color/lum taken from color select bits in name

D7-D0 = 0 Pixel color/lum=COLBK

Name Byte



D7-D6 = 0,0 Pixel data=1, pixel color/lum=COLPF0
= 0,1 Pixel data=1, pixel color/lum=COLPF1
= 1,0 Pixel data=1, pixel color/lum=COLPF2
= 1,1 Pixel data=1, pixel color/lum=COLPF3

D5-D0 = 00-3F Character Name (address)

Figure 7) Character Data and Name Byte Organization (cont'd)

3.5) Bit Map Graphics Modes

Bit map graphics are used when the picture to be displayed can not be created out of character blocks. Bit map graphics provide complete control of every pixel on the television screen.

Unlike character map graphics which use the memory scan counter to fetch character names from memory, bit map modes use the MSC to fetch data from system memory to be displayed directly on the screen. As with character data, bit map data is displayed on the TV screen in a left to right, top to bottom fashion.

Graphics data must be stored in memory in a precise manner in order to appear on the screen properly. Memory bytes are displayed bit-wise from left to right and from low addresses to higher addresses. This means that pixels defined by the most significant bit(s) of a data byte are displayed first and are either to the left or to the top of the pixels that are defined by the least significant bit(s) of the data byte. Likewise, pixels that are defined by a data byte whose address is higher than some other data byte will be displayed either to the right or to the bottom of the other pixels.

The CGIA provides eight different bit map modes from which to choose. These modes differ in pixel resolution, number of colors per pixel, and number of memory bits required to define a pixel. Figure 8 lists the characteristics of the eight different bit map modes.

3.6) Graphics Modes 8,A,D and E

Bit graphics modes 8,A,D and E provide the maximum number of colors per pixel of all the standard bit map graphics modes. They differ from each other in pixel resolution and the number of data bytes required to display a standard mode line. Each pixel is defined by two data bits allowing four pixels per data byte. These two bits of data are used to select on of three playfield colors plus background for a total of four possible colors. Pixel colors are selected as outlined below:

Pixel Data Bits (Binary)		Color Register Selected
0	0	COLBK
0	1	COLPF0
1	0	COLPF1
1	1	COLPF2

Graphics mode 8 provides 40 pixels per standard mode line and requires 10 bytes of data to define these 40 pixels. Each pixel is equal to 4 color clocks horizontally by 8 TV scan lines vertically. Graphics mode A provides 80 pixels per standard mode line and requires 20 bytes of data to define these 80 pixels. Each pixel is equal to 2 color clocks horizontally by 4 TV scan lines vertically. Mode A pixels have twice the resolution of mode 8 pixels. Graphics mode D provides 160 pixels per standard mode line and requires 40 bytes of data

Display Opcode (HEX)	Colors Per Pixel	Pixels Per Std. Line	Bits Per Pixel	Bytes Per Std. Line	TV Scan Lines Per Pix.	Color Clocks Per Pix.	Bit Values In Pix.	Color Reg. Selected
X8	4	40	2	10	8	4	0 0 0 1 1 0 1 1	COLBK COLPF0 COLPF1 COLPF2
X9	2	80	1	10	4	2	0 1	COLBK COLPF0
XA	4	80	2	20	4	2	0 0 0 1 1 0 1 1	COLBK COLPF0 COLPF1 COLPF2
XB	2	160	1	20	2	1	0 1	COLBK COLPF0
XC	2	160	1	20	1	1	0 1	COLBK COLPF0
XD	4	160	2	40	2	1	0 0 0 1 1 0 1 1	COLBK COLPF0 COLPF1 COLPF2
XE	4	160	2	40	1	1	0 0 0 1 1 0 1 1	COLBK COLPF0 COLPF1 COLPF2
XF	2	320	1	40	1	½	0 1	COLPF2 COLPF1 (Luminance)

Figure 8) Bit Map Display Mode Characteristics

3.6) Graphics Modes 8,A,D and E (cont'd)

to define these 160 pixels. Each pixel is equal to 1 color clock horizontally by 2 TV scan lines vertically. Mode D pixels have twice the resolution of mode A pixels. Graphics mode E also provides 160 pixels per standard mode line which are defined by 40 bytes of data. Each pixel is equal to 1 color clock horizontally by 1 TV scan line vertically. Mode E pixels have twice the vertical resolution of mode D pixels.

The number of pixels and data bytes required for non-standard playfield widths for modes 8,A,D and E are as follows:

Narrow Playfield (128 Color Clocks Wide)

Mode 8: 32 pixels per mode line
8 bytes of graphics data

Mode A: 64 pixels per mode line
16 bytes of graphics data

Modes D,E: 128 pixels per mode line
32 bytes of graphics data

Wide Playfield (192 Color Clocks Wide)

Mode 8: 48 pixels per mode line
12 bytes of graphics data

Mode A: 96 pixels per mode line
24 bytes of graphics data

Modes D,E: 192 pixels per mode line
48 bytes of graphics data

3.7) Graphics Modes 9,B and C

Bit graphics modes 9,B and C provide only two colors per pixel but do not require as many data bits to define a pixel. Each mode 9,B and C pixel is defined by a single data bit allowing eight pixels per data byte. This bit of data is used to select one of the two colors available for display. If the pixel value is a zero, the pixel will be displayed using the background color/lum register. If the pixel value is a one, the pixel will be displayed using the playfield 0 color/lum register. Modes 9,B and C differ only in pixel resolution and the number of data bytes required per mode line.

Graphics mode 9 provides 80 pixels per standard mode line and requires 10 bytes of data to define these 80 pixels. Each mode 9 pixel is equal to 2 color clocks horizontally by 4 TV scan lines vertically. Graphics mode B provides 160 pixels per standard mode line and requires 20 bytes of data to define these 160 pixels. Each mode B pixel is equal to 1 color clock horizontally by 2 TV scan

3.7) Graphics Modes 9,B and C (cont'd)

lines vertically. Mode B pixels have twice the horizontal and vertical resolution of mode 9 pixels. Graphics mode C also provides 160 pixels per standard mode line and requires 20 bytes of data to define these 160 pixels. Each mode C pixel is equal to 1 color clock horizontally by 1 TV scan line vertically. Mode C pixels have twice the vertical resolution of mode 9 pixels.

The number of pixels and data bytes required for non-standard playfield widths for modes 9,B and C are as follows:

Narrow Playfield (128 Color Clocks Wide)

Mode 9: 64 pixels per mode line
8 bytes of graphics data

Modes B,C: 128 pixels per mode line
16 bytes of graphics data

Wide Playfield (192 Color Clocks Wide)

Mode 9: 96 pixels per mode line
12 bytes of graphics data

Modes B,C: 192 pixels per mode line
24 bytes of graphics data

3.8) Graphics Mode F

Bit map graphics mode F is the highest resolution graphics mode available on the CGIA. Each pixel is equal to ½ color clock horizontally by 1 TV scan line vertically. Each pixel can be one of two colors. Colors are selected in the same manner as character modes 2 and 3. Only one bit of data is required to define each pixel. If the pixel value is equal to zero, the pixel takes its color and luminance from playfield color/lum register 2. If the pixel value is equal to one, the pixel takes its color from playfield register 2 and luminance from playfield color/lum register 1. Note that pixels are of the same color but can have different luminance values.

There are a total of 256 pixels per mode line when using a narrow playfield which are defined by 32 bytes of data. There are 320 pixels per mode line when using a standard width playfield which are defined by 40 bytes of data. Finally, there are 384 pixels per mode line when using a wide playfield which require 48 bytes of data.

3.9) Special Functions--Mode F (GTIA Modes)

The CGIA provides three expanded graphics modes. These modes are intended to be used in conjunction with bit map graphics mode F. The special graphics modes are enabled by setting certain bits in the priority control register (PRIOR). Priority control register bits D7 and D6 select one of the three special playfield graphics modes plus the standard CTIA playfield mode. These playfield modes are selected as follows:

Priority Control Register Bits		Mode Select
D7	D6	
0	0	Standard CTIA Mode (4 Colors-4 Luminances)
0	1	GTIA Mode 1 (1 Color-16 Luminances)
1	0	GTIA Mode 2 (9 Colors-9 Luminances)
1	1	GTIA Mode 3 (16 Colors-1 Luminance)

When the priority register bits are set for the standard CTIA mode, the CGIA functions as outlined in the previous sections. In the GTIA graphics modes (display instruction mode F and PRIOR bits D7,D6 $\neq$ 0,0), each pixel is equal to two color clocks horizontally by one TV scan line vertically. There are 80 pixels per standard mode line with each pixel defined by four data bits. This means that 40 bytes of graphics data are required to define a standard width mode line. The pixel can assume a variety of colors and luminances depending of the GTIA graphics mode chosen.

GTIA Mode 1 (1 Color-16 Lums)--In GTIA mode 1, the graphics data bits are used to select one of sixteen different luminance values. Each pixel takes its color from the background color/lum register. The luminance value selected by the pixel data bits is logically "ORed" with the luminance bits of the background color/lum register. In this mode, COLBK register bits D3-D1 should be set to zero (COLBK bit D0 is always equal to 0). Note that GTIA mode 1 allows luminance bit D0 to be displayed which is not available in the standard CTIA modes (luminance bit D0 is forced to be zero).

3.9) Special Functions Mode F (GTIA Modes)--cont'd

GTIA Mode 2 (9 Colors-9 Lums)--In GTIA mode 2, the graphics data bits are used to select one of the nine color/lum registers on the CGIA. The color registers are selected as follows:

Pixel Data Bits (Binary)	Color Register Selected
0 0 0 0	COLPM0
0 0 0 1	COLPM1
0 0 1 0	COLPM2
0 0 1 1	COLPM3
X 1 0 0	COLPF0
X 1 0 1	COLPF1
X 1 1 0	COLPF2
X 1 1 1	COLPF3
1 0 X X	COLBK

X=don't care

GTIA Mode 3 (16 Colors-1 Lum)--In GTIA mode 3, the graphics data bits are used to select one of sixteen different color values. Each pixel takes its luminance from the background color/lum register. The color value selected by the pixel data bits is logically "ORed" with the color bits of the COLBK color/lum register. In this mode, COLBK register bits D7-D4 should be set to zero.

3.10) Player/Missile Graphics

Players and missiles are small objects which are not considered to be part of the playfield. There are four players available on the CGIA. Each player is associated with its own missle. Each player/missile pair derives its color and luminance from one of the four player/missile color/lum registers. Players and missiles are created from data bits stored in individual graphics registers. Each player is eight bits wide and has its own eight-bit graphics register. Each missile is two bits wide and shares a graphics register with the other three missiles. Player/missile graphics registers are usually changed during the horizontal and vertical blank periods to avoid unwanted glitches on the TV screen. Each player/missile pixel is normally equal to one color clock horizontally by one TV scan line vertically. The width and height of each pixel can be changed by setting bits in the player/missile size registers and DMA control register. By changing bits in the player/missile size registers, each player/missile pixel can be changed from one color clock wide horizontally to two or four. By changing bit D4 of the DMACTL register from a one to a zero, all player/missile pixels can be changed from one TV scan vertically to two.

All players and missiles have their own 8-bit horizontal position registers. The values of the horizontal position registers range from 0 to FF (HEX). HEX 30 is the left edge of a standard width playfield screen and HEX D0

3.10) Player/Missile Graphics (cont'd)

is the right edge. When the horizontal counter value equals a player or missile's horizontal position register value, the graphics information in the corresponding graphics register is "added" to the playfield information. In order to decide which pixel (playfield or player/missile) should be displayed, object priorities are established. A player or missile graphics data bit that is equal to zero has no priority, so the playfield pixel being displayed will appear on the TV screen. Player/missile graphics data bits that are equal to one are given priority as determined by the contents of the priority control register. Refer to the section on graphics priority control for more information on object priority.

The same player/missile data will be displayed every TV scan line unless the player/missile graphics register data is changed. This will tend to create bands or strips that extend from the top to the bottom of the screen at the horizontal position of each player and missile. If players and missiles are not used, the player/missile graphics registers should be loaded with all zeros. This will keep the player missile bands from appearing on the screen. The horizontal position registers can also be loaded with a value that is off of the screen.

All missiles can be combined into a fifth player by setting bit D4 of the priority control register to a one. The fifth player gets its color and luminance from playfield color/lum register 3. This means that if the fifth player is positioned over a playfield 3 pixel, the player and playfield pixels will merge together and portions or all of the player will appear to disappear. The fifth player is moved horizontally on the screen by changing the contents of all four missile horizontal position registers to the same value.

The CGIA has the ability to display players and missiles that overlap on the screen with a third color in the overlapping region. Multiple color players and missiles are enabled by setting bit D5 in the priority control register to a one. Any time a player/missile 0 pixel overlaps a player/missile 1 pixel, the pixel color and luminance will be the result of the logical OR of the corresponding color/lum registers. Likewise, any overlapping player/missile 2 and player/missile 3 pixels will have their color/lum register bits logically ORed.

3.11) Creating Players and Missiles by Using DMA

Players and missiles can be created by using direct memory access. This allows player/missile graphics for every TV scan line to be stored in system memory and loaded into the player/missile graphics registers automatically. In order to use player/missile DMA, the graphics data must first be stored in memory. The player/missile base address register (PMBASE) is used to specify the most significant 5 bits (A_{15}-A_{11}), when using one line player/missile resolution, and the most significant 6 bits (A_{15}-A_{10}),when using two line player/missile resolution, of the address where the player/missile graphics data is stored in memory. This means that player/missile graphics data must start at an acceptable memory boundary. The remaining 10 or 11 bits (A_{10},A_9-A_0) are automatically calculated by the CGIA. Due to the nature

of this calculation, an offset of either 100 (HEX) for one line resolution, or 80 (HEX), for two line resolution, must added to PMBASE to get the starting address of the player/missile graphics data. This 256 or 128 bytes of memory between the player/missile base address and the start of actual graphics data is not used by the CGIA and can be used for other purposes. Two line resolution means that the same graphics data is used for two TV scan lines. 640 (decimal) bytes (5 X 128) are required for two line resolution and 1280 bytes (5 X 256) bytes are requires for one line resolution. Figure 9 illustrates how the player/missile graphics addresses are calculated.

Once the player/missile graphics data has been stored in memory, player/missile DMA has to be enabled. This is accomplished by setting bits D3 and D2 of the DMACTL register and bits D1 and D0 of the GRACTL register to ones. When using player/missile DMA, each player or missile can be delayed by one TV scan line. The VDELAY register is used to give one line resolution in the vertical positioning of a player or missile when two line resolution is used. Setting a bit in the VDELAY register to a one will move the corresponding player or missile down by one TV scan line.

Player-Missile Base Address (PMBASE) = MSB of address.
Resolution is controlled by bit 4 of DMACTL.

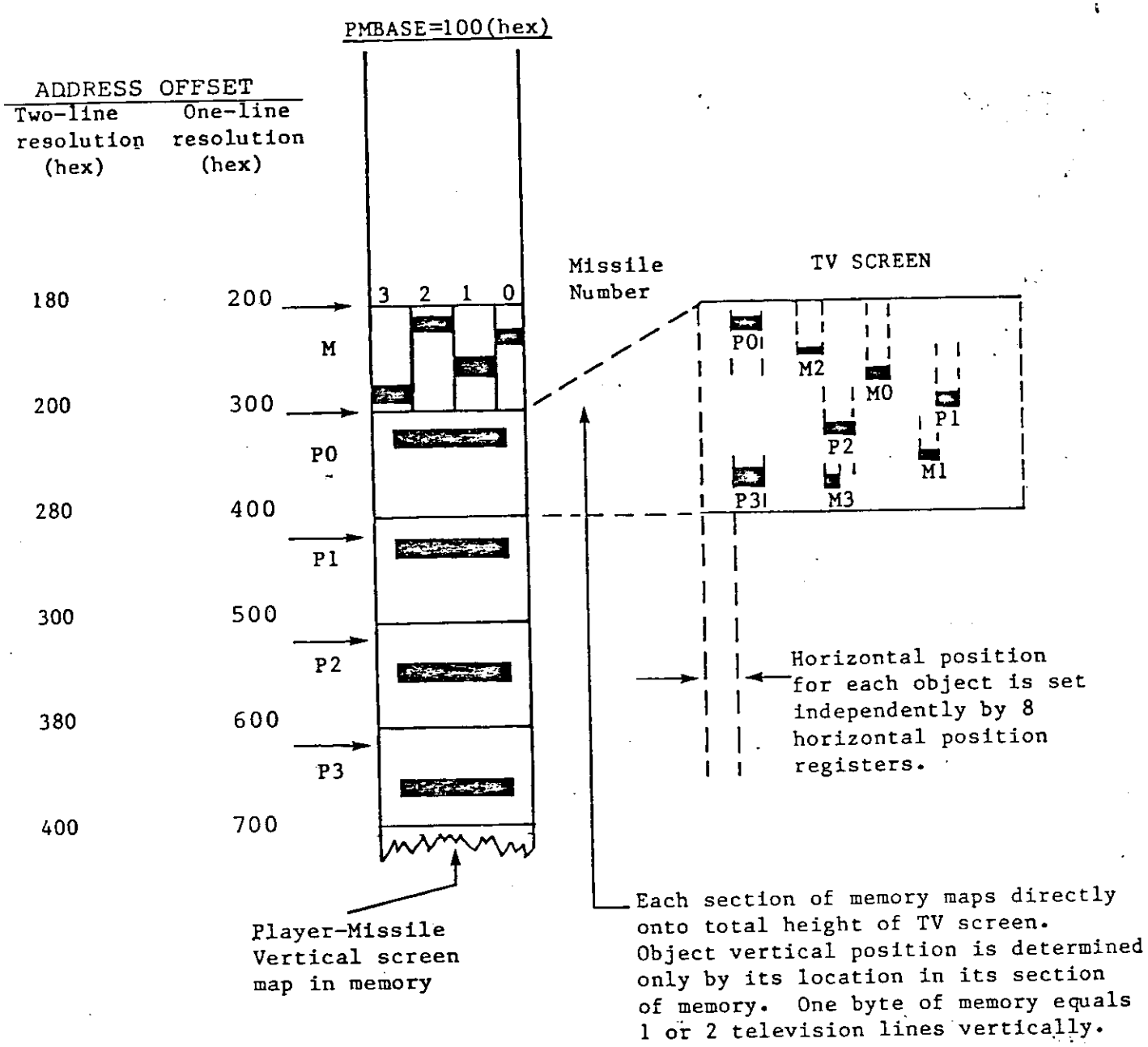


Figure 9) Player/Missile Graphics Data Addressing

4.0) CGIA Direct Memory Access

The CGIA uses direct memory access to accomplish several different functions. The primary function is to fetch display instructions and playfield graphics data from memory for display. The secondary function is to fetch player/missile graphics data for display. All aspects of DMA are controlled by the DMA control register. The DMA control register can enable/disable display instruction fetches, as well as, playfield graphics data fetches, select one of the three playfield display widths, enable/disable player/missile graphics data fetches, and select between one or two line player/missile vertical resolution.

The DMA process is accomplished by suspending all microprocessor operations, accessing system memory, reading data stored in memory, and returning control of the system back to the microprocessor. The HALT output of the CGIA is used to suspend the ϕ0 clock of the microprocessor.

4.1) Display Instruction DMA

DMACTL register bit D5 is used to enable display list instruction fetches. If bit D5 is equal to zero, display instructions will not be fetched and the TV screen will display background color. If bit D5 is equal to one, display instructions will be fetched from memory during the horizontal blank period, just before the start of the display line. The display instruction tells the CGIA how to DMA the graphics data if playfield DMA is enabled (DMACTL bits D1, and D0 not equal to zero).

4.2) Playfield DMA

DMACTL register bits D1 and D0 are used to enable playfield graphics DMA and to select one of three playfield display widths. If DMACTL D1,D0 = 0,0, the CGIA will not fetch playfield data for display. In this case, the playfield will be displayed as background color. If DMACTL D1,D0 = 0,1, playfield graphics data will be fetched for a narrow playfield display. The narrow playfield is only 128 color clocks wide. The remaining area on the screen will be filled in using background color. If DMACTL D1,D0 = 1,0, playfield data will be fetched for a standard playfield display. The standard playfield is 160 color clocks wide. This playfield width pretty much covers the whole screen. Any open areas on the border are filled in with background color. If DMACTL D1,D0 = 1,1, playfield graphics data will be fetched for a wide playfield display. The wide playfield is 192 color clocks wide. Not all of the data that is fetched for a wide playfield can be displayed on the TV screen. There are a total of 228 color clocks per TV scan line. Fourty clocks are used for horizontal blank. Ten clocks are used to display background at the start of the line. This leaves 178 clocks for playfield data to be displayed.

4.2) Playfield DMA (cont'd)

During the last two clock positions, the CGIA attempts to display some of the leftover data. This data is different from frame to frame which appears on the screen (if it could be seen) as garbage. This leaves a total of 176 color clock positions for a wide playfield display. Narrow playfields require fewer bytes of DMA memory, therefore, the number of microprocessor instruction cycles stolen are reduced.

4.3) Player/Missile DMA

Player and missile graphics data can be fetched from memory by setting DMACTL bits D3 and D2 to ones. If player/missile DMA is enabled, the CGIA will fetch the graphics data during every horizontal blank period for display on the next TV scan line. The advantage of player/missile graphics is that they can be created through software or DMA techniques. Missile graphics data is always DMAed before player graphics. If only player DMA is enabled, the CGIA will also fetch the missile graphics data, due to the nature of the DMA circuitry. Once player DMA is disabled, missile data can only be DMAed by setting the missile DMA enable bit. Figure 10 shows the different sources for generating player/missile and playfield graphics data.

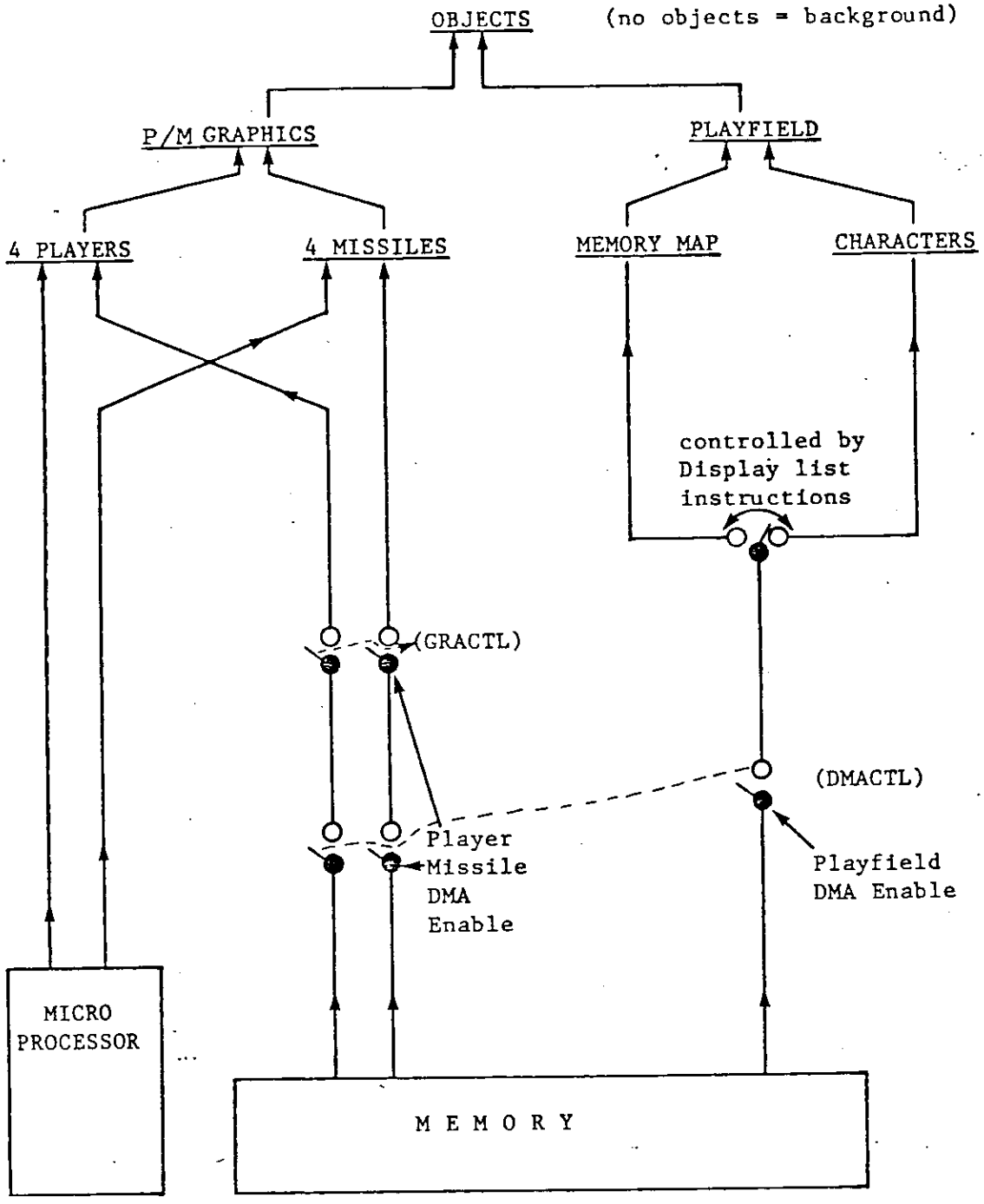


Figure 10) Graphics DMA Sources

4.4) DMA Cycle Counting

It may be necessary, at times, to keep track of the number of DMA cycles generated by the CGIA. The 6502 microprocessor machine cycle is equal to two CGIA color clocks. This averages out to be 29,868 microprocessor machine cycles per television field. Since all microprocessor functions are suspended during the DMA cycle, the total number of microprocessor instruction cycles per field are reduced. The general rule of thumb is that for every byte of information DMAed there will be one less microprocessor cycle available for program execution. The only exception to this rule is RAM refresh. RAM refresh steals 9 machine cycles per horizontal scan line. Every RAM refresh request, however, is not answered. Character graphics modes 2,3,4 and 5 can override a RAM refresh request. The following information can be used to calculate the number of DMA cycles per field.

DMA Type	No. of cycles	Frequency
RAM Refresh	9	Per Horizontal Scan Line
Display List Instructions	1	Per Opcode/Operand Byte
Graphics Modes 2,3,4 and 5	+288	Per Narrow Width Mode Line
	*360	Per Standard Width Mode Line
	*432	Per Wide Playfield Mode Line
Graphics Modes 6 and 7	144	Per Narrow Width Mode Line
	180	Per Standard Width Mode Line
	216	Per Wide Playfield Mode Line
Graphics Modes 8 and 9	8	Per Narrow Width Mode Line
	10	Per Standard Width Mode Line
	12	Per Wide Playfield Mode Line
Graphics Modes A,B and C	16	Per Narrow Width Mode Line
	20	Per Standard Width Mode Line
	24	Per Wide Playfield Mode Line
Graphics Modes D,E and F	32	Per Narrow Width Mode Line
	40	Per Standard Width Mode Line
	48	Per Wide Playfield Mode Line
Missile Graphics	226	Per Field (1 line resolution)
	113	Per Field (2 line resolution)
Player Graphics#	904	Per Field (1 line resolution)
	452	Per Field (2 line resolution)

+ * Subtract 7 refresh cycles for each mode line used.
* Subtract 8 refresh cycles for each mode line used.
# Missile graphics DMA cycles must also be added, even if missile DMA is not enabled.

5.0) RAM Refresh

RAM refresh can be considered a form of direct memory access. The reason for this is because microprocessor operation is suspended during the refresh process. The primary difference between RAM refresh and graphics DMA is that RAM refresh can not be disabled. The CGIA provides circuitry on chip which will generate RAM refresh address for dynamic RAMs. The refresh cycle provided is equal to 256 refresh row addresses once every 2 mS. This is equivalent to nine refresh cycles per horizontal scan line. When a RAM refresh request is generated inside the CGIA, the $\overline{\text{HALT}}$ line goes low to freeze the microprocessor which allows the CGIA to take control of the address and data bus as well as the R/$\overline{\text{W}}$ line. One color clock later, the $\overline{\text{REF}}$ line goes low to indicate that the address on the data bus is a valid RAM refresh address. The R/$\overline{\text{W}}$ line is always high as an output. The refresh row address is output on A_7-A_0 with A_7 being the most significant bit of the row address.

This page intentionally left blank

6.0) Video Generation

The CGIA is designed to generate color and luminance, as well as, horizontal and vertical sync signals for an NTSC video standards television system. The following sections simply describe how these signals are generated.

6.1) Monochrome Signal

All televisions, regardless of the video standard used, create a picture on the TV screen in much the same way. The color video signal is composed of two primary analog signals: monochrome (luminance) and chroma. The most important portion of the video signal is the monochrome or luminance signal. This signal is further subdivided into the active display and the horizontal and vertical blank periods. The luminance signal for the active display portion of the display is derived from the luminance bits D3-D1 of the CGIA color/lum registers (display data in GTIA mode 1). The luminance signal is responsible for determining the brightness of the pixels or dots on the TV screen. The higher the binary luminance value, the brighter the dot on the screen. A binary value of all zeros will cause the electron beam to be turned off. This condition is more commonly called a "blank" or "blacker than black" level. A binary value of all ones will cause the electron beam to glow at its brightest. This condition is more commonly called a "white level." As the electron beam sweeps across the screen, the luminance level is constantly changing to reflect the luminance values of the graphics data that is stored in memory.

Horizontal Blank and Horizontal Sync--In order to create a television picture by the scanning method described in section 1.0, the electron beam must be returned from the right edge of the screen to the left edge of the screen in a periodic fashion. The horizontal sync signal causes the horizontal deflection coils to return the electron beam to the left edge of the screen. The horizontal sync pulse is generated in a section of the video signal known as "horizontal blank." The horizontal blank period is divided into three parts, front porch, sync, and the back porch or horizontal retrace. The front porch is the period where the electron beam is first turned off before the sync pulse is generated. This ensures that the beam will not be seen as it is moved back across the screen. The sync period is the time during which the horizontal sync output level is generated. The sync voltage level causes the horizontal deflection coils to reposition the electron beam at the left edge of the screen. The back porch is the period during which the electron beam is actually being moved from the right edge of the screen to the left edge.

6.1) Monochrome Signal (cont'd)

The process of horizontal blank takes exactly 40 color clocks on the CGIA. The time period is broken down as follows:

Front Porch = 6 color clocks

Horizontal Sync = 16 color clocks

Back Porch = 18 color clocks

Horizontal Blank Total = 40 color clocks or
= 11.16 uS

Vertical Blank and Vertical Sync--The vertical blank period is the time during which the electron beam is returned from the bottom-right of the screen to the top-left. As with horizontal blank, the vertical blank period is also divided into three parts, blank, sync, and vertical retrace. The blank period is the time during which the electron beam is first turned off before the vertical sync pulse is generated. This ensures that the beam will not be seen as it is move to the top of the screen. The sync period is the time during which the vertical sync output levels are generated. The sync pulse consists of three horizontal scan lines where the luminance levels are inverted for a normal blank display line. The sync pulse causes the vertical and horizontal deflection coils to return the electron beam to the top-left corner of the screen. The vertical retrace period is the time during which the electron beam is actually being moved from the bottom-right to the top-left of the screen.

The CGIA incorporates the video digital to analog converter on-chip for cost reduction purposes. The sync and four luminance bits are converted into an analog output signal (CLUM pin) by means of a resistor-divider network. Each lum and sync bit is weighted according to the desired output voltage. Refer to the D.C. Operating Characteristics for more information on the CLUM output levels.

6.2) Chrominance Signal

In order to add color to objects displayed on the screen, a color signal must be added to the monochrome video output signal. The color signal is a phase/amplitude modulated signal. This means that colors are generated by outputting a signal that is different in phase and amplitude from a reference signal. This reference is generally referred to as the "color burst" signal. The color burst signal on the CGIA has a phase shift of 0° and zero luminance. A color television picture is composed of three primary colors, red, green, and blue. These three colors are mixed or blended together to provide a variety of colors. The CGIA has the capability to display sixteen different colors or hues (including black), with each color having sixteen different saturation or luminance values. Color saturation is taken from the monochrome signal. Color phase shifts are generated inside the CGIA by passing the color signal (COL pin) through a series of delay elements. There are fifteen delay elements with

6.2) Chrominance Signal (cont'd)

each element providing the same amount of delay or phase shift. The color register value selects one of the 15 delay taps in order to create the desired phase shift. Each color is different in phase from the color burst by 24° (or 18.6 nS when using a 3.579 MHz color burst frequency). The gray levels (16th color) is merely the absence of the color output frequency. The color burst signal generated by the CGIA consists of 12 color clock cycles of output on the COL line during the back porch of horizontal blank. This output is also equal to color register value one. As long as the color being displayed is the same across the TV line, the duty cycle of the COL output frequency is equal to 50%. Any time the display goes from one color to another, the duty cycle of the COL output frequency will change from 50% to some other value for that color clock. This duty cycle difference depends on the color output value. This duty cycle change is interpreted by the television as a double change in phase angle and will cause an intermediate color to be displayed between different color pixels. This phenomenon is referred to as "color artifacting." Color artifacting can also be created by drastic changes in luminance. The phase angle shifts for the primary colors are listed below:

RED	76.5°
BLUE	192.0°
GREEN	299.9°

The phase angle shifts for the colors generated by the CGIA are listed below:

Color Register Value (HEX)	Output Color	Phase Angle (degrees)
0	Grey (no color)	No Output
1	Gold	0 (reference)
2	Orange	24
3	Red-Orange	48
4	Pink	72
5	Purple	96
6	Purple-Blue	120
7	Blue	144
8	Blue	168
9	Light-Blue	192
A	Turquoise	216
B	Green-Blue	240
C	Green	264
D	Yellow-Green	288
E	Orange-Green	312
F	Light-Orange	336

Figure 11 is a phase angle vector plot of the COL output for different color register values. In order to create a composite video signal, the CLUM and COL output signals must be added or combined together. This composite signal must then be RF modulated for display on the television.

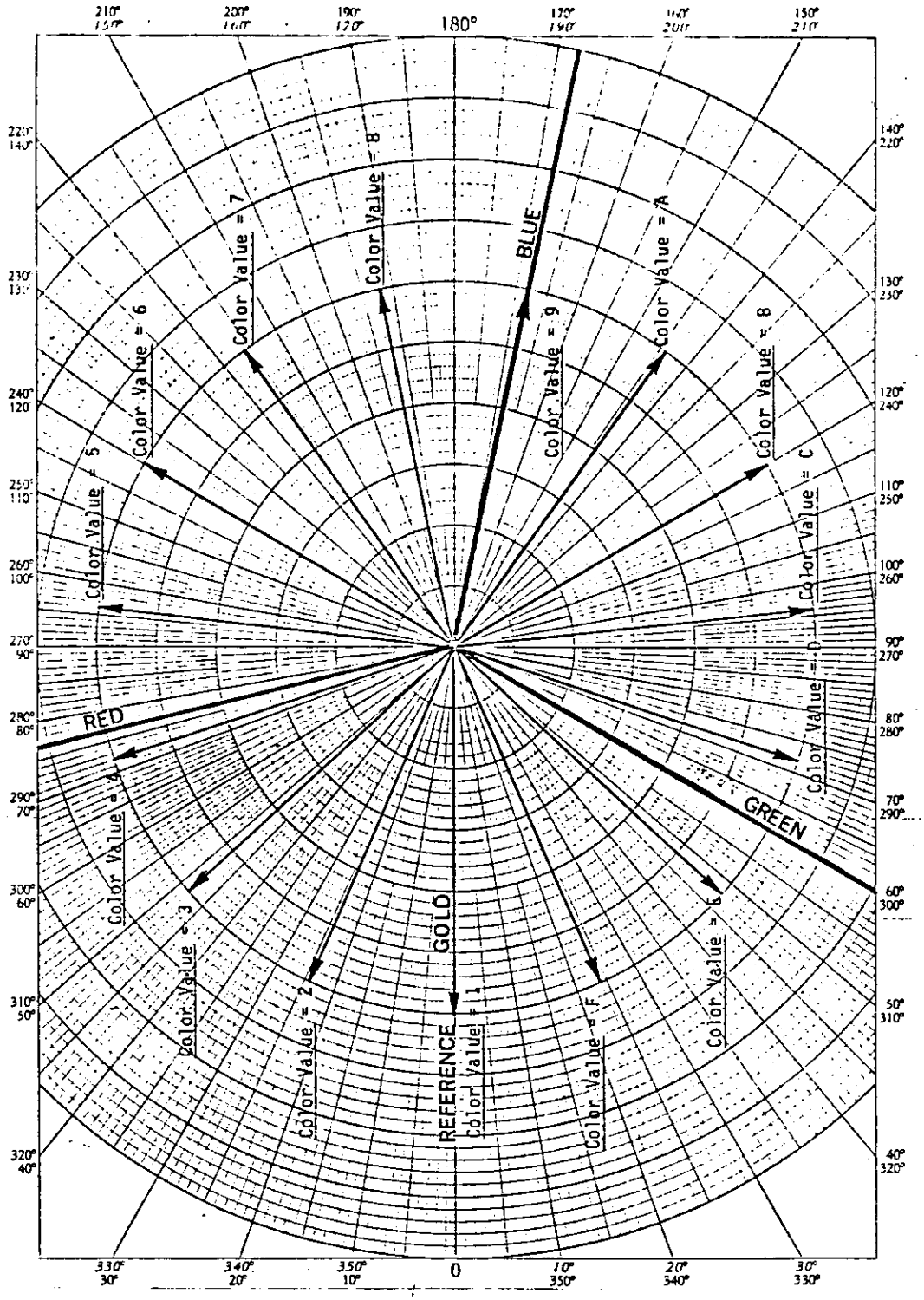


Figure 11) Phase Angle Plot of Color Output Values

7.0) General Purpose I/O Functions

7.1) Trigger Input Port (T0-T3)

The trigger input port is used to interface to the joystick trigger buttons. The trigger input data can be read at any time just by reading the TRIG0-TRIG3 input registers. These registers are only one-bit wide and show the current status of the corresponding trigger input. The trigger button is considered to have been pressed if the input data is equal to a zero. The trigger inputs can be latched by setting bit D2 in the GRACTL register. When this bit is set to a one, all trigger inputs are latched when they make a transition from high to low. This information remains latched in the trigger input registers until GRACTL bit D2 is reset.

7.2) Switch I/O Port (S0-S3)

The Switch I/O lines are used for general purpose I/O functions. The S_0-S_3 outputs are open-drain outputs and the inputs have internal pull-up resistors to V_{CC}. Figure 12 is a close schematical representation of the S_0-S_3 I/O port. The output data is latched in the CGIA by writing to the CONSOL write register. The output of the data latch controls the switch line output transistor. If the output data is a one, the output transistor is turned on and the switch line is grounded. If the output data is a zero, the output transistor is turned off and the switch line is internally pulled-up to V_{CC}. It is important to remember that the data on the switch lines is always the inverse of what is written to the CONSOL register.

NOTE: READ ADDRESSES CX16-CX18 ARE USED FOR TEST PURPOSES ONLY AND SHOULD NOT BE READ. READING THESE ADDRESSES CAUSES TEST DATA TO BE OUTPUT ON THE S_0-S_3 LINES. WRITING TO THE SWITCH OUTPUT PORT CAUSES THE SWITCH LINES TO ASSUME NORMAL OPERATION.

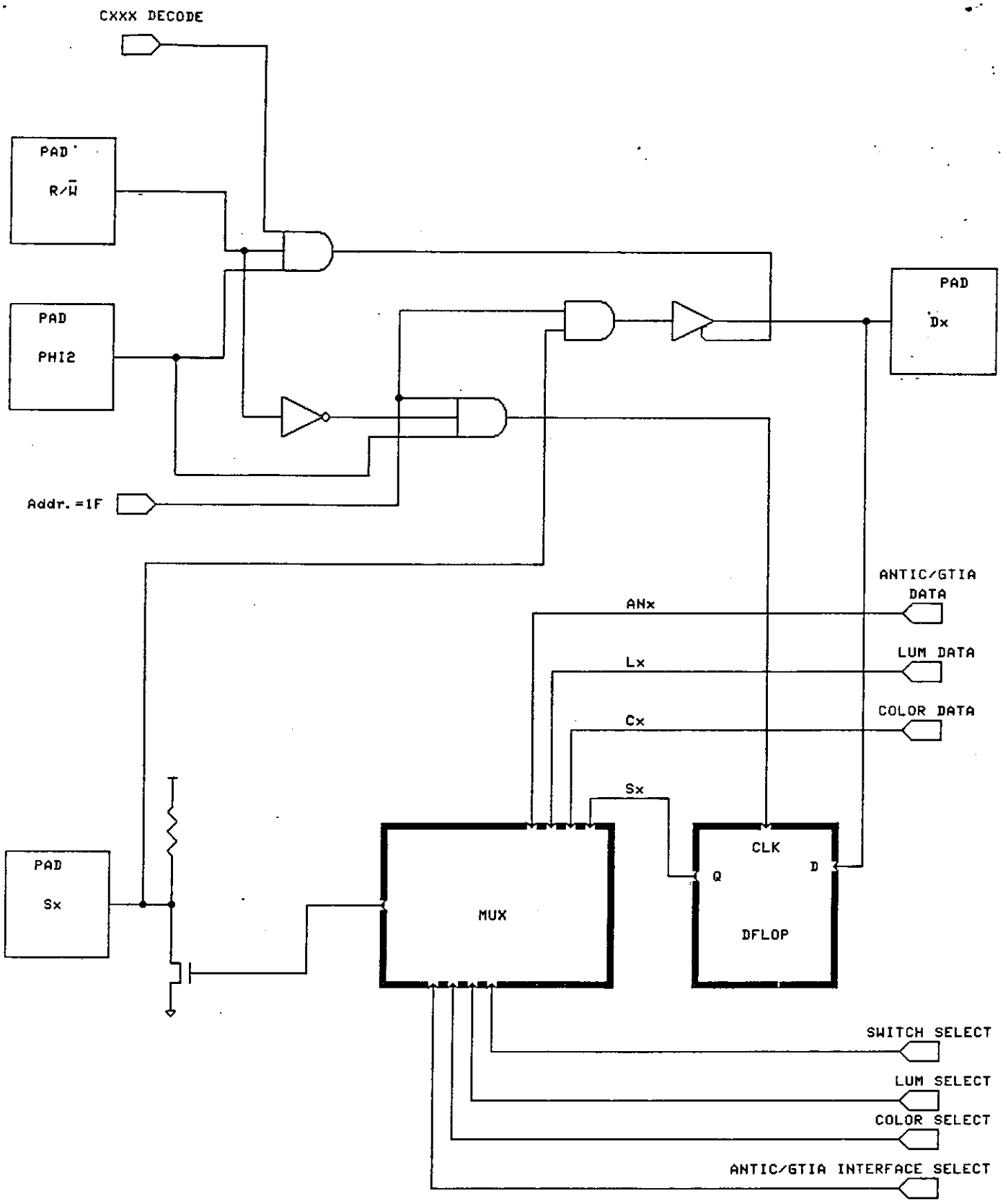


Figure 12) Schematical Representation of Switch I/O Port

SPECIAL FUNCTIONS

8.1) Object Priority Control

The CGIA controls all aspects of the playfield and player/missile graphics. Playfield and player/missile graphics are created separately and must be combined together for display on the TV screen. This is accomplished by establishing priorities between the different graphics objects. The CGIA accomplishes this by using the information in the priority control register. Bits D3-D0 are used to assign priorities to all graphics objects. The use of the priority control bits is a non-exclusive function. This means that no two priority bits (D3-D0) should be equal to one. If this happens object priorities will be in conflict and the overlapping pixels will be blanked. There are four different sets of object priorities. When the fifth player is used, the playfield 3 color/lum register is used to give the player its color. The fifth player will not have priority over playfield 3 objects. This means that if the fifth player is used and is positioned over a playfield 3 object, the overlapping pixels will blend together. For more information on the use of the priority bits, refer to the register description section of this specification.

8.2) Object Collision Detection

In order to determine if there has been a collision (overlap) between objects, the CGIA provides 60 bits of data to detect and store overlap conditions between players, missiles and playfield. These bits can be read at any time. Collision detection bits are provided for player to player collisions, player to missile collisions, player to playfield collisions and missile to playfield collisions. There is, however, no provision for missile to missile collisions or playfield to playfield collision since this condition can not exist (except when using 5^{th} player, in which case 5^{th} player collisions with playfield are not recorded)

In character graphics modes 2 and 3 and bit map graphics mode F, the pixel colors are generated by using the playfield 2 color/lum register and the luminance bits (D3-D1) of the playfield 1 color/lum register. In this case, object collisions are read as a playfield 2 collision.

8.3) Determining Vertical Position

The current TV scan line being displayed can be determined by reading the contents of the vertical position counter (VCOUNT). The vertical position counter is a nine bit counter, but only the most significant 8 bits can be read. Reading the VCOUNT register is only able to provide two line vertical resolution. The vertical counter counts up from decimal 0 to 130 with the 0 point near the end of vertical blank.

8.4) Determining The CGIA Video Standard

The PAL register is used to determine which version of the CGIA is being used. If the PAL register value is equal to 0F (HEX), the NTSC video version of the CGIA is being used. If the PAL register value is equal to 01 (HEX), the PAL video version of the CGIA is being used. The PAL register is provided so that software can make the necessary program adjustments.

8.5) Wait For Horizontal SYNC

The CGIA provides a strobe register (WSYNC) which will cause the RDY input to the microprocessor to go low causing the microprocessor to halt program execution until horizontal blank occurs. This function is very useful because it allows the programmer to synchronize the software with the CGIA. The wait for sync function can be used to change playfield or player/missile graphics as well as the color registers. The RDY line goes back high 10 color clocks or 5 microprocessor instruction cycles before the occurrence of horizontal blank.

8.6) Non-Maskable Interrupts

There are three sources of non-maskable interrupts on the CGIA. These three sources are: the occurrence of vertical blank, display list interrupt, and the $\overline{\text{RNMI}}$ input going low. The vertical blank and display list interrupts can be disabled (masked out) by setting bits D7 and D6 in the NMIEN register. The $\overline{\text{RNMI}}$ interrupt can not be disabled. The RNMI interrupt is useful since it will always be answered. This allows a program to be restarted if for some reason there is a glitch in the system and the program gets lost. The vertical blank interrupt is useful for synchronizing the software to the CGIA. The vertical blank period is a good time to change the display list or any of the CGIA's graphics or color registers. The display list interrupt is also useful for this same purpose. The cause of the interrupt can be determined by reading bits in the NMIST register. The source of the interrupt can be cleared by writing to the NMIRES register (reset NMIST bits). Even though non-maskable interrupts can be disabled on the CGIA, once they are generated, they will always be answered by the microprocessor.

8.7) Using An External Light Pen

A light pen is a device that can detect the electron beam as it sweeps across the TV screen. It is used to point directly at an image on the television display. Applications include selecting menu items and drawing lines.

When the $\overline{\text{LP}}$ input to the CGIA is pulled low, the CGIA takes the current VCOUNT value and stores it in PENV. The least horizontal color clock value (0-227 decimal) is stored in PENH. The least significant bit is inaccurate and should be ignored. Since there are a number of delays involved in displaying the data and changing the light pen registers. Each system must be calibrated. Software which uses the light pen should contain a user-interactive calibration routine. For example, the user could point the light pen at a cross-hair in

8.7) Using An External Light Pen (cont'd)

the center of the screen and the program could compute the required horizontal off set. PENH will wrap around from 227 to 0 near the right hand edge of a standard width display because of the delay. The pen will not work if it is pointed at a black area of the screen, since the electron beam is turned off. It is a good idea to read two (or more) values and average them, since the user will probably not hold the pen perfectly steady.

8.8) Special Test Functions

Due to the complexity of the CGIA design, special test functions were added in order to make LSI testing easier. These functions are enabled by executing a read operation at addresses CX16, CX17, and CX18. When one of these addresses is read, internal control signals are output on the switch I/O port. It is VERY IMPORTANT that these addresses are not read at any time during normal operation. Doing so could result in program failure. The switch lines are returned to their normal functions any time the switch I/O register is written to. For more information on the special test functions, refer to test suppliment TS20577.

9.0) Write-Only Registers

9.1) Graphics DMA Control

DMACTL (Playfield Graphics DMA Control) Address = D400

Not Used	D5	D4	D3	D2	D1	D0

D5=1 Enable display list instruction DMA
D5=0 Disable display list instruction DMA (power-up state)

D4=1 One line player/missile resolution
D4=0 Two line player/missile resolution (power-up state)

D3=1 Enable player graphics DMA
D3=0 Disable player graphics DMA (power-up state)

D2=1 Enable missile graphics DMA
D2=0 Disable missile graphics DMA (power-up state)

D1,D0=00 No playfield graphics DMA
D1,D0=01 Narrow playfield graphics DMA
D1,D0=10 Standard playfield graphics DMA
D1,D0=11 Wide playfield graphics DMA

The DMA control register is used to enable or disable graphics DMA. Bit D5 is used to enable/disable display list instruction fetches. If this bit is cleared, display list instructions will not be fetched and the output on the television screen will be background color and luminance. DMACTL bit D5 is cleared when the CGIA is reset and can be enabled once the display list has been created.

DMACTL bit D4 is used to select between one line or two line player/missile vertical resolution. If DMACTL bit D4 is cleared, graphics data for all players and missiles will be repeated on the next scan line, hence two line resolution. Using two line resolution requires less memory to define a player or missile, however the vertical resolution is not as good. If DMACTL bit D4 is set, graphics data for all players and missiles is used for a single line only. One line resolution requires twice as much memory as two line resolution but players and missiles have twice the vertical resolution.

DMACTL register bit D3 is used to enable/disable player graphics data DMA. If this bit is cleared, the CGIA will not fetch player graphics information from system memory. If this bit is set, player graphics information will be fetched from memory. GRACTL register bit D1 should also be set for player graphics DMA.

9.1) Graphics DMA Control (cont'd)

DMACTL register bit D2 is used to enable/disable missile graphics data DMA. If this bit is cleared, the CGIA will not fetch missile graphics information from system memory. If this bit is set, missile graphics information will be fetched from memory. Missile graphics data will be fetched from memory if player DMA is enabled regardless of whether or not missile DMA is enabled. Once player DMA is disabled missile data can only be enabled by setting DMACTL bit D2. GRACTL register bit D0 should also be set for missile graphics DMA.

DMACTL register bits D1 and D0 are used to select the width of the playfield. If bits D1 and D0 are both zero, the CGIA will not fetch playfield graphics information from system memory. If bits D1=0 and D0=1, the CGIA will DMA graphics data for a narrow playfield (128 color clocks wide). If D1=1 and D0=0, the CGIA will DMA data for a standard playfield (160 color clocks wide). If bits D1=1 and D0=1, the CGIA will DMA data for a wide playfield (192 clocks wide). Increasing playfield widths requires more system memory to store that data as well as an increased number of processor machine cycles stolen in order to access that data.

9.2) Character Display Control

CHACTL (Character Display Control) Address = D401

Not Used	D2	D1	D0

D2=1 Enable vertical reflect
D2=0 Disable vertical reflect

D1=1 Enable inverse video
D1=0 Disable inverse video

D0=1 Enable character blank
D0=0 Disable character blank

The character control register controls the way in which characters are displayed on the screen. CHACTL controls such aspects as vertical reflect, inverse video, and character blank.

Bit D2 of CHACTL is used to enable/disable the vertical reflection of a character. At the beginning of each character mode line, this bit is tested and if true causes the line of characters to be reflected (displayed upside down).

Bit D1 of CHACTL is used to enable/disable inverse video. The inverse video mode can only be used in character modes 2 and 3. If inverse video is enabled, all characters with character name bit D7 equal to one will be displayed in inverse video (off pixel on and on pixel off).

9.2) Character Display Control (cont'd)

Bit D0 of CHACTL is used to enable/disable character blank. The character blank mode can only be used in character modes 2 and 3. If character blank is enabled, all characters with character name bit D7 equal to one will be blanked (off pixel luminance displayed). If both inverse video and character blank are enabled, the character will appear as an inverse video blank character (solid square).

9.3) Display List Pointer

DLISTL (Display List Pointer-Low Byte) Address = D402
DLISTH (Display List Pointer-High Byte) Address = D403

D7	D6	D5	D4	D3	D2	D1	D0

The display list pointer is used to point to the next display instruction. The display list pointer is a sixteen bit pointer and is initialized by writing to DLISTH and DLISTL. The least significant ten bits (D0-D9) are configured as a counter and automatically incremented in order to fetch the next display instruction. The most significant six bits (D9-D15) are fixed and can only be changed by writing to DLISTH or by using a jump instruction. This also means that display lists can not cross a 1K Byte memory boundary unless a jump instruction is used.

9.4) Horizontal Scrolling

HSCROL (Horizontal Scroll Count) Address = D404

Not Used	D3	D2	D1	D0

D3-D0 = 0 to 15 color clock shifts
to the right of the screen

The HSCROL register controls the number of color clock positions to the right of the screen a graphics element should be shifted if horizontal scrolling is enabled. Horizontal scrolling is enabled by setting bit D4 of the display instruction to a one. When horizontal scrolling is enabled, more bytes of display data per line are needed. For a narrow playfield there should be the same number of bytes per mode line as for a standard playfield without scrolling. Similarly, the same number of bytes are required for a standard playfield with scrolling as for a wide playfield and no scrolling. For wide playfield, there is no change in the number of bytes per mode line and background color is shifted into the vacant locations.

9.5) Vertical Scrolling

VSCROL (Vertical Scroll Count) Address = D405

Not Used	D2	D1	D0

8,4,2,1 Line Display Modes

D2-D0 = 0 to 7 scan line shifts
to the top of the screen

Not Used	D3	D2	D1	D0

16,10 Line Display Modes

D3-D0 = 0 to 15 scan line shifts
to the top of the screen

The VSCROL register controls the number of television scan lines that a display block should be shifted to the top of the screen if vertical scrolling is enabled. Vertical scrolling is enabled by setting bit D5 of the display instruction to a one. The scrolled area will terminate with the first display instruction having a zero in bit D5.

9.6) Player/Missile Base Address

PMBASE (Player/Missile Base Address) Address = D407

D7	D6	D5	D4	D3	D2	Not Used

One Line Resolution

D7	D6	D5	D4	D3	D2	D1	Not Used

Two Line Resolution

The PMBASE register specifies the most significant 5 or 6 bits of the address of the player/missile graphics data that is stored in system memory. The starting location of the graphics data is determined by adding an offset to PMBASE. The offset is 128 bytes for two line resolution and 256 bytes for one line resolution. Player/missile graphics data can not cross a 1K byte address boundary.

9.7) Character Base Address

CHBASE (Character Base Address) Address = D409

D7	D6	D5	D4	D3	D2	Not Used

40 Character Modes

D7	D6	D5	D4	D3	D2	D1	Not Used

20 Character Modes

The CHBASE register specifies the most significant 6 or 7 bits of the address where the character graphics data (stamp) is stored. D0 and D1 are not used and are assumed to be zero.

9.8) Wait For Horizontal Blank Synchronization

WSYNC (Wait for Horizontal Blank) Address = D40A

Not Used

The WSYNC register is not actually a register but instead a control signal strobe. By writing to the WSYNC location, the CGIA will generate a logic low condition on the RDY output. This causes the microprocessor to finish executing the current instruction and then halt program execution until the RDY line returns to the logic one state. The CGIA will change the RDY output from low to high 10 color clocks (5 instruction cycles) before horizontal blank occurs.

9.9) Enable Non-Maskable Interrupts

NMIEN (NMI Interrupt Enables) Address = D40E

D7	D6	Not Used

D7=0 Disable display list instruction interrupts (power-up state)
D7=1 Enable display list instruction interrupts

D6=0 Disable vertical blank interrupts (power-up state)
D6=1 Enable vertical blank interrupts

9.9) Enable Non-Maskable Interrupts (cont'd)

Even though NMI interrupts are "unmaskable" on the microprocessor, the CGIA has interrupt enable (mask) bits for the NMI functions. The NMIEN register is used to enable/disable non-maskable functions. The NMIEN register is used to enable/disable non-maskable interrupts. When bits D7 and D6 of NMIEN are zero, NMI interrupts are disabled (masked) and prevented from causing a microprocessor NMI interrupt. There are three sources for non-maskable interrupts. Setting bit D7 of any display instruction causes a non-maskable interrupt to be generated when that mode line is displayed. NMIEN bit D7 is used to enable and disable display list interrupts. Non-maskable interrupts are generated on every occurrence of vertical blank. NMIEN bit D6 is used to enable or disable these interrupts. The third non-maskable interrupt is generated when the $\overline{\text{RNMI}}$ input makes a transition from high to low. The $\overline{\text{RNMI}}$ interrupt can not be disabled, therefore there is not an enable bit for $\overline{\text{RNMI}}$ in the NMIEN register. The NMIEN register is cleared on reset. Even though non-maskable interrupts are considered to be non-maskable, once enabled they are always answered by the microprocessor.

9.10) Reset Non-Maskable Interrupt Status Register

NMIRES (NMI Status Register Reset) Address = D40F

Not Used

The NMIRES register is not actually a register but instead a reset strobe. By writing to the NMIRES location, the NMI Status Register (NMIST, Addr.=D40F, read) will be reset. This causes all non-maskable interrupt conditions to be cleared. NMIRES should be written to after reading the NMI status register in order to clear the source of the interrupt.

9.11) Player/Missile Horizontal Position

PLAYER HORIZONTAL POSITION:

HPOSP0 (Player 0 position) Addr. = C000
HPOSP1 (Player 1 position) Addr. = C001
HPOSP2 (Player 2 position) Addr. = C002
HPOSP3 (Player 3 position) Addr. = C003

9.11) Player/Missile Horizontal Position (cont'd)

MISSILE HORIZONTAL POSITION:

HPOSM0 (Missile 0 position) Addr. = C004
HPOSM1 (Missile 1 position) Addr. = C005
HPOSP2 (Missile 2 position) Addr. = C006
HPOSM3 (Missile 3 position) Addr. = C007

D7	D6	D5	D4	D3	D2	D1	D0

Players and missiles are small objects which can be moved in the horizontal direction by changing their position registers. The horizontal position value determines the color clock location of the left edge of the object. Hex 30 is the left edge of a standard width screen. Hex D0 is the right edge of a standard screen.

There are a total of four players and four missiles. The four missiles may be combined together and used as a 5th player. The horizontal position registers may be reloaded at any time by the processor, allowing an object to be replicated many times across a horizontal scan line.

9.12) Player/Missile Size

PLAYER SIZE:

SIZEP0 (Player 0 size) Addr. = C008
SIZEP1 (Player 1 size) Addr. = C009
SIZEP2 (Player 2 size) Addr. = C00A
SIZEP3 (Player 3 size) Addr. = C00B

Not Used	D1	D0

0 0 = Normal Size
(8 color clocks wide)

0 1 = Twice Normal Size
(16 color clocks wide)

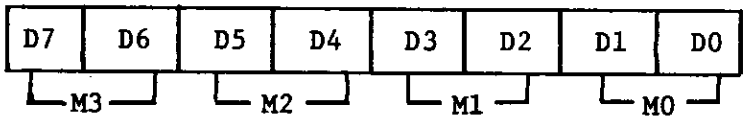
1 0 = Normal Size

1 1 = Four Times Normal Size
(32 color clocks wide)

9.12) Player/Missile Size (cont'd)

MISSILE SIZE:

SIZEM (All missile sizes) Addr. = C00C



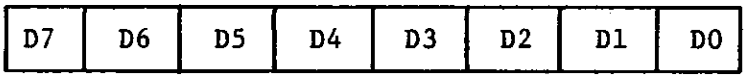
0	0 = Normal Size (2 color clocks wide)
0	1 = Twice Normal Size (4 color clocks wide)
1	0 = Normal Size (2 color clocks wide)
1	1 = Four Times Normal Size (8 color clocks wide)

Each player and missile can be displayed in three different sizes. There is normal, two times normal, and four times normal. Normal size is one color clock per bit in the graphics register.

9.13) Player/Missile Graphics

PLAYER GRAPHICS:

GRAFP0 (Player 0 graphics) Addr. = C00D
GRAFP1 (Player 1 graphics) Addr. = C00E
GRAFP2 (Player 2 graphics) Addr. = C00F
GRAFP3 (Player 3 graphics) Addr. = C010

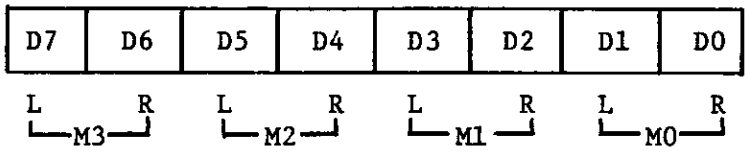


Left Right

Player on T.V. Screen

MISSILE GRAPHICS:

GRAFM (All missile graphics) Addr. = C011



9.13) Player/Missile Graphics (cont'd)

The shape of a player or missile is determined by the data in the graphics register. The players have independent eight bit graphics registers. Each missile is defined by two bits of data which is stored in a register location shared by the other three missiles. These registers may be reloaded at any time by the microprocessor, although they are usually changed during horizontal blank time. The data in these graphics registers is placed on the display whenever the horizontal counter equals the corresponding horizontal position registers. The same data will be displayed every line unless the graphics registers are reloaded with new data. These player/missile graphics registers may also be reloaded automatically from memory with direct memory access (DMA).

9.14) Color-Luminance Control

PLAYER/MISSILE COLOR-LUM:

COLPM0 (Color-lum of player/missile pair 0) Addr. = C012
COLPM1 (Color-lum of player/missile pair 1) Addr. = C013
COLPM2 (Color-lum of player/missile pair 2) Addr. = C014
COLPM3 (Color-lum of player/missile pair 3) Addr. = C015

PLAYFIELD COLOR-LUM:

COLPF0 (Color-lum of playfield 0) Addr. = C016
COLPF1 (Color-lum of playfield 1) Addr. = C017
COLPF2 (Color-lum of playfield 2) Addr. = C018
COLPF3 (Color-lum of playfield 3) Addr. = C019

BACKGROUND COLOR-LUM:

COLBK (Color-lum of backfield) Addr. = C01A

9.14) Color-Luminance Control (cont'd)

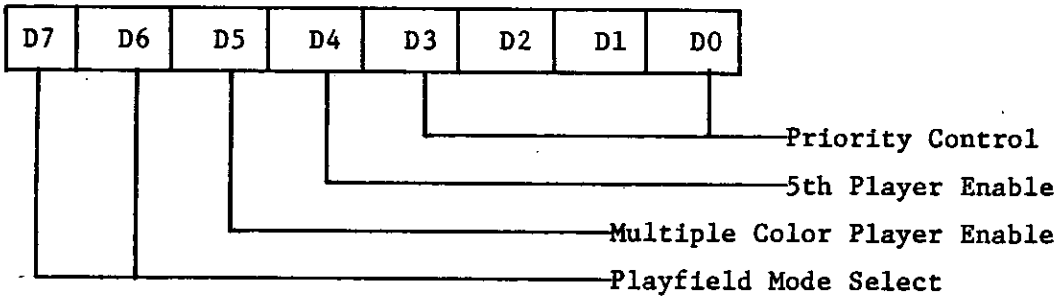
Color				Luminance			
D7	D6	D5	D4	D3	D2	D1	Not Used
X	X	X	X	0	0	0	Zero Luminance (black)
X	X	X	X	0	0	1	
					Etc.		
X	X	X	X	1	1	1	Max. Luminance (white)
0	0	0	0	Grey			
0	0	0	1	Gold			
0	0	1	0	Orange			
0	0	1	1	Red-Orange			
0	1	0	0	Pink			
0	1	0	1	Purple			
0	1	1	0	Purple-Blue			
0	1	1	1	Blue			
1	0	0	0	Blue			
1	0	0	1	Light Blue			
1	0	1	0	Turquoise			
1	0	1	1	Green-Blue			
1	1	0	0	Green			
1	1	0	1	Yellow-Green			
1	1	1	0	Orange-Green			
1	1	1	1	Light Orange			

A color-luminance register is used on the CGIA for each player/missile pair and playfield type. Each color-luminance register is loaded by the microprocessor with a code representing the desired color and luminance of its corresponding player/missile or playfield type. As the serial data of the different objects pass through the CGIA, it gets "impressed" with the color and luminance values in these registers. Therefore, when a player, missile or playfield is turned on, the corresponding color and luminance will be turned on. To prevent a color-luminance conflict, priority is established.

9.15) Priority Control

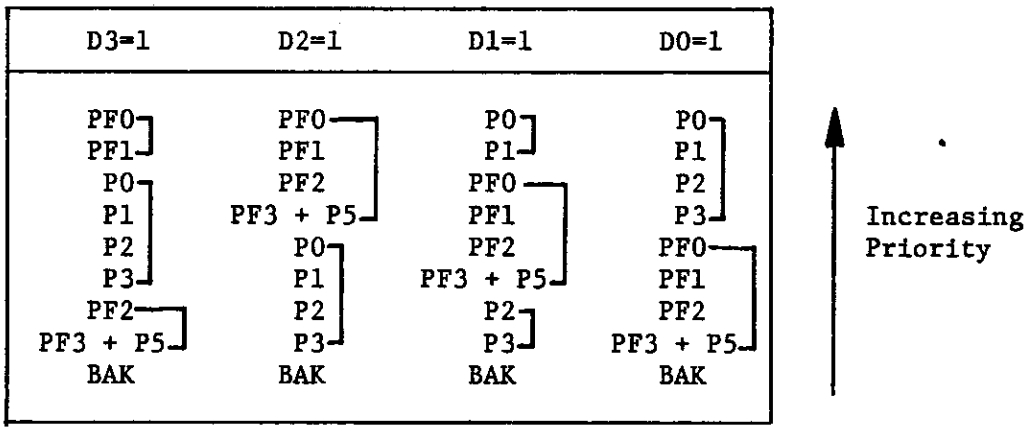
PRIORITY:

PRIOR (Priority, 5th player, playfield mode) Addr. = C01B



When moving objects such as players, missiles and playfield overlap on the T.V. screen, a decision must be made as to which object shows in front of the other. Objects which appear to pass in front of the other objects are said to have priority over them. Priority is assigned to each object by the CGIA before the serial data from each object is combined with the other objects and sent out to the T.V. screen. Setting the priority is done by writing to the CGIA priority control register.

Priority Select--(Mutually Exclusive)
Bits 0-3 select one of four types of priority.
Objects with higher priority will appear to move
in front of objects with lower priority.



Note: The use of priority bits is a "non-exclusive" mode (ie. not more than one bit true at the same time). More than one bit true at any time will result in objects whose priorities are in conflict to turn black in the overlap region.

9.16) Fifth Player Control

The priority control register also controls the fifth player. The fifth player is the combination of all four missiles and is shown as playfield 3 color-lum. However, there is no priority between playfields, therefore, the fifth player would have no priority between playfields. Player/missile combinations are of the same priority.

D4	PRIOR Addr. = C01B

This bit causes all missiles to assume the color of playfield 3. This allows missiles to be positioned together with a common color for use as a fifth player.

9.17) Multiple Color Player Enable

The priority control register also controls multiple color players and missiles. When enabled, the color-lum of player/missile 0 and player/missile 1 is to be logically "ORed." Also the color-lum of player/missile 2 and player/missile 3 are to be logically "ORed." This permits overlapping the position of two players with the choice of a third color in the overlapping region.

D5	PRIOR Addr. = C01B

This bit causes the logical "OR" function of the bits of the colors of player/missile 0 with player/missile 1, and also of player/missile 2 with player/missile 3 when these player/missile pairs are overlapped.

9.18) Playfield Mode Control

Besides priority, fifth player and multiple color player control, the priority control register also controls the playfield data interpretation. There are four playfield modes: They are 4 color-4 luminance (standard mode), 1 color-16 luminance (GTIA mode), 9 color-9 luminance (GTIA mode), and 16 color-1 luminance (GTIA mode).

D7	D6	PRIOR addr. = C01B
0	0	Standard no GTIA mode (4 color, 4 luminance)
0	1	1 color, 16 luminances mode
1	0	9 colors, 9 luminances mode
1	1	16 colors, 1 luminance mode

9.19) Player/Missile Vertical Delay

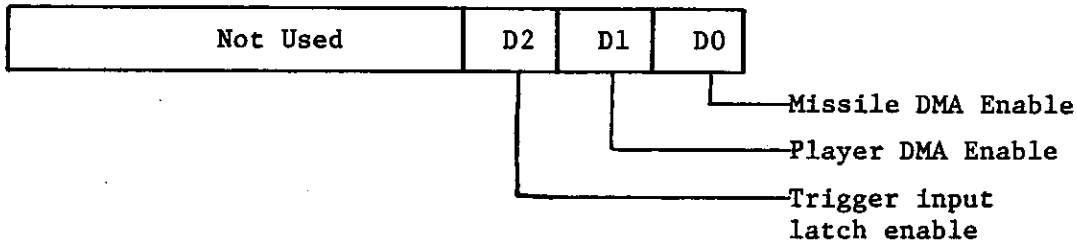
VDELAY (Vertical delay for players and missiles) Addr. = C01C

D7	D6	D5	D4	D3	D2	D1	D0
P3	P2	P1	P0	M3	M2	M1	M0

Each player or missile can be delayed by one vertical line. VDELAY is used to give one-line resolution in the vertical positioning of an object when the two-line resolution is enabled. Setting a bit in the VDELAY register to a logical one will move the corresponding object down by one T.V. line. Note that vertical delay can only be used when player/missile DMA is enabled.

9.20) Graphics Control

GRACTL (Player/missile DMA and trigger latch control) Addr. = C01D



D0=1 Enable missile DMA to missile graphics register
D1=1 Enable player DMA to player graphics registers
D2=1 Enable latches on T_0-T_3 inputs (latches are cleared and T_0-T_3 act as normal inputs when this bit is zero).

Player/missile graphics are handled automatically when the DMA control bits of the GRACTL register are enabled. The graphics data is loaded into the graphics registers during the horizontal blank time of every scan line. DMACTL bits D3 and D2 must be set.

The GRACTL register also controls the T_0-T_3 input latches. If the latch enable bit is set, the trigger input data is latched in the trigger register. The trigger register data is not cleared until GRACTL bit 2 is reset.

9.21) Collision Register Clear

HITCLR (Collision "hit" clear) Addr. = C01E

Not Used

The collision register bits can be cleared by writing to a single register HITCLR. All collision register bits are cleared when this is done.

10.0) Read-Only Registers

10.1) Vertical Line Counter

VCOUNT (Vertical Line Counter) Addr. = D40B

D7	D6	D5	D4	D3	D2	D1	D0
V8	V7	V6	V5	V4	V3	V2	V1

V0

V0 is not read.
Two line resolution supplied.

The current TV line may be determined by reading the vertical line counter (VCOUNT). This register gives the line count divided by 2. There are 262 lines per field, so VCOUNT runs from 0 to 130. The zero point occurs near the end of vertical blank.

10.2) Horizontal and Vertical Light Pen Registers

PENH (Horizontal Light Pen Value) Addr. = D40C

D7	D6	D5	D4	D3	D2	D1	D0
PH7	PH6	PH5	PH4	PH3	PH2	PH1	PH0

PENV (Vertical Light Pen Value) Addr. = D40D

D7	D6	D5	D4	D3	D2	D1	D0
PV8	PV7	PV6	PV5	PV4	PV3	PV2	PV1

PV0

V0 is not read.
The same as VCOUNT.

The PENH and PENV registers are used in conjunction with the $\overline{LP}$ input. When the $\overline{LP}$ input makes a transition from high to low, the current horizontal clock value is stored in PENH and the current VCOUNT value is stored in the PENV register. The PENV register is similar to the VCOUNT register in that the line count is divided by two, which provides two line resolution.

10.3) Non-Maskable Interrupt Status

NMIST (NMI Interrupt Status) Addr. = D40F

D7	D6	D5	Not Used (Zero Forced)

D7=1 Interrupt caused by display list interrupt
D7=0 No display list interrupt

D6=1 Interrupt caused by occurrence of vertical blank
D6=0 No vertical blank interrupt

D5=1 Interrupt caused by $\overline{\text{RNMI}}$ input low transition
D5=0 No $\overline{\text{RNMI}}$ input transition

The NMIST register is used to indicate which of the three NMI sources caused the NMI. The NMIST register should be read first and then cleared by writing to the NMIRES register to clear the source of the interrupt.

10.4) Missile to Playfield Collisions

M0PF (Missile 0 to playfield collisions) Addr. = C000
M1PF (Missile 1 to playfield collisions) Addr. = C001
M2PF (Missile 2 to playfield collisions) Addr. = C002
M3PF (Missile 3 to playfield collisions) Addr. = C003

Not Used (zero forced)	D3	D2	D1	D0	
	3	2	1	0	Playfield Type

These registers are used to determine if there has been a collision of one of the four missiles with the different playfields. A collision is detected as a logic one. Writing to the HITCLR register will cause all collision register bits to be reset.

10.5) Player to Playfield Collisions

P0PF (Player 0 to playfield collisions) Addr. = C004
P1PF (Player 1 to playfield collisions) Addr. = C005
P2PF (Player 2 to playfield collisions) Addr. = C006
P3PF (Player 3 to playfield collisions) Addr. = C007

Not Used (zero forced)	D3	D2	D1	D0	
	3	2	1	0	Playfield Type

These registers are used to determine if there has been a collision of one of the four players with the different playfields. A collision is detected as a logic one. Writing to the HITCLR register will cause all collision register bits to be reset.

10.6) Missile to Player Collisions

M0PL (Missile 0 to player collisions) Addr. = C008
M1PL (Missile 1 to player collisions) Addr. = C009
M2PL (Missile 2 to player collisions) Addr. = C00A
M3PL (Missile 3 to player collisions) Addr. = C00B

Not Used (zero forced)	D3	D2	D1	D0	
	3	2	1	0	Player Number

These registers are used to determine if there has been a collision of one of the four missiles with the different players. A collision is detected as a logic one. Writing to the HITCLR register will cause all collision register bits to be reset.

10.7) Player to Player Collisions

P0PL (Player 0 to player collisions) Addr. = C00C
P1PL (Player 1 to player collisions) Addr. = C00D
P2P1 (Player 2 to player collisions) Addr. = C00E
P3PL (Player 3 to player collisions) Addr. = C00F

Not Used (zero forced)	D3	D2	D1	D0	
	3	2	1	0	Player Number

These registers are used to determine if there has been a collision of one of the four players with any of the other players. A collision is detected as a logic one. Player to same player collisions are always forced to a logic zero. Writing to the HITCLR register will cause all collision register bits to be reset.

10.8) Collisions (Special Conditions)

Altogether there are 60 bits of collision detection provided to detect and store overlap (hits) between players, missiles, and playfield. These collisions can be read by the microprocessor at any time but are generally read during the vertical blank time. There are no bits for missile to missile or playfield to playfield collisions. There are only 12 bits of player to player collision because P0 to P0, etc. will always read as zero. In the high resolution mode (one pixel per ½ color clock), the playfield is represented by playfield 1 luminance and playfield 2 color. In this mode, playfield collision is stored as playfield 2 collision.

10.9) Trigger Input Latches

TRIG0 (Trigger 0 input latch) Addr. = C010
TRIG1 (Trigger 1 input latch) Addr. = C011
TRIG2 (Trigger 2 input latch) Addr. = C012
TRIG3 (Trigger 3 input latch) Addr. = C013

Not Used (zero forced)	D0

0 = Button Pressed
1 = Button Not Pressed

Trigger button data is accessed by the microprocessor by reading TRIG0-TRIG3. If bit 2 of GRACTL is set to a logic one, trigger input data is latched whenever the T_0-T_3 inputs go to a logic zero. These latches are reset (logic one) when bit 2 of GRACTL is set to a logic zero.

10.10) Television Standards Register

PAL (Television standard identification) Addr. = C014

Not Used (zero forced)	D3	D2	D1	D0	
	1	1	1	1	= NTSC Video Standard
	0	0	0	1	= PAL Video Standard

The PAL register is used to determine which version of the CGIA is being used. If bits D3-D0 = HEX F, then the NTSC video standard version of the CGIA is being used. If bits D3-D0 = Hex 1, then the PAL video standard version of the CGIA is being used.

10.11) Enable Test Functions

TEST1 (Enable test function one) Addr. = C016
TEST2 (Enable test function two) Addr. = C017
TEST3 (Enable test function three) Addr. = C018

Not Used (zero forced)

These addresses enable special test functions which are intended to be used for LSI test purposes only. Software should not read these addresses under any circumstances. Reading these addresses will cause internal control signals to be output on the S_0-S_3 output lines. The special test functions can be disabled at any time by writing to the CONSOL register at address C01F. The test functions are not disabled on reset, so it is a good idea to write to the CONSOL register on power-up.

11.0) Read/Write Registers

There is only one read/write register on the CGIA. This register is used to input and output four bits of parallel data on the S0-S3 I/O lines. Data direction is achieved indirectly by writing to the switch I/O port.

11.1) CONSOL (Switch I/O Port) Read/Write Register

Write location

CONSOL (Write to switch I/O port) Addr. = C01F (write)

Not Used	D3	D2	D1	D0
	S3	S2	S1	S0

0 = Output a logic one and enable input
1 = Output a logic zero and disable input

The switch lines have open-drain output devices which are pulled-up to V_{CC}. In order to use the switch lines as inputs, the output transistor must be turned off by writing a zero to the CONSOL write register. Writing to the CONSOL register will reset the special test functions (refer to special test functions for more information). It is advisable to read the switch I/O port on power-up in order to disable the test functions because the test functions are not pre-set to any particular state on a cold start.

Read Location

CONSOL (Read switch I/O port) Addr. = C01F (read)

Not Used (zero forced)	D3	D2	D1	D0
	S3	S2	S1	S0

This register reads the conditions of the switch lines (S0-S3). In order to use a switch line as an input, the output bit(s) (CONSOL write) must be zero. Any write bit which is set to a logic one will cause the corresponding read register bit to read as a logic 0 regardless of the input condition.

12.0) CGIA Memory Map

The CGIA is the combination of the ANTIC and GTIA custom LSI devices. In order to maintain system compatibility, the address decode is the same as it is on the 5200XL (PAM) home video game system.

The ANTIC section of the CGIA occupies address space D4XX. Many of these address locations are not actually used or are memory shadows. All of the registers are either read-only or write-only registers. The same address location may select two different registers depending on whether the memory access is a read or write operation. There are four read-only addresses in the ANTIC section of the CGIA. These registers are accessed at D4XB-D4XD and D4XF. The other read addresses from D4X0-D4XA and D4XE are not used and will return invalid data if read. There are 11 write-only registers in the ANTIC section of the CGIA. These registers are accessed at D4X0-D4X5, D4X7, D4X9-D4XA and D4XE-D4XF. The other write addresses, D4X6 and D4XB-D4XD are not used.

The GTIA section of the CGIA occupies address space CXXX. Many of these address locations are not actually used or are memory shadows. All of the registers are either read-only or write-only registers with the exception of the Switch I/O port, which is a read/write register. The same address location may select two different registers depending on whether the memory access is a read or write operation. There are 24 read-only addresses in the GTIA section of the CGIA. These registers are accessed at CX00-CX14 and CX16-Cx18. The other read addresses from CX15 and CX19-CX1E are not used and will return invalid data if read. There are 31 write-only addresses in the GTIA section of the CGIA. These registers are accessed at CX00-CX1E. The single read/write register is accessed at CX1F. Refer to the CGIA address table at the end of this specification for more information.

IT SHOULD BE NOTED THAT READ ADDRESSES CX16-CX18 ARE RESERVED FOR LSI TESTING PURPOSES AND SHOULD NOT BE READ BY SOFTWARE DURING NORMAL OPERATION.

ABSOLUTE MAXIMUM RATINGS

Voltage at any pin (with respect to V_{SS}).....-0.5V to +9.0V
Operating temperature range.....0°C to +70°C
Storage temperature range.....-40°C to +90°C

D.C. OPERATING CHARACTERISTICS

V_{CC}=5 Volts ±5%, V_{SS} = 0 Volts, T_A = 0 to +70°C

Pin Name	Parameter	Symbol	Min.	Max.	Units	Conditions/Comments
A_0-A_4	Input High Voltage	V_{IH}	2.0	V_{CC}	Volts	
A_8-A_{15}	Input Low Voltage	V_{IL}	V_{SS}	0.8	Volts	
	Leakage Current	I_L		10.0	uA	V_{IN}= +7.0V, output turned off
	Input Capacitance	C_{IN}		10.0	pF	
A_0-A_{15}	Output High Voltage	V_{OH}	2.4		Volts	I_{LOAD}=-0.1mA
	Output Low Voltage	V_{OL}		0.4	Volts	I_{LOAD}=+1.6mA
	Load Capacitance	C_{LOAD}		30.0	pF	
$\overline{NMI}$	Output High Voltage	V_{OH}	2.8		Volts	I_{LOAD}=-0.1mA
	Output Low Voltage	V_{OL}		0.4	Volts	I_{LOAD}=+1.6mA
	Load Capacitance	C_{LOAD}		30.0	pF	
RDY	Output Low Voltage	V_{OL}		0.4	Volts	I_{LOAD}=+1.6mA/ Open-Drain Output
	Leakage Current	I_L		10.0	uA	V_{IN}=+7.0V, pull-down turned off
	Load Capacitance	C_{LOAD}		30.0	pF	
ϕ0	Output High Voltage	V_{OH}	2.8		Volts	I_{LOAD}=-0.1mA
	Output Low Voltage	V_{OL}		0.4	Volts	I_{LOAD}=+1.6mA
	Load Capacitance	C_{LOAD}		25.0	pF	

D.C. OPERATING CHARACTERISTICS (cont'd)

V_{CC}=5 Volts ±5%, V_{SS} = 0 Volts, T_A= 0 to +70°C

Pin Name	Parameter	Symbol	Min.	Max.	Units	Conditions/Comments
$\overline{REF}$	Output High Voltage	V_{OH}	2.8		Volts	I_{LOAD}=0.1mA
	Output Low Voltage	V_{OL}		0.4	Volts	I_{LOAD}=+1.6mA
	Load Capacitance	C_{LOAD}		25.0	pF	
$\overline{RNMI}$	Input High Voltage	V_{IH}	2.0	V_{CC}	Volts	Internal Pull-Up
	Input Low Voltage	V_{IL}	V_{SS}	0.8	Volts	
	Pull-up Current	I_P	-100.0		uA	V_{IN}=2.4V
	Input Capacitance	C_{IN}		7.0	pF	
$\overline{RES}$	Pos. Thres. Voltage	V_{T+}	1.9	2.6	Volts	Schmitt Trigger Input
	Neg. Thres. Voltage	V_{T-}	1.0	2.1	Volts	
	Hysteresis	V_{HYS}	0.3		Volts	
	Leakage Current	I_L		10.0	uA	V_{IN}=+7.0V
	Input Capacitance	C_{IN}		7.0	pF	
$\overline{LP}$	Pos. Thres. Voltage	V_{T+}	1.9	2.6	Volts	Schmitt Trigger Input
	Neg. Thres. Voltage	V_{T-}	1.0	2.1	Volts	
	Hysteresis	V_{HYS}	0.3		Volts	
	Leakage Current	I_L		10.0	uA	V_{IN}=+7.0V
	Input Capacitance	C_{IN}		7.0	pF	

D.C. OPERATING CHARACTERISTICS (cont'd)

V_{CC}=5 Volts ±5%, V_{SS} = 0 Volts, T_A = 0 to +70°C

Pin Name	Parameter	Symbol	Min.	Max.	Units	Conditions/Comments
D_0-D_7	Input High Voltage	V_{IH}	2.0	V_{CC}	Volts	
	Input Low Voltage	V_{IL}	V_{SS}	0.8	Volts	
	Leakage Current	I_L		10.0	pF	
	Output High Voltage	V_{OH}	2.4		Volts	I_{LOAD}=0.1mA
	Output Low Voltage	V_{OL}		0.4	Volts	I_{LOAD}=+1.6mA
	Load Capacitance	C_{LOAD}		130.0	pF	
R/$\overline{W}$	Input High Voltage	V_{IH}	2.0	V_{CC}	Volts	
	Input Low Voltage	V_{IL}	V_{SS}	0.8	Volts	
	Leakage Current	I_L		10.0	uA	V_{IN}=+7.0V, output turned off
	Input Capacitance	C_{IN}		7.0	pF	
	Output High Voltage	V_{OH}	2.4		Volts	I_{LOAD}= -0.1mA
	Load Capacitance	C_{LOAD}		30.0	pF	
$\overline{HALT}$	Output High Voltage	V_{OH}	2.8		Volts	I_{LOAD}=-0.1mA
	Output Low Voltage	V_{OL}		0.4	Volts	I_{LOAD}=+1.6mA
	Load Capacitance	C_{LOAD}		30.0	pF	
ϕ2	Input High Voltage	V_{IH}	2.0	V_{CC}	Volts	
	Input Low Voltage	V_{IL}	V_{SS}	0.8	Volts	
	Leakage Current	I_L		10.0	uA	V_{IN}=+7.0V
	Input Capacitance	C_{IN}		10.0	pF	

D.C. OPERATING CHARACTERISTICS (Cont'd)

V_{CC}=5 Volts ±5%, V_{SS} = 0 Volts, T_A = 0 to +70°C

Pin Name	Parameter	Symbol	Min.	Max.	Units	Conditions/Comments
T_0-T_3	Input High Voltage	V_{IH}	2.0	V_{CC}	Volts	Internal Pull-Ups
	Input Low Voltage	V_{IH}	V_{SS}	0.8	Volts	
	Pull-Up Current	I_P	-100.0		uA	V_{IN}=2.4V
	Input Capacitance	C_{IN}		25.0	pF	
S_0-S_3	Input High Voltage	V_{IH}	2.0	V_{CC}	Volts	Internal Pull-up
	Input Low Voltage	V_{IL}	V_{SS}	0.8	Volts	
	Pull-Up Current	I_P	-100.0		uA	V_{IN}=2.4V
	Input Capacitance	C_{IN}		25.0	pF	
	Output Low Voltage	V_{OL}		0.4	Volts	I_{LOAD}=+1.6mA/ Open-Drain Output
	Load Capacitance	C_{LOAD}		30.0	pF	
OSC	Input High Voltage	V_{IN}	2.8	V_{CC}	Volts	
	Input Low Voltage	V_{IL}	V_{SS}	0.8	Volts	
	Leakage Current	I_L		10.0	uA	V_{IN}=+7.0V
	Input Capacitance	C_{IN}		7.0	pF	
COL	Output Low Voltage	V_{OL}		0.4	Volts	I_{LOAD}=+1.6mA/ Open-Drain Output
	Leakage Current	I_L		10.0	uA	V_{IN}=+7.0V, pull-down turned off
	Load Capacitance	C_{LOAD}		25.0	pF	
POWER REQUIREMENTS						
V_{CC}	Supply Voltage	V_{SUP}	4.75	5.25	Volts	
	Supply Current	I_{CC}		160.0	mA	

D.C. OPERATING CHARACTERISTICS (cont'd)

V_{CC}=5 Volts ±5%, V_{SS} = 0 Volts, T_A = 0 to +70°C

Pin Name	Parameter	Symbol	Min.	Max.	Units	Conditions/Comments
CLUM	Pull-Up Impedance	Z_{PU}	0.55	1.10	KOhms	Measured between V_{CC} and CLUM
	Load Capacitance	C_{LOAD}		50.0	pF	
	Output Voltage	V_{OUT}	SEE TABLE BELOW			

Digital Output Value SLLLL YUUUU NMMMM C3210	Multiplier	Output Variance (± Percent)	Output Minimum (Volts)	Output Typical (Volts)	Output Maximum (Volts)
	All output voltages are measured at V_{CC}=5 Volts and no load current.				
00000	0.5651	1.0	2.5686	2.8255	3.1081
10000	0.6941	0.5	3.3052	3.4705	3.6440
10001	0.7091	0.5	3.3767	3.5455	3.7228
10010	0.7241	0.5	3.4481	3.6205	3.8015
10011	0.7401	0.5	3.5243	3.7005	3.8855
10100	0.7560	0.5	3.6000	3.7800	3.9690
10101	0.7741	0.5	3.6862	3.8705	4.0640
10110	0.7931	0.5	3.7767	3.9655	4.1638
10111	0.8121	0.5	3.8671	4.0605	4.2635
11000	0.8260	0.5	3.9333	4.1300	4.3365
11001	0.8470	0.5	4.0333	4.2350	4.4468
11010	0.8700	0.5	4.1429	4.3500	4.5675
11011	0.8930	0.5	4.2524	4.4650	4.6883
11100	0.9160	0.5	4.3619	4.5800	4.8090
11101	0.9420	0.5	4.4857	4.7100	4.9455
11110	0.9690	0.5	4.6143	4.8450	5.000
11111	1.0000	0.5	4.7619	5.0000	5.000

Notes: The sync and lum values are representative of the CSYNC and L_0-L_3 output signals on the GTIA. Luminance values L_1-L_3 are derived from the color/lum register bits D_1-D_3. Luminance values L_0-L_3 are derived from graphics data stored in system memory when in GTIA mode 1 (1 color-16 luminances).

$$V_{OUT}=(M \times V_{CC}) \pm V$$

Where: M=Multiplier from above
V=Variance from above

$$Z_{OUT}=\frac{(M \times Z_{PU})}{1-M}$$

Where: M=Multiplier from above
Z_{PU}=Pull-Up Impedance from above

Z_{OUT} is measured between CLUM and V_{SS}. The CLUM output is a high impedance output. The CLUM output characteristics are subject to change with varying output loads.

D.C. OPERATING CHARACTERISTICS (cont'd)

CLUM Characteristic Output Voltage Levels

SLLLL YUUUU NMMMM C3210	Output Voltage @ V_{CC}=4.75V T_A=100°C	Output Voltage @ V_{CC}=5.00V T_A=25°C	Output Voltage @ V_{CC}=5.25V T_A=0°C
00000	2.684V-2.697V	2.818V-2.826V	2.955V-2.960V
10000	3.293V-3.301V	3.462V-3.467V	3.632V-3.635V
10001	3.364V-3.372V	3.536V-3.542V	3.711V-3.714V
10010	3.438V-3.446V	3.614V-3.620V	3.792V-3.796V
10011	3.516V-3.524V	3.696V-3.701V	3.878V-3.881V
10100	3.592V-3.599V	3.777V-3.782V	3.963V-3.966V
10101	3.677V-3.684V	3.866V-3.871V	4.056V-4.600V
10110	3.765V-3.773V	3.959V-3.964V	4.154V-4.158V
10111	3.859V-3.866V	4.057V-4.062V	4.257V-4.261V
11000	3.921V-3.924V	4.125V-4.127V	4.331V-4.332V
11001	4.022V-4.025V	4.232V-4.234V	4.443V-4.444V
11010	4.128V-4.131V	4.344V-4.346V	4.561V
11011	4.241V-4.243V	4.462V-4.464V	4.685V-4.686V
11100	4.352V-4.353V	4.581V	4.809V-4.810V
11101	4.477V-4.478V	4.713V	4.948V
11110	4.609V	4.852V	5.090V
11111	4.750V	5.000V	5.250V

Notes: The sync and lum values are representative of the CSYNC and L_0-L_3 output signals on the GTIA. Luminance values L_1-L_3 are derived from the color/lum register bits D_1-D_3. Luminance values L_0-L_3 are derived from graphics data stored in system memory when in GTIA mode 1 (1 color-16 luminances).

The CLUM output voltage values in the table above are guaranteed by design only and are not tested for. This information is presented for characterization purposes only. These values illustrate worst case speed and power conditions as well as the characteristic CLUM output voltage under normal operating conditions.

DYNAMIC OPERATING CHARACTERISTICS

V_{CC} = 5 Volts ± 5%, V_{SS} = 0 Volts, T_A=0 to +70°C

Ref. No.	Parameter	Description	Ref. Pt.	Min.	Typ.	Max.	Units
	F_{OSC} $F_{\phi 0}$ $F_{\phi 2}$	OSC Clock Input Frequency $\phi 0$ Clock Output Frequency $\phi 2$ Clock Input Frequency			3.579 $F_{OSC}/2$ $F_{OSC}/2$		MHz
1	T_{ROSC}	OSC Clock Rise Time				15	nS
2	T_{FOSC}	OSC Clock Fall Time				15	nS
3	T_{OSCHI}	OSC Clock High Time		135		145	nS
4	T_{OSCYC}	OSC Clock Cycle Time			279		nS
5	T_{RC0}	$\phi 0$ Clock Rise Time				50	nS
6	T_{FC0}	$\phi 0$ Clock Fall Time				50	nS
7	T_{C0HI}	$\phi 0$ Clock High Time		270		290	nS
8	T_{C0D}	$\phi 0$ Clock Output Delay	ALE OSC			165	nS
9	T_{RC2}	$\phi 2$ Clock Rise Time				25	nS
10	T_{FC2}	$\phi 2$ Clock Fall Time				25	nS
11	T_{C2HI}	$\phi 2$ Clock High Time		230		260	nS
12	T_{ADSI}	Address Setup Time (input)	BLE $\phi 2$	130			nS
13	T_{ADHI}	Address Hold Time (input)	ATE $\phi 2$	30			nS
14	T_{ADSO}	Address Setup Time (output)	ATE $\phi 2$			145	nS
15	T_{ADHO}	Address Hold Time (output)	ATE $\phi 2$	14			nS
16	T_{DSI}	Data Setup Time (input)	BTE $\phi 2$	50			nS
17	T_{DHI}	Data Hold Time (input)	ATE $\phi 2$	10			nS
18	T_{DSO}	Data Setup Time (output)	ALE $\phi 2$			185	nS
19	T_{DHO}	Data Hold Time (output)	ATE $\phi 2$	20			nS
20	T_{RWSI}	R/$\overline{W}$ Setup Time (input)	BLE $\phi 2$	130			nS
21	T_{RWHI}	R/$\overline{W}$ Hold Time (input)	ATE $\phi 2$	30			nS
22	T_{RWSO}	R/$\overline{W}$ Hold Time (output)	ATE $\phi 2$			230	nS
23	T_{RWHO}	R/$\overline{W}$ Hold Time (output)	ATE $\phi 2$	23			nS
24	T_{RDYS}	RDY Output Setup Time	ATE $\phi 2$			180	nS
25	T_{RDYH}	RDY Output Hold Time	ATE $\phi 2$	18			nS

DYNAMIC OPERATING CHARACTERISTICS (cont'd)

V_{CC} = 5 Volts ± 5%, V_{SS} = 0 Volts, T_A = 0 to +70°C

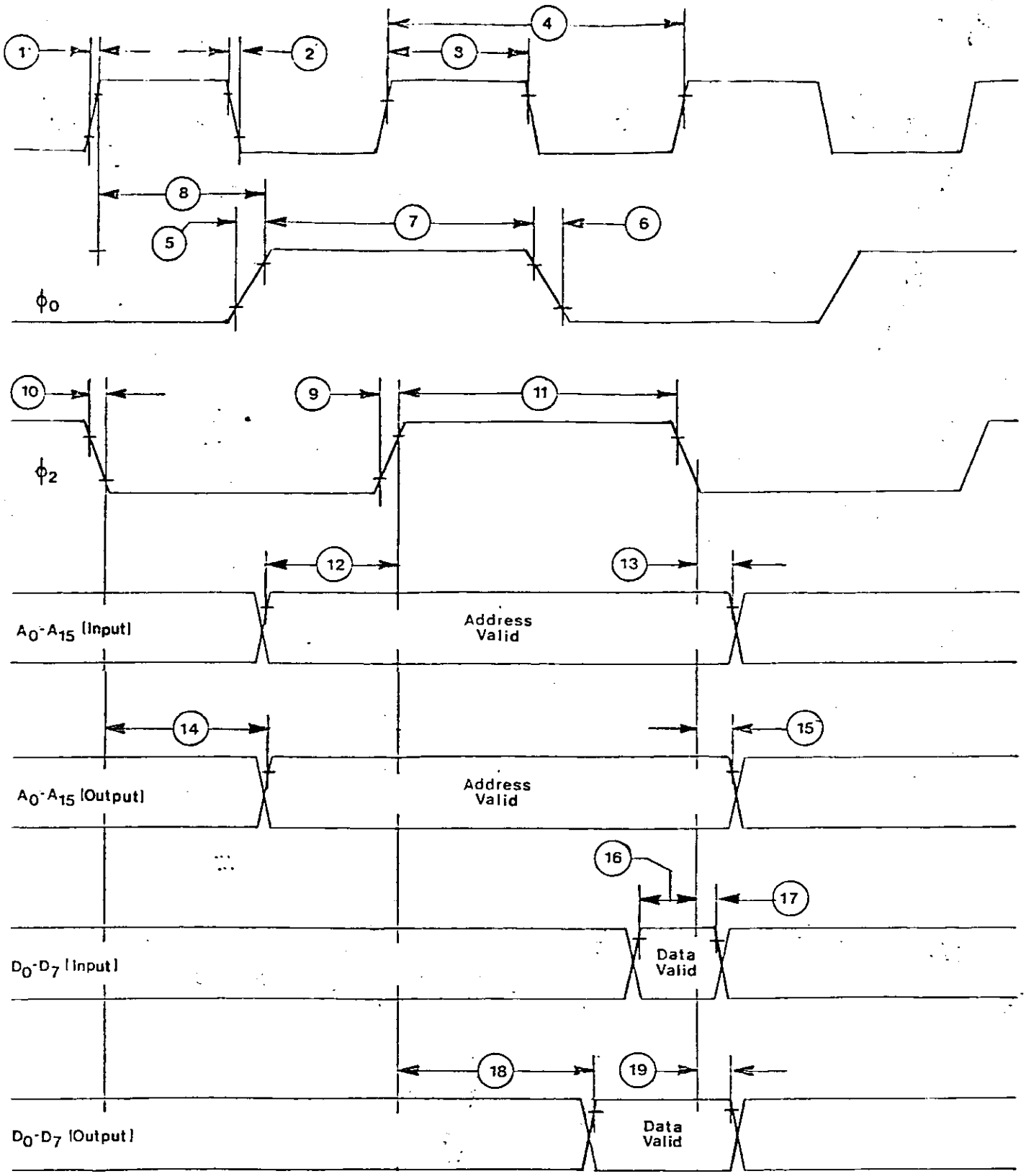
Ref. No.	Parameter	Description	Ref. Pt.	Min.	Typ.	Max.	Units
26	T_{HNS}	$\overline{HALT}$, $\overline{NMI}$ Setup Time	ATE ϕ2			350	nS
27	T_{HNH}	$\overline{HALT}$, $\overline{NMI}$ Hold Time	ATE ϕ2	35			nS
28	T_{RNIS}	$\overline{RNMI}$ Input Setup Time	BTE ϕ2	50			nS
29	T_{RNIH}	$\overline{RNMI}$ INput Hold Time	ATE ϕ2	10			nS
30	T_{RESS}	$\overline{RES}$ Input Setup Time	BTE OSC	50			nS
31	T_{RESH}	$\overline{RES}$ Input Hold Time	ATE OSC	130			nS
32	T_{REFS}	$\overline{REF}$ Output Setup Time	ATE ϕ2			150	nS
33	T_{REFH}	$\overline{REF}$ Output Hold Time	ATE ϕ2	15			nS
34	T_{LPS}	$\overline{LP}$ Input Setup Time	BTE ϕ2	50			nS
			BLE ϕ2	50			nS
35	T_{LPH}	$\overline{LP}$ Input Hold Time	ATE ϕ2	850			nS
			ALE ϕ2	850			nS
36	T_{STSI}	S0-S3,T0-T3 Input Setup Time	BLE ϕ2	100			nS
37	T_{STHI}	S0-S3,T0-T3 Input Hold Time	ATE ϕ2	100			nS
38	T_{SSO}	S0-S3 Output Setup Time	ATE ϕ2			800	nS
39	T_{CLUM}	CLUM Output Transition Time	ALE OSC			450	nS
			ATE OSC			450	nS

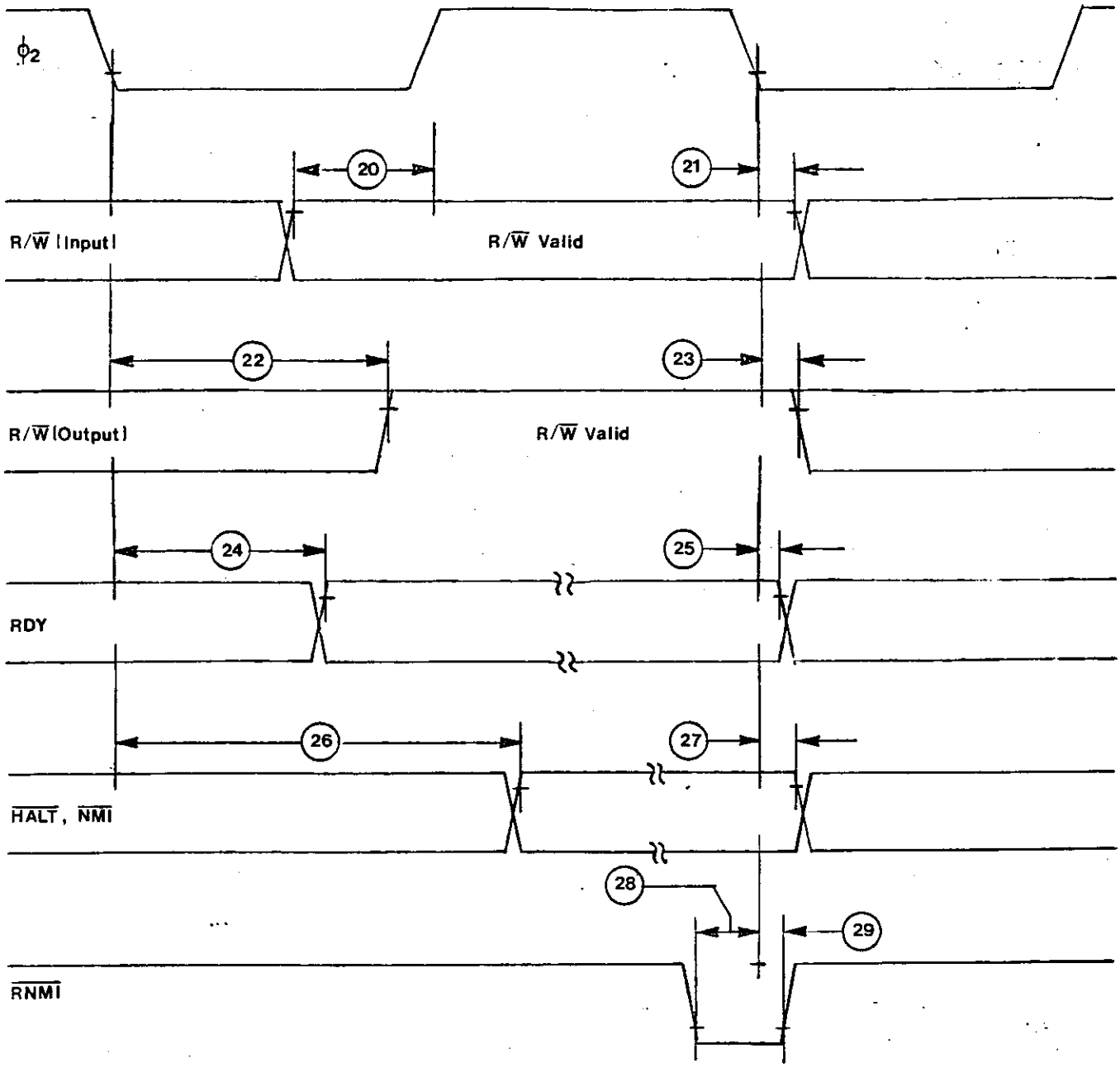
DYNAMIC OPERATING CHARACTERISTICS (Cont'd)

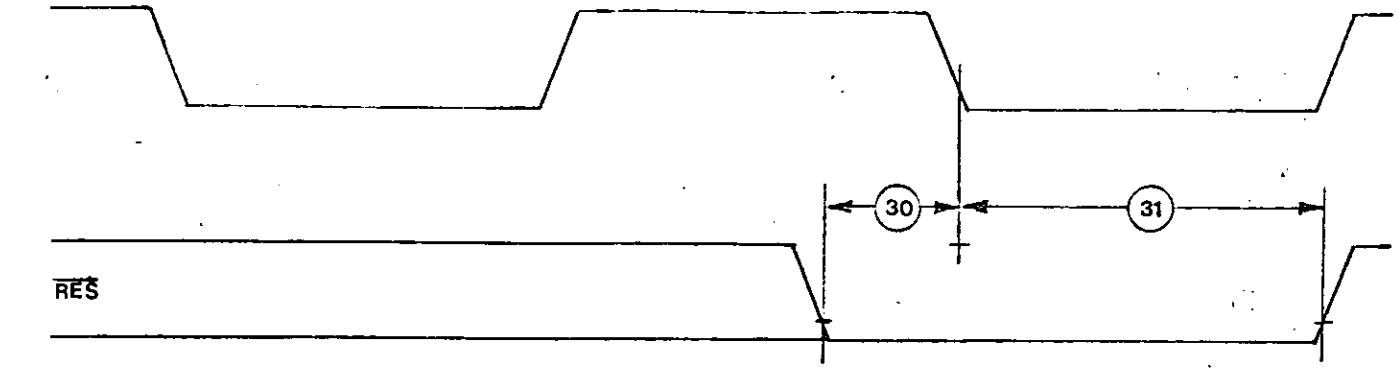
V_{CC} = 5 Volts ± .5%, V_{SS} = 0 Volts, T_A = 0 to +70°C

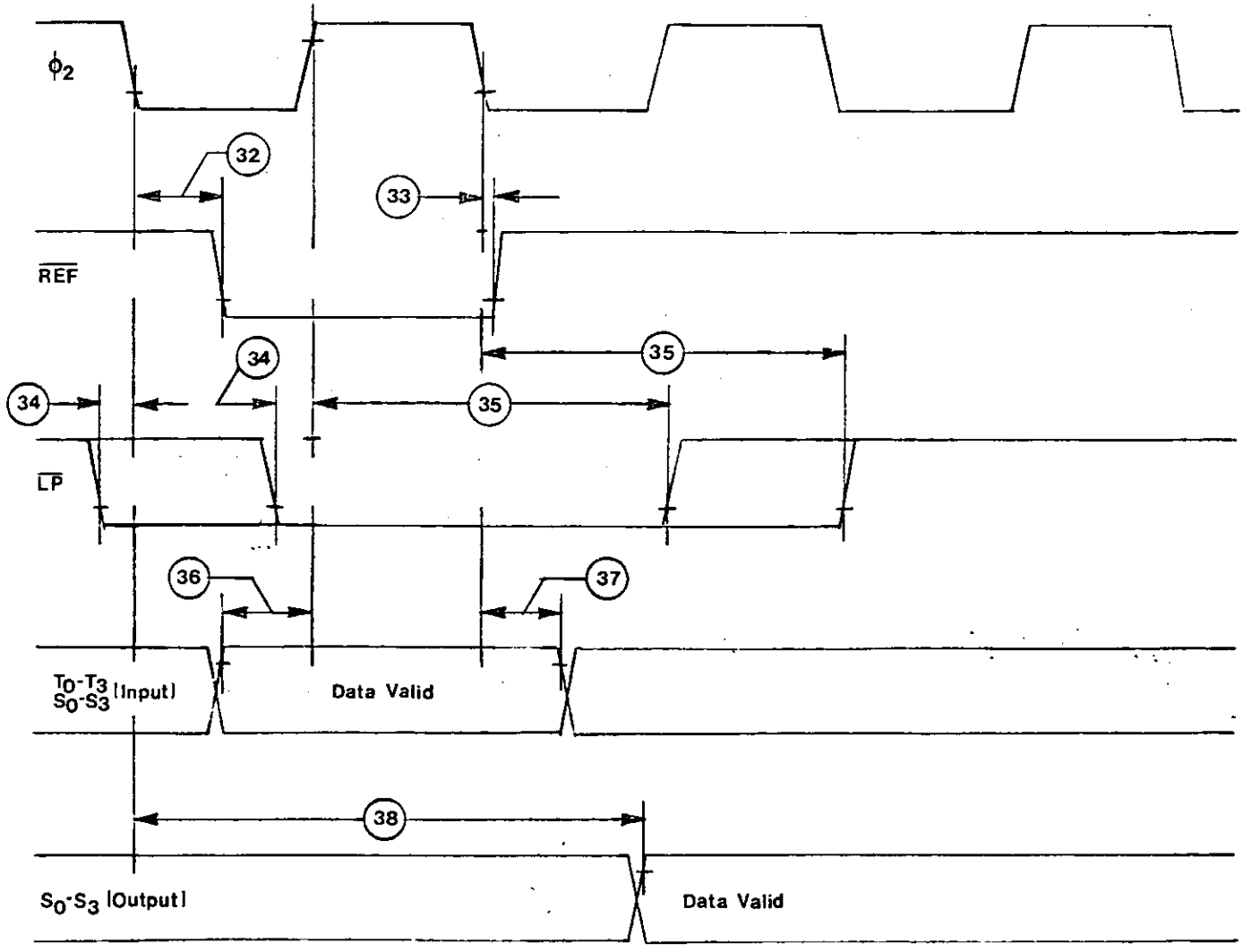
Ref. No.	Parameter	Description	Ref. Pt.	Min.	Typ.	Max.	Units
40	T_{INV}	COL Output Delay (1st stage)	ATE OSC			190	nS
41	Δt	Delta Output Delay Time		16		22	nS

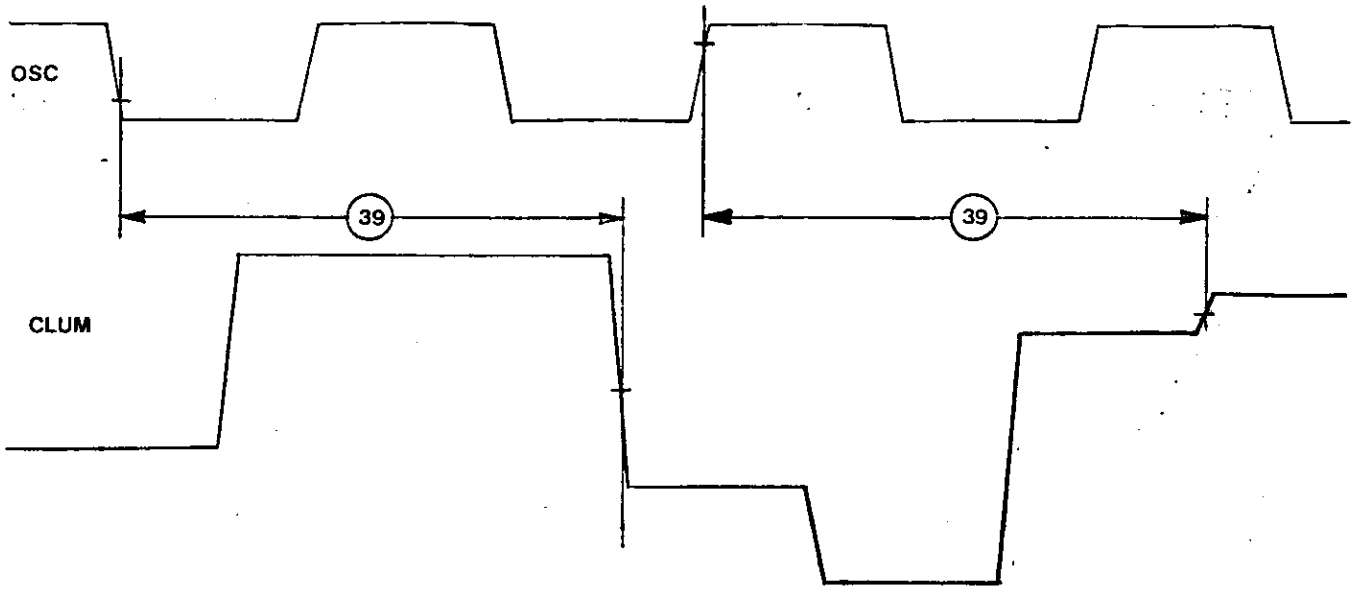
Color Register Value (HEX)	Color Displayed	Delay Time for Color Value
0	Grey	NO COLOR OUT
1	Gold	T_{INV} + 0 (Δt)
2	Orange	T_{INV} + 1 (Δt)
3	Red-Orange	T_{INV} + 2 (Δt)
4	Pink	T_{INV} + 3 (Δt)
5	Purple	T_{INV} + 4 (Δt)
6	Purple-Blue	T_{INV} + 5 (Δt)
7	Blue	T_{INV} + 6 (Δt)
8	Blue	T_{INV} + 7 (Δt)
9	Light-Blue	T_{INV} + 8 (Δt)
A	Turquoise	T_{INV} + 9 (Δt)
B	Green-Blue	T_{INV} + 10 (Δt)
C	Green	T_{INV} + 11 (Δt)
D	Yellow-Green	T_{INV} + 12 (Δt)
E	Orange-Green	T_{INV} + 13 (Δt)
F	Light-Orange	T_{INV} + 14 (Δt)

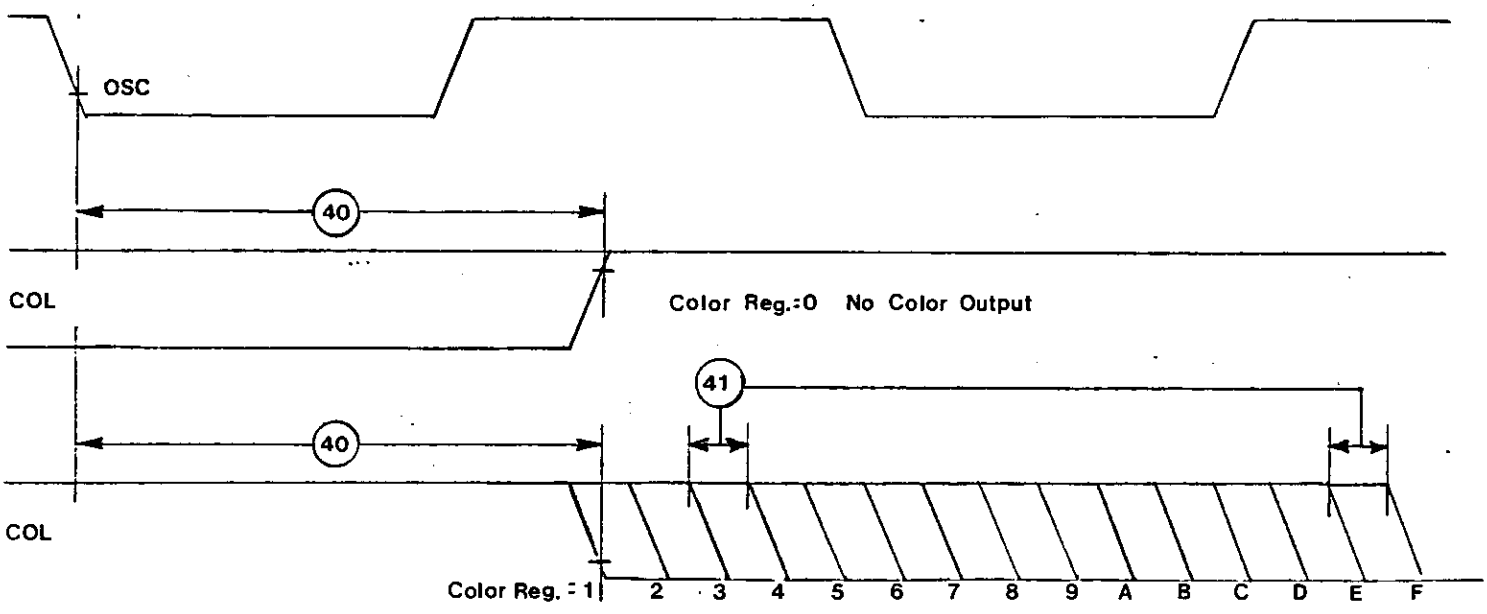

1
2
3
4
5
6
7
8
ϕ_0
10
9
11
ϕ_2
12
13
A_0-A_{15} (Input)
Address Valid
14
15
A_0-A_{15} (Output)
Address Valid
16
17
D_0-D_7 (Input)
Data Valid
18
19
D_0-D_7 (Output)
Data Valid


ϕ_2
20
21
R/$\overline{W}$ (Input)
R/$\overline{W}$ Valid
22
23
R/$\overline{W}$ (Output)
R/$\overline{W}$ Valid
24
25
RDY
26
27
$\overline{HALT}$, $\overline{NMI}$
28
29
$\overline{RNMI}$


30
31
RES


ϕ_2
32
33
REF
34
35
34
35
LP
36
37
T_0-T_3
S_0-S_3 (Input)
Data Valid
38
S_0-S_3 (Output)
Data Valid


OSC
39
39
CLUM


OSC
40
COL
Color Reg.=0 No Color Output
41
40
COL
Color Reg. = 1
2
3
4
5
6
7
8
9
A
B
C
D
E
F

CGIA Address Table
GTIA Section

Address	WRITE		READ	
	Name	Description	Name	Description
C000	HPOSP0	Horz. Posit. Player 0	M0PF	Read Missile to Playfield Collisions
C001	HPOSP1	Horz. Posit. Player 1	M1PF	
C002	HPOSP2	Horz. Posit. Player 2	M2PF	
C003	HPOSP3	Horz. Posit. Player 3	M3PF	
C004	HPOSM0	Horz. Posit. Missile 0	P0PF	Read Player to Playfield Collisions
C005	HPOSM1	Horz. Posit. Missile 1	P1PF	
C006	HPOSM2	Horz. Posit. Missile 2	P2PF	
C007	HPOSM3	Horz. Posit. Missile 3	P3PF	
C008	SIZEP0	Size Player 0	M0PL	Read Missile to Player Collisions
C009	SIZEP1	Size Player 1	M1PL	
C00A	SIZEP2	Size Player 2	M2PL	
C00B	SIZEP3	Size Player 3	M3PL	
C00C	SIZEM	Size All Missiles	P0PL	Read Player to Player Collisions
C00D	GRAFP0	Graphics Player 0	P1PL	
C00E	GRAFP1	Graphics Player 1	P2PL	
C00F	GRAFP2	Graphics Player 2	P3PL	
C010	GRAFP3	Graphics Player 3	TRIG0	Read Joystick Trigger Buttons
C011	GRAFM	Graphics All Missiles	TRIG1	
C012	COLPM0	Color-lum of Player-Missile 0	TRIG2	
C013	COLPM1	Color-lum of Player-Missile 1	TRIG3	
C014	COLPM2	Color-lum of Player-Missile 2	PAL	Read PAL/NTSC Bits
C015	COLPM3	Color-lum of Player-Missile 3		
C016	COLPF0	Color-lum of Playfield 0	TEST1	DO NOT READ
C017	COLPF1	Color-lum of Playfield 1	TEST2	DO NOT READ
C018	COLPF2	Color-lum of Playfield 2	TEST3	DO NOT READ
C019	COLPF3	Color-lum of Playfield 3		
C01A	COLBK	Color-lum of Background		
C01B	PRIOR	Priority Select		
C01C	VDELAY	Vertical Delay		
C01D	GRACTL	Graphic Control		
C01E	HITCLR	Collision Clear		
C01F	CONSOL	Write to Switch I/O Port CANCEL TEST MODE	CONSOL	Read from Switch I/O Port
C020 ↕ CFFF	REPEATED 127 TIMES AS ABOVE			

CGIA Address Table
ANTIC Section

ADDRESS	WRITE		READ	
	NAME	DESCRIPTION	NAME	DESCRIPTION
D400	DMACTL	DMA Control Register		
D401	CHACTL	Character Control Register		
D402	DLISTL	Display List Pointer (LOW BYTE)		
D403	DLISTH	Display List Pointer (HIGH BYTE)		
D404	HSCROL	Horizontal Scroll Register		
D405	VSCROL	Vertical Scroll Register		
D406				
D407	PMBASE	Player/Missile Base Address Register		
D408				
D409	CHBASE	Character Base Address Register		
D40A	WSYNC	Wait For Horizontal Sync		
D40B			VCOUNT	Vertical Line Counter
D40C			PENH	Horizontal Light Pen Register
D40D			PENV	Vertical Light Pen Register
D40E	NMIEN	Enable NMI Interrupts		
D40F	NMIRES	Reset NMI Interrupt Status Register	NMIST	NMI Interrupt Status Register
D410 ↕ D4FF	REPEATED 15 TIMES AS ABOVE			