The Engineering Staff of
TEXAS INSTRUMENTS INCORPORATED
Semiconductor Group

BILL BOURNE

TMS 9918A
VIDEO DISPLAY
PROCESSOR
DATA MANUAL

NOVEMBER 1980

TEXAS INSTRUMENTS
INCORPORATED

1. INTRODUCTION

1.1 DESCRIPTION

The TMS 9918A Video Display Processor (VDP) is an N-channel MOS LSI device used in video systems where data display on a raster-scanned home color television set or color monitor is desired. The TMS 9918A generates all necessary video, control, and synchronization signals and also controls the storage, retrieval, and refresh of display data in the dynamic screen refresh memory. The interfaces to the microprocessor, refresh memory, and the TV require a minimum of additional electronics.

The VDP has four video display modes: Graphics I, Graphics II, Multicolor and Text mode. The Text mode provides twenty-four 40-character rows in two colors and is intended to maximize the capacity of the TV screen to display alphanumeric characters. The Multicolor mode provides an unrestricted 64 X 48 color-dot display utilizing 15 colors plus transparent. The Graphics I mode provides a 256 X 192 pixel display for generating pattern graphics in 15 colors plus transparent. The Graphics II mode is an enhancement of Graphics I mode, providing the capability to generate more complex color and pattern displays.

The video display consists of 35 planes, external video, backdrop, pattern plane, and 32 Sprite Planes. The planes are vertically stacked with the external video being the bottom or innermost plane. The backdrop plane is the next plane followed by the pattern plane that contains Graphics I and Graphics II patterns with the 32 Sprite Planes as the top planes. (A sprite is an object-oriented animation pattern that can be moved smoothly across the screen.)

The TMS 9918A VDP utilizes either a 4K, 8K, or 16K-type low-cost dynamic memory (TMS 4027, TMS 4108, TMS 4116) for storage of the display parameters.

1.2 FEATURES

- Single-chip interface to color TV's (excluding RAM and RF modulator).
- 256 X 192 resolution on TV screen
- 15 unique colors plus transparent
- General 8-bit bidirectional interface to CPU
- Direct wiring to 4K, 8K, or 16K dynamic RAM memories
- Automatic and transparent refresh of dynamic RAMs
- External video input capability
- NTSC - standard composite video output
- Unique planar representation for 3D simulation
- Standard 40-pin package

1.3 TYPICAL APPLICATIONS

- Color computer terminals
- Home computers
- Drafting/design aids
- Teaching aids
- Industrial process monitoring
- Home educational systems
- Animation aids

The following example of a typical application may help introduce the user to the TMS 9918A VDP. Figure 1-1 is block diagram of a typical application. Each of the concepts presented below is described more fully in later sections of this manual.

The VDP basically has three interfaces: CPU, color television, and dynamic refresh RAM (VRAM) the contents of which define the TV image. The TMS 9918A VDP also has eight write-only registers and a read-only status register.

The VDP communicates with the CPU via an 8-bit bidirectional data bus. Three control lines, decoded from the CPU address and enable lines, determine interpretation of the bus. Through the bus, the CPU can write to VRAM, read from VRAM, write to VDP registers, and read the VDP status. The VDP also generates an interrupt signal after every refresh of the TV display, which may be useful to the CPU.

The dynamic RAM interface consists of direct wiring of eight 4K X 1, 8K X 1, or 16K X 1 dynamic RAS/CAS-type RAMs to the TMS 9918 VDP. The amount of RAM required is dependent upon the features selected for use in the application.

The interface to the TV can consist of wiring the VDP's composite video output pin (suitably buffered) to the input of a color or black-and-white monitor, or an appropriate RF modulator may be used to feed the signal into a TV antenna terminal.

The VDP can operate in any one of four modes, each of which can affect the way the VRAM is mapped onto the television screen. In Graphics I and II modes, characters are mapped onto the screen in 8 X 8 pixel blocks, yielding 24 lines of 32 blocks (or "pattern positions") each. In Text mode, there are 24 lines of 40 blocks, each of which is 6 X 8 pixels. In Multicolor mode, there are 48 lines of 64 blocks, each of which is composed of 4 X 4 pixels, all of one solid color. In addition to these, objects termed "sprites" can be superimposed onto the television image. Furthermore, signals entering the VDP through the external video input can be used as a background to VDP-generated images.

1.4 DEFINITIONS

The following definitions will be useful in understanding the use of the TMS 9918 VDP:

- pixel — the smallest point on the TV screen that can be independently controlled
- NTSC — National Television Standards Committee which specifies the television signal standard for the USA
- VRAM — Video RAM; refers to the dynamic RAMs that connect to the VDP and whose contents define the TV image
- sprite — An object whose pattern is relative to a specified X, Y coordinate and whose position can therefore be controlled by that coordinate with a positional resolution of one pixel

2. ARCHITECTURE

The TMS 9918A Video Display Processor (VDP) is designed to provide a simple interface between a microprocessor and a raster-scanned color television. Shown in Figure 2-1 is a block diagram of the major portions of the VDP architecture. Described below are details of the various interfaces to the VDP, CPU, VRAM, and color television.

2.1 CPU INTERFACE

The VDP interfaces to the CPU using an 8-bit bidirectional data bus, three control lines, and an interrupt as shown in Figure 2-2. Through this interface the CPU can conduct four operations: write data bytes to VRAM, read data bytes from VRAM, write to one of the eight VDP write-only registers, and read the VDP Status Register. Each of these operations requires one or more data transfers to take place over the CPU/VDP data bus interface. The interpretation of the data transfer is determined by the three control lines of the VDP. It should be noted that the CPU can communicate with the VDP simultaneously and asynchronously with the VDP's TV screen refresh operations. The VDP performs memory management and allows periodic intervals of CPU access to VRAM even in the middle of a raster scan.

2.1.1 CPU Interface Control Signals

The type and direction of data transfers are controlled by the $\overline{CSW}$, $\overline{CSR}$, and MODE inputs. $\overline{CSW}$ is the CPU-to-VDP write select. When it is active (low), the 8 bits on D0-D7 are strobed into the VDP. $\overline{CSR}$ is the CPU-from-VDP read select. When it is active (low), the VDP outputs 8 bits on D0-D7 to the CPU. $\overline{CSW}$ and $\overline{CSR}$ should never be simultaneously low. If both are low, the VDP outputs data on D0-D7 and latches in invalid data.

MODE determines the source or destination of a read or write data transfer. MODE is normally tied to a CPU low order address line (A14 for TMS 9900).

2.1.2 CPU Write to VDP Register

The VDP has eight write-only registers and one read-only status register. The write-only registers control the VDP operation and determine the way in which VRAM is allocated. The status register contains interrupt, sprite coincidence and fifth sprite status flags.

Each of the eight VDP write-only registers can be loaded using two 8-bit data transfers from the CPU. Table 1 describes the required format for the two bytes. The first byte transferred is the data byte, and the second byte transferred controls the destination. The most-significant bit of the second byte must be a '1'. The next four bits are '0's, and the lowest three bits make up the destination register number. The MODE input is high for both byte transfers.

To rewrite the data for an internal register after a byte of data has been loaded, the status register must be read so that internal logic will accept the next byte as data and not as a register destination. This situation may be encountered in interrupt-driven program environments. Whenever the status of VDP write parameters is in question, this procedure should be used. Note that the CPU address is destroyed by writing to the VDP register.

2.1.3 CPU Write to VRAM

The CPU transfers data to the VRAM through the VDP using a 14-bit autoincrementing address register. Two-byte transfers are required to set up the address register. A one-byte-transfer is then required to write the data to the addressed VRAM byte. The address register is then autoincremented. Sequential VRAM write require only one-byte-transfer since the address register is already set up. During setup of the address register, the two most-significant bits of the second address byte must be '0' and '1' respectively. MODE is high for both address transfers and low for the data transfer. $\overline{CSW}$ is used in all transfers to strobe the 8 bits into the VDP. See Table 1.

TABLE 1 – CPU/VDP DATA TRANSFERS

OPERATION	BIT 0	1	2	3	4	5	6	7	$\overline{CSW}$	$\overline{CSR}$	MODE
WRITE TO VDP REGISTER											
BYTE 1 DATA WRITE	D_0	D_1	D_2	D_3	D_4	D_5	D_6	D_7	0	1	1
BYTE 2 REGISTER SELECT	1	0	0	0	0	RS_0	RS_1	RS_2	0	1	1
WRITE TO VRAM											
BYTE 1 ADDRESS SET UP	A_6	A_7	A_8	A_9	A_{10}	A_{11}	A_{12}	A_{13}	0	1	1
BYTE 2 ADDRESS SET UP	0	1	A_0	A_1	A_2	A_3	A_4	A_5	0	1	1
BYTE 3 DATA WRITE	D_0	D_1	D_2	D_3	D_4	D_5	D_6	D_7	0	1	0
READ FROM VDP REGISTER											
BYTE 1 DATA READ	D_0	D_1	D_2	D_3	D_4	D_5	D_6	D_7	1	0	1
READ FROM VRAM											
BYTE 1 ADDRESS SET UP	A_6	A_7	A_8	A_9	A_{10}	A_{11}	A_{12}	A_{13}	0	1	1
BYTE 2 ADDRESS SET UP	0	0	A_0	A_1	A_2	A_3	A_4	A_5	0	1	1
BYTE 3 DATA READ	D_0	D_1	D_2	D_3	D_4	D_5	D_6	D_7	1	0	0

2.1.4 CPU Read from VDP Status Register

The CPU can read the contents of the status register with a single-byte transfer. MODE is high for the transfer. $\overline{CSR}$ is used to signal the VDP that a read operation is required.

2.1.5 CPU Read from VRAM

The CPU reads data from the VRAM through the VDP using the autoincrementing address register. A one-byte transfer is then required to read the data from the addressed VRAM byte. The address register is then autoincremented. Sequential VRAM data reads require only a one-byte transfer since the address register is already set up. During setup of the address register, the two most-significant bits of the second address byte must be '0's. By setting up the address this way, a read cycle to VRAM is initiated and read data will be available for the first data transfer to the CPU. (see Table 1). MODE is high for the address byte transfers and low for the data transfers. The VDP requires approximately 8 microseconds to fetch the VRAM byte following a data transfer and 3 microseconds following address setup.

2.1.6 VDP Interrupt

The VDP $\overline{INT}$ output pin is used to generate an interrupt at the end of each active-display scan, which is about every 1/60 second (color burst frequency/60, 192). The $\overline{INT}$ output is active when the interrupt Enable bit (IE) in VPD register 1 is a '1' and the F bit of the status register is a '1'. Interrupts are cleared when the status register is read.

2.1.7 VDP Initialization

The VDP is externally initialized whenever the $\overline{RESET}$ input is active (low) and must be held low for a minimum of 3 microseconds. The external reset synchronizes all clocks with its falling edge, sets the horizontal and vertical counters to known states, and clears VDP registers 0 and 1. The video display is automatically blanked since the BLANK bit in VDP register 1 becomes a '0'. The VDP, however, continues to refresh the VRAM even though the display is blanked. While the $\overline{RESET}$ line is active, the VDP does not refresh VRAM.

2.2 VDP/VRAM INTERFACE

The VDP can access up to 16,384 bytes of VRAM using a 14-bit VRAM address. The VDP fetches data from the VRAM in order to process the video image as described later. The VDP also stores data in or reads in data from the VRAM during a CPU-VRAM data transfer. The VDP automatically refreshes the VRAM.

2.2.1 VRAM Interface Control Signals

The VDP-VRAM interface consists of two unidirectional 8-bit data buses and three control lines as shown in Figure 2-3. The VRAM outputs data to the VDP on the VRAM read data bus (RD0-RD7). The VDP outputs both the address and data to the VRAM over the VRAM address/data bus (AD0-AD7). The VRAM row address is output when $\overline{RAS}$ is active (low). The columrf address is output when $\overline{CAS}$ is active (low). Data is output to the VRAM when $\overline{R/W}$ is active (low).

2.2.2 VRAM Memory Types

The VDP can utilize 4027-type 4K, 4108-type 8K, or 4116-type 16K dynamic RAMs. The 4/16K bit in VDP register 1 is a '0' for 4027 type RAMs and a '1' for 4108- and 4116-type RAMs. Note that there is a minor difference between the way 4027's and 4108's/4116's are wired to the VDP. In the 4027, all $\overline{CE}$ pins are tied to ground. In the 4108/4116 the A6 lines on the 4116 and 4108 (the same pin as $\overline{CE}$ on 4027's) are all tied to AD1 on the TMS 9918A. A jumper can be used to select the VRAM-type.

2.3 VDP/TV INTERFACE

The composite video output signal coming from the VDP drives an NTSC color monitor. This signal incorporates all necessary horizontal and vertical synchronization signals as well as luminance and chrominance information. monitor applications, the requirements of the monitor should be studied to determine if the VDP can be conr directly to it. The internal output buffer device on the composite video pin is a source-follower MOS transistor that requires an external pull-down reisstor to V_{SS} as shown in Figure 2-4. Typically a 1 kilohm resistor is recommended to provide a 1.9-volt synchronization level.

In some cases, it may be necessary to provide a simple interface circuit to match the VDP output voltages with the monitor specifications. To drive a standard television that is not outfitted with a composite video input, the signal can be run into the television antenna terminals by using an appropriate RF modulator on the VDP output. Care should be taken to ensure proper matchup between VDP, RF modulator, and TV.

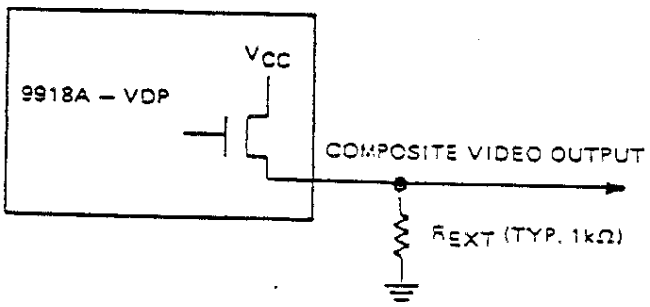

FIGURE 2-4 – COMPOSITE VIDEO PULL-DOWN CIRCUIT

2.4 WRITE-ONLY REGISTERS

The eight VDP write-only registers are shown in Figure 2-5. They are loaded by the CPU as described in Section 2.1.2. Registers 0 and 1 contain flags to enable or disable various VDP features and modes Registers 2 through 6 contain values that specify starting locations of various sub-blocks of VRAM. The definitions of these sub-blocks are described in Section 2.7. Register 7 is used to define backdrop and text colors.

The following is a description of each register:

2.4.1 Register 0

Register 0 contains two VDP option control bits. All other bits are reserved for future use and must be '0's.

BIT 6 M3 (mode bit 3) See Section 2.4.2 for table and description.

BIT 7 External Video enable/disable

'1' enables external video input
'0' disables external video input

2.4.2 Register 1

Register 1 contains 8 VDP option control bits.

BIT 0 4/16K selection

'0' selects 4027 RAM operation
'1' selects 4108/4116 RAM operation

BIT 1 BLANK enable/disable

'0' causes the active display area to blank
'1' enables the active display
Blanking causes the display to show border color only

BIT 2 IE (Interrupt Enable)

'0' disable VDP interrupt
'1' enable VDP interrupt

BIT 3,4 M1, M2 (mode bits 1 and 2)
M1, M2 and M3 determine the operating mode of the VDP:

M1	M2	M3	
0	0	0	Graphics I mode
0	0	1	Graphics II mode
0	1	0	Multicolor mode
1	0	0	Text mode

BIT 5 Reserved

BIT 6 Size (sprite size select)

'0' selects Size 0 sprites (8 X 8 bit)
'1' selects Size 1 sprites (16 X 16 bits)

BIT 7 MAG (Magnification option for sprites)

'0' selects MAG0 sprites (1X)
'1' selects MAG1 sprites (2X)

2.4.3 Register 2

Register 2 defines the base address of the Name Table sub-block. The range on its contents is from 0 to 15. The con tents of the register form the upper 4 bits of the 14-bit Name Table addresses; thus the Name Table base address equal to (register 2) * 400 (hex).

2.4.4 Register 3

Register 3 defines the base address of the Color Table sub-block. The range on its contents is from 0 to 255. The con tents of the register form the upper 8 bits of the 14-bit Color Table addresses; thus the Color Table base address equal to (register 3) * 40 (hex).

2.4.5 Register 4

Register 4 defines the base address of the Pattern, Text or Multicolor Generator sub-block. The range of its content is 0 through 7. The contents of the register form the upper 3 bits of the 14-bit Generator addresses; thus the Generato base address is equal to (register 4) * 800 (hex).

2.4.6 Register 5

Register 5 defines the base address of the Sprite Attribute Table sub-block. The range of its contents is from through 127. The contents of the register form the upper 7 bits of the 14-bit Sprite Attribute Table addresses; thu the base address is equal to (register 5) * 80 (hex).

2.4.7 Register 6

Register 6 defines the base address of the Sprite Pattern Generator sub-block. The range of its contents is 0 throug 7. The contents of the register form the upper 3 bits of the 14-bit Sprite Pattern Generator addresses; thus th Sprite Pattern Generator base address is equal to (register 6) * 800 (hex).

2.4 Register 7

The upper 4 bits of register 7 contain the color code of color 1 in the Text mode. The lower 4 bits contain the color code for color 0 in the Text mode and the backdrop color in all modes. See Table 3 for color codes.

2.5 STATUS REGISTER

The VDP has a single 8-bit status register that can be accessed by the CPU. The status register contains the interrupt pending flag, the sprite coincidence flag, the fifth sprite flag, and the fifth sprite number, if one exists. The format of the status register is shown in Figure 2-5. A discussion of the contents follows.

The status register may be read at any time to test the F, C, and 5S status bits. Reading the status register will clear the interrupt flag, F. Asynchronous reads will, however, cause the frame flag (F) bit to be reset and therefore missed. Consequently, the status register should be read only when the VDP interrupt is pending.

2.5.1 Interrupt Flag (F)

The F status flag in the status register is set to '1' at the end of the raster scan of the last line of the active display. It is reset to a '0' after the status register is read or when the VDP is externally reset. If the Interrupt Enable bit in VDP register 1 is active ('1'), the VDP interrupt output ($\overline{INT}$) will be active (low) whenever the F status flag is a '1'.

2.5.2 Coincidence Flag (C)

The C status flag in the status register is set to a '1' if two or more sprites "coincide". Coincidence occurs if any two sprites on the screen have one or more overlapping pixels. Transparent colored sprites, as well as those that are partially or completely off the screen, are also considered. Sprites beyond the Sprite Attribute Table terminator ($D0_{16}$) are not considered. The 'C' flag is cleared to a '0' after the status register is read or the VDP is externally reset.

2.5.3 Fifth Sprite Flag (5S) and Number

The 5S status flag in the status register is set to a '1' whenever there are five or more sprites on a horizontal line (lines 0 to 192) and the frame flag is equal to a '0'. The 5S status flag is cleared to a '0' after the status register is read or the VDP is externally reset. The number of the fifth sprite is placed into the lower 5 bits of the status register when the 5S flag is set and is valid whenever the 5S flag is '1'. The setting of the fifth sprite flag will not cause an interrupt.

2.6 OSCILLATOR AND CLOCK GENERATION

The VDP is designed to operate with a 10.738635 megahertz ±50 ppm crystal input to generate the required internal clock signals. A fundamental frequency, parallel-mode crystal is used as the frequency reference for the internal clock oscillator, which is the master time base for all system operations. This master clock is divided by two to generate the pixel clock (5.3 megahertz) and by three to provide the CPUCLK (3.58 megahertz). The GROMCLK is developed from the master clock frequency divided by 24.

2.6.1 Color Phase Generation

The 10.7+ megahertz master clock and its complement are used to generate an internal six-phase 3.579545 megahertz (±10 hertz) clock to provide the video color signals and the color burst reference for use in developing the composite video output signal. While the VDP signals are not exact equivalents to the standard NTSC colors, the differences can easily be adjusted with the color and tint controls of the target color television.

2.6.2 Video Sync and Control Generation

The vertical and horizontal control signals are generated by decoding the outputs of the horizontal and vertical counters. The horizontal counter is driven by the pixel clock, and the horizontal counter in turn increments the vertical counter. Table 2 gives the relative count values of the screen display parameters. Within the active display area during Graphics I mode, the 3 least-significant bits of the horizontal counter address the individual picture element of each pattern displayed. Also, during the vertical active display period, the 3 least-significant bits of the vertical counter address each individual line in the 8 X 8 patterns. The Graphics II, Multicolor and Text modes use the counters similarly.

The VDP operates at 262 lines per frame and approximately 60 frames per second in a non-interlaced mode of operation.

TABLE 2 – SCREEN DISPLAY PARAMETERS

PARAMETER	PIXEL CLOCK CYCLES	
HORIZONTAL	PATTERN OR MULTICOLOR	TEXT
HORIZONTAL ACTIVE DISPLAY	256	240
RIGHT BORDER	15	25
RIGHT BLANKING	8	8
HORIZONTAL SYNC	26	26
LEFT BLANKING	2	2
COLOR BURST	14	14
LEFT BLANKING	8	8
LEFT BORDER	13	19
	342	342
VERTICAL	LINE	
VERTICAL ACTIVE DISPLAY	192	
BOTTOM BORDER	24	
BOTTOM BLANKING	3	
VERTICAL SYNC	3	
TOP BLANKING	13	
TOP BORDER	27	
	262	

2.7 VIDEO DISPLAY MODES

The VDP displays an image on the screen that can best be envisioned as a set of display planes sandwiched together. Figure 2-6a shows the definition of each of the planes. Objects on planes closest to the viewer have higher priority. In cases where two entities on two different planes are occupying the same spot on the screen, the entity on the higher priority plane will show at that point. For an entity on a specific plane to show through, all planes in front of that plane must be transparent at that point. The first 32 planes (Figure 2-6b) each may contain a single sprite. (Sprites are pattern objects whose positions on the screen are defined by horizontal and vertical coordinates in VRAM.) The areas of the Sprite Planes, outside of the sprite itself, are transparent. Since the coordinates of the sprite are in terms of pixels, the sprite can be positioned and moved about very accurately. Sprites are available in three sizes: 8 X 8 pixels, 16 X 16 pixels, and 32 X 32 pixels. Behind the Sprite Plane is the Pattern Plane. The Pattern Plane is used for textual and graphics images generated by the Text, Graphics I, Graphics II, or Multicolor modes. Behind the Pattern Plane is the backdrop, which is larger in area than the other planes so that it forms a border around the other planes. The last and lowest priority plane is the External Video Plane. Its image is defined by the external video input pin. The backdrop consists of a single color used for the display borders and as the default color for the active display area. The default color is stored in the VDP register 7. When the backdrop color register contains the transparent code, the backdrop automatically defaults to black if the external video mode is not selected.

The 32 Sprite Planes are used for the 32 sprites in the Multicolor and Graphics modes. They are not used in the Text mode and are automatically transparent. Each of the sprites can cover an 8 X 8, 16 X 16, or 32 X 32 pixel area on its plane. Any part of the plane not covered by the sprite is transparent. All or part of each sprite may also be transparent. Sprite 0 is on the outside or highest plane, and sprite 31 is on the plane immediately adjacent to Pattern Plane. Whenever a pixel in a Sprite Plane is transparent, the color of the next plane can be seen through that plane. If, however, the sprite pixel is non-transparent, the colors of the lower planes are automatically replaced by the sprite color. There is also a restriction on the number of sprites on a line. Only four sprites can be active on any horizontal line. Additional sprites on a line will be automatically made transparent for that line. Only those sprites that are active on the display will cause the coincidence flag to set. The VDP status register provides a flag bit and the number of the fifth sprite whenever this occurs. The Pattern Plane is used in the Text, Multicolor, and Graphics modes for display of the graphic patterns of characters. Whenever a pixel on the Pattern Plane is non-transparent, the backdrop color is automatically replaced by the Pattern Plane color. When a pixel in the Pattern Plane is transparent, the backdrop color can be seen through the Pattern Plane.

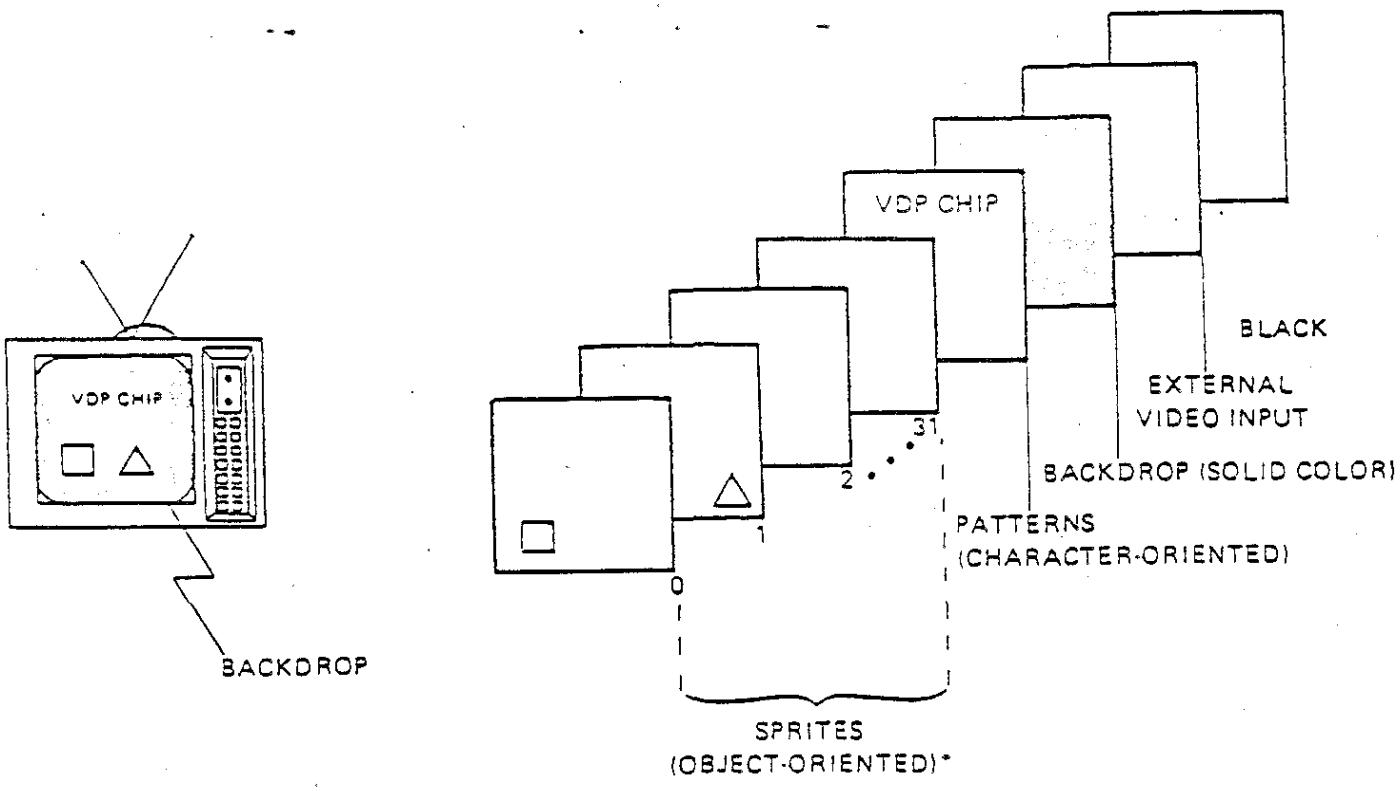

FIGURE 2-6a — VDP DISPLAY PLANES

The VDP has four video color display modes that appear on the Pattern Plane: Graphics I mode, Graphics II mode, Text mode, and Multicolor mode. Graphics I and Graphics II modes cause the Pattern Plane to be broken up into groups of 8 X 8 pixels, called pattern positions. Since the full image is 256 X 192 pixels, there are 32 X 24 pattern positions on the screen in the graphics modes. In Graphics I mode, 256 possible patterns may be defined for the 768 pattern positions with two unique colors allowed for each pattern definition. Graphics II mode provides, through a unique mapping scheme, 768 pattern definitions for the 768 pattern positions. Graphics II mode also allows the selection of two unique colors for each line of a pattern definition. Thus, all 15 colors plus transparent may be used in a single pattern position. In Text mode, the Pattern Plane is broken into groups of 6 X 8 pixels, called text positions. There are 40 X 24 text positions on the screen in this mode. In Text mode, sprites do not appear on the screen and two colors are defined for the entire screen. In Multicolor mode, the screen is broken into a grid of 64 X 48 positions, each of which is a 4 X 4 pixel. Within each position, one unique color is allowed.

The VDP registers define the base addresses for several sub-blocks within VRAM. These sub-blocks form tables which are used to produce the desired image on the TV screen. The Pattern Name Table, the Pattern Generator Table and the Sprite Generator Table are used to form the sprites. The contents of these tables must all be provided by the microprocessor. Animation is achieved by altering the contents of VRAM in real time.

The VDP can display the 15 colors shown in Table 3. The VDP colors also provide eight different gray levels for displays on monochrome televisions; the luminance values in the table indicate these levels, 0.00 being black and 1.00 being white. Whenever all planes are of the transparent color at a given point, the color shown at that point will be black.

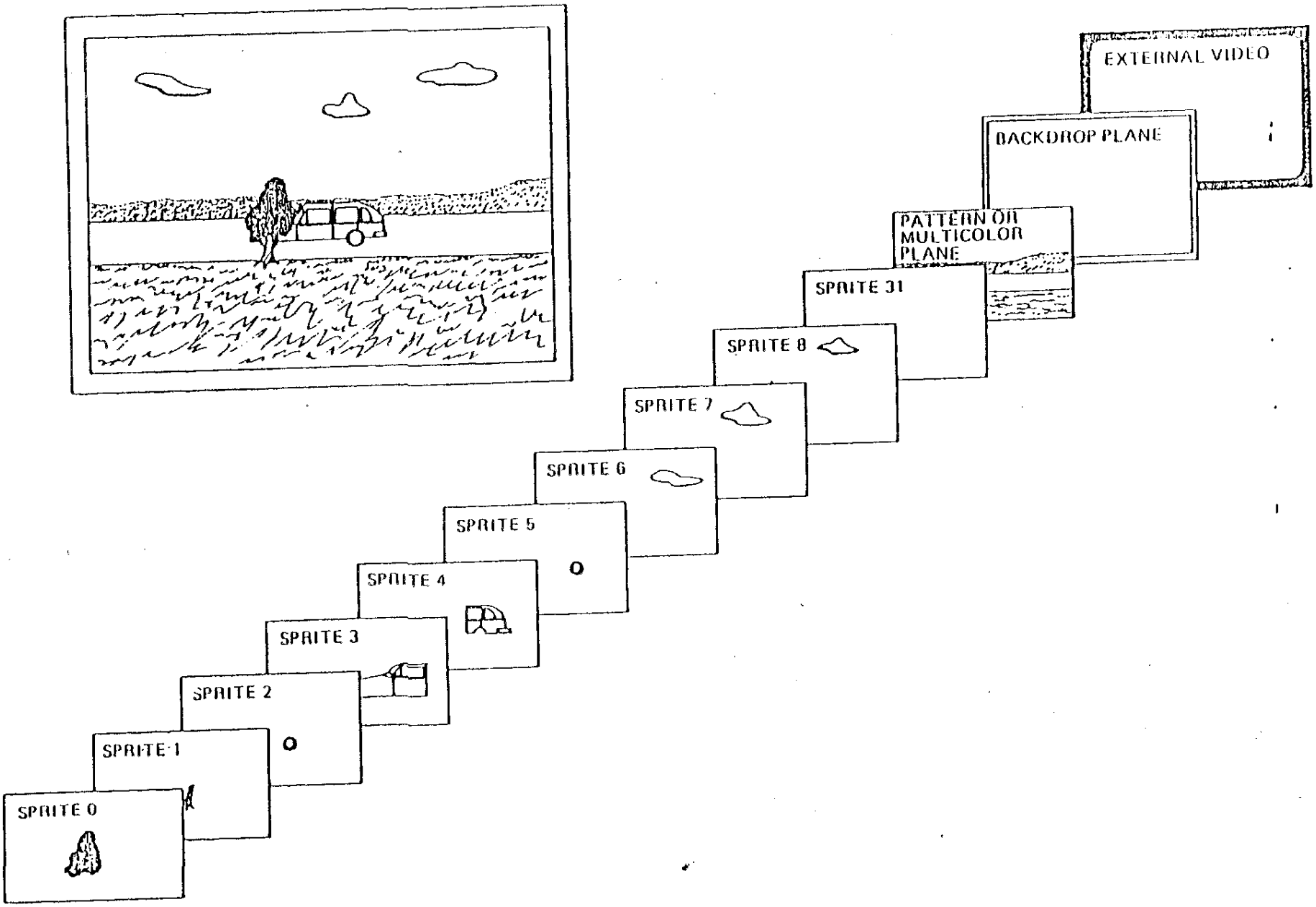

FIGURE 2-6b – VDP DISPLAY PLANES

TABLE 3. COLOR ASSIGNMENTS

COLOR (HEX)	COLOR	LUMINANCE (DC VALUE)	CHROMINANCE (AC VALUE)
0	TRANSPARENT	0.00	—
1	BLACK	0.00	—
2	MEDIUM GREEN	.60	.60
3	LIGHT GREEN	.80	.53
4	DARK BLUE	.47	.73
5	LIGHT BLUE	.67	.60
6	DARK RED	.53	.53
7	CYAN TURQUOISE	.80	.73
8	MEDIUM RED	.67	.73
9	LIGHT RED	.80	.73
A	DARK YELLOW	.87	.53
B	LIGHT YELLOW	1.00	.40
C	DARK GREEN	.47	.60
D	MAGENTA purple	.60	.47
E	GRAY		—
F	WHITE	1.00	—
—	BLACK LEVEL	0.00	—
—	COLOR BURST	0.00	.40
—	SYNC LEVEL	−0.40	—

2.7.1 Graphics I Mode

The VDP is in Graphics I mode when M1, M2, and M3 bits in VDP registers 1 and 0 are zero. In Graphics I mode the Pattern Plane is divided into a grid of 32 columns by 24 rows of pattern positions (see Figure 2-7). Each of the pattern positions contains 8 X 8 pixels. The tables in VRAM used to generate the Pattern Plane are the Pattern Color Table. Figure 2-8 illustrates the mapping of these tables into the Pattern Plane. A total of 2848 VRAM bytes are required for the Pattern Name, Color and Generator tables. Less memory is required if all 256 possible pattern definitions are not required. The tables can be overlapped to reduce the amount of VRAM needed for pattern generation. Examples of VRAM memory allocation are provided in Section 3.

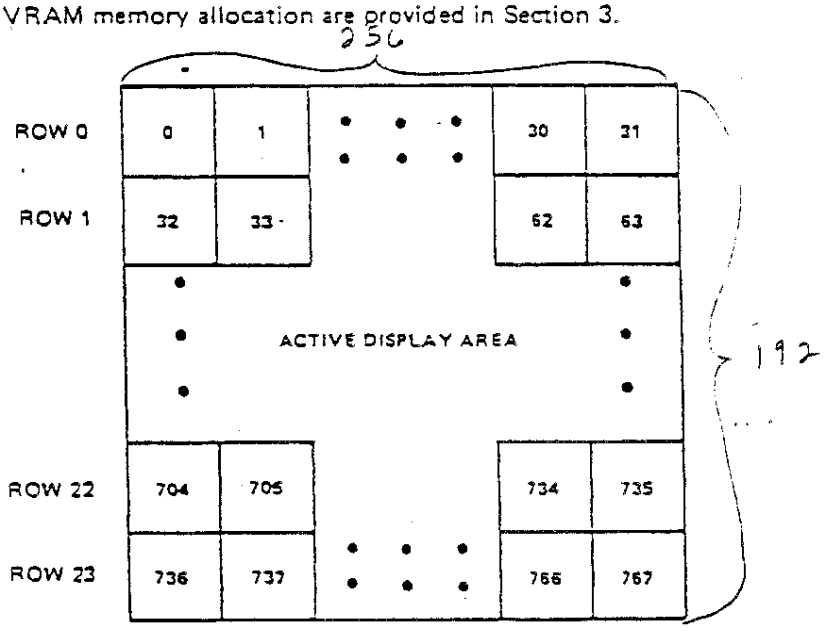

FIGURE 2-7 – PATTERN GRAPHICS NAME TABLE MAPPING

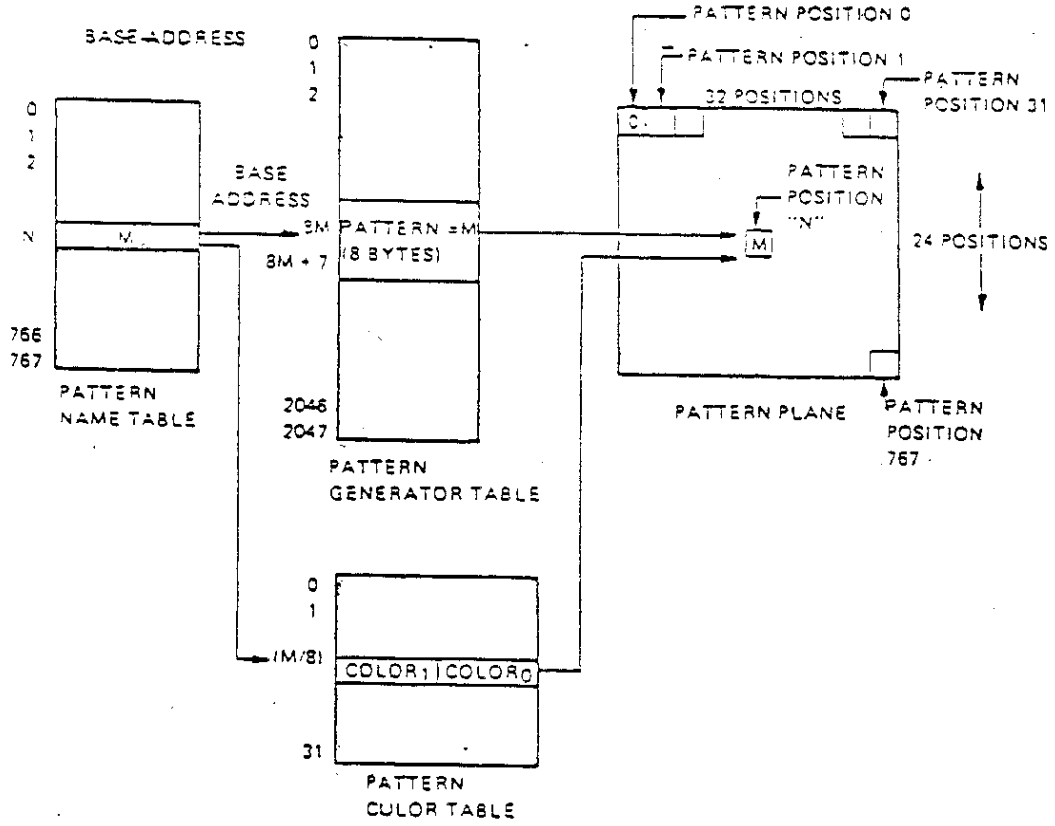

FIGURE 2-8 – PATTERN MODE MAPPING

The Pattern Generator Table contains a library of patterns that can be displayed in the pattern positions. It is 2048 bytes long, and is arranged into 256 patterns, each of which is eight bytes long, yielding 8 X 8 bits. All of the '1's in the eight-byte pattern can designate one color (color 1), while all the '0's can designate another color (color 0).

The full 8-bit pattern name is used to select one of the 256 pattern definitions in the Pattern Generator Table. The table is a 2048-byte block in VRAM beginning on a 2 kilobyte boundary. The starting address of the table is determined by the generator base address in VDP register 4. The base address forms the three most-significant bits of the 14-bit VRAM address for each Pattern Generator Tableentry. The next 8 bits indicate the 8-bit name of the selected pattern definition. The lowest 3 bits of the VRAM address indicate the row number within the pattern definition.

Eight bytes are required for each of the 256 possible unique 8 X 8 pattern definitions. The first byte defines the first row of the pattern, and the second byte defines the second row. The first bit of each of the eight bytes define the first column of the pattern. The remaining rows and columns are similarily defined. Each bit entry in the pattern definition selects one of the two colors for that pattern. A '1' bit selects the color code (color 1) contained in the most-significant four bits of the corresponding color table byte. A '0' bit selects the other color code (color 0). An example of pattern definition mapping is provided in Figure 2-9.

	PATTERN						PATTERN DEFINITION							
0		C	C	C	C	C	0	1	1	1	1	1	0	0
1						C	0	0	0	0	0	1	0	0
2						C	0	0	0	0	0	1	0	0
3			C	C	C	C	0	0	1	1	1	1	0	0
4						C	0	0	0	0	0	1	0	0
5						C	0	0	0	0	0	1	0	0
6		C	C	C	C	C	0	1	1	1	1	1	0	0
7							0	0	0	0	0	0	0	0
ROW/BYTE	COLUMN						BIT							
	0	1	2	3	4	5	0	1	2	3	4	5	6	7

NOTES: VDP register 7 entry: 71_{16}.
Color code 7 is cyan (signified above by 'C').
Color code 1 is black (signified above by a space).
Bit 0 is the most significant bit of each data byte.

FIGURE 2-9 — PATTERN DISPLAY MAPPING

The color of the '1's and '0's is defined by the Pattern Color Table that contains 32 entries each of which is one b long. Each entry defines two colors: the most-significant 4 bits of each entry define the color of the '1's, and the least-signigicant 4 bits define the color of the '0's. The first entry in the color table defines the colors for patterns 0 to 7; the next entry for patterns 8 to 15, and so on. (See Table 4 for assignments.) Thus, 32 different pairs of colors may be displayed simultaneously.

The Pattern Name Table is located in a contiguous 768-byte block in VRAM beginning on a 1 kilobyte boundary. The starting address of the Name Table is determined by the 4-bit Name Table base address field in VDP register 2. The base address forms the upper four bits of the 14-bit VRAM address. The lower 10 bits of the VRAM address are formed from the row and column counters. An example of pattern name table addressing is given in Section 3.

Each byte entry in the Name Table is the name of or the pointer to a pattern definition in the Pattern Generator Table. The upper five bits of the eight-bit name identify the color group of the pattern. There are 32 groups of eight patterns. The same two colors are used for all eight patterns in a group; the color codes are stored in the VDP Color Table. The Color Table is located in a 32-byte block in VRAM beginning on a 64-byte boundary. The table starting address is determined by the 8-bit Color Table base address in VDP register 3. The base address forms the upper eight bits of the 14-bit Color Table entry VRAM address. The next bit is a '0' and the lowest 5 bits are equal to the upper 5 bits of the corresponding Name Table entries. An example of Color Table addressing is provided in Section 3.

Since the tables in VRAM have their base addresses defined by the VDP registers, a complete switch of the values in the tables can be made by simply changing the values in the VDP registers. This is especially useful when one wishes to time-slice between two or more screens of graphics.

When the Pattern Generator Table is loaded with a pattern set, manipulation of the Pattern Name Table contents can change the appearance of the screen. Alternatively, a dynamically changing set of patterns throughout the course of a graphics session is easily accomplished since all tables are in VRAM.

For textual applications, the desired character set is typically loaded into the Pattern Generator first. The official USASCII character set might be loaded into the Pattern Generator in such a way that the pattern numbers correspond to the 8-bit ASCII codes for that pattern; e.g., the pattern for the letter "A" would be loaded into pattern number 41hex in the Pattern Generator. Next the Pattern Color Table would be loaded up with the proper color set. To print a textual message on the screen, write the proper ASCII codes out to the Pattern Name Table.

Images can be formed using the Pattern Plane. To display an object of size 8 X 8 pixels or smaller, only one pattern would need to be defined. To display a larger figure, the figure should be broken up into smaller 8 X 8 squares. Then multiple patterns can be defined, and the Pattern Generator and Pattern Name Table set up appropriately. Note that rough motion of objects requires merely updating entries in the Pattern Name Table.

TABLE 4. PATTERN COLOR TABLE

Byte No.	Pattern No.
0	0..7
1	8..15
2	16..23
3	24..31
4	32..39
5	40..47
6	48..55
7	56..63
8	64..71
9	72..79
10	80..87
11	88..95
12	96..103
13	104..111
14	112..119
15	120..127
16	128..135
17	136..143
18	144..151
19	152..159
20	160..167
21	168..175
22	176..183
23	184..191
24	192..199
25	280..207
26	208..215
27	216..223
28	224..231
29	232..239
30	240..247
31	248-255

A total of 2848 VRAM bytes are required for the Pattern, Name, Color and Generator tables. Less memory is needed if all 256 possible pattern definitions are not required; the tables can be overlapped to reduce the amount of VRAM needed for pattern generation. Examples of VRAM memory allocation are provided in Section 3.

2.7.2 Graphics II Mode

The VDP is in the Graphics II mode when mode bits M1 = 0, M2 = 0, and M3 = 1. The Graphics II mode is similar to Graphics I mode except it allows a larger library of patterns so that a unique pattern generator entry may be made for each of the 768 (32 X 24) pattern positions on the video screen. Additionally, more color information is included in each 8 X 8 graphics pattern. Thus two unique colors may be specified for each byte of the 8 X 8 pattern larger amount of VRAM (12 kilobytes) is required to implement the full usage of the Graphics II mode.

Like Graphics I mode, the Graphics II mode Pattern Name Table contains 768 entries which correspond to the 768 pattern positions on the display screen. Because the Graphics I mode pattern names are only 8 bits in length, a maximum of 256 pattern definitions may be addressed using the addressing scheme discussed in the previous section. Graphics II mode, however, segments the display screen into three equal parts of 256 pattern positions each and also segments the Pattern Generator Table into three equal blocks of 2048 bytes each. Pattern definitions in the first third correspond to pattern positions in the upper third of the display screen. Likewise pattern definitions in the second and third blocks of the Pattern Generator Table correspond to the second and third areas of the Pattern Plane. The pattern Name Table is also segmented into three blocks of 256 names each so that names found in the upper third, reference pattern definitions found in the upper 2048 bytes in Pattern Generator Table. Likewise the second and third blocks reference pattern definitions in the second 2048 byte block and third 2048 byte block respectively. Thus, if 768 patterns are uniquely specified, an 8-bit pattern name will be used three times, once in each segment of the Pattern Name Table. The Pattern Generator Table falls on eight kilobyte boundaries and may be located in the upper or lower half of 16K memory based on the MSB of the pattern generator base in VDP register 4. The LSB's must be set to all '1's.

The Color Table is also 6144 bytes long and is segmented into three equal blocks of 2048 bytes. Each entry in the Pattern Color Table is eight bytes which provides the capability to uniquely specify color 1 and color 0 for each of the eight bytes of the corresponding pattern definition. The addressing scheme is exactly like that of the Pattern Generator Table except for the location of the table in VRAM. This is controlled by the loading of the MSB of the color base in VDP register 3. The LSB's must be set to all '1's.

Figure 2-10 is an example of the Graphics II mode mapping scheme. Note that pattern names P1, P2, P3 correspond to pattern generator entries in the three blocks of the Pattern Generator Table. Note also how these three nam map to the display screen. Figure 2-11 an example of a Pattern Generator and Pattern Color Table entry.

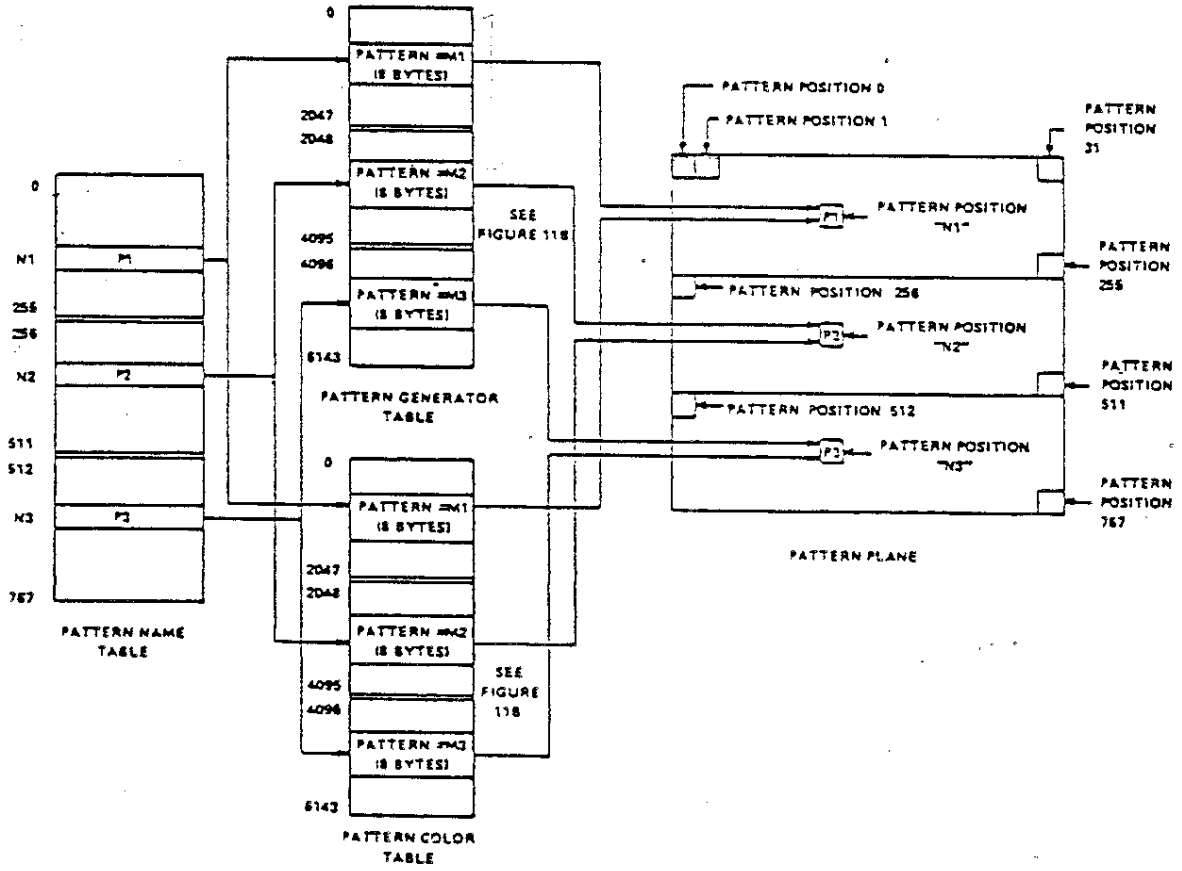

FIGURE 2-10 – GRAPHICS II MODE MAPPING

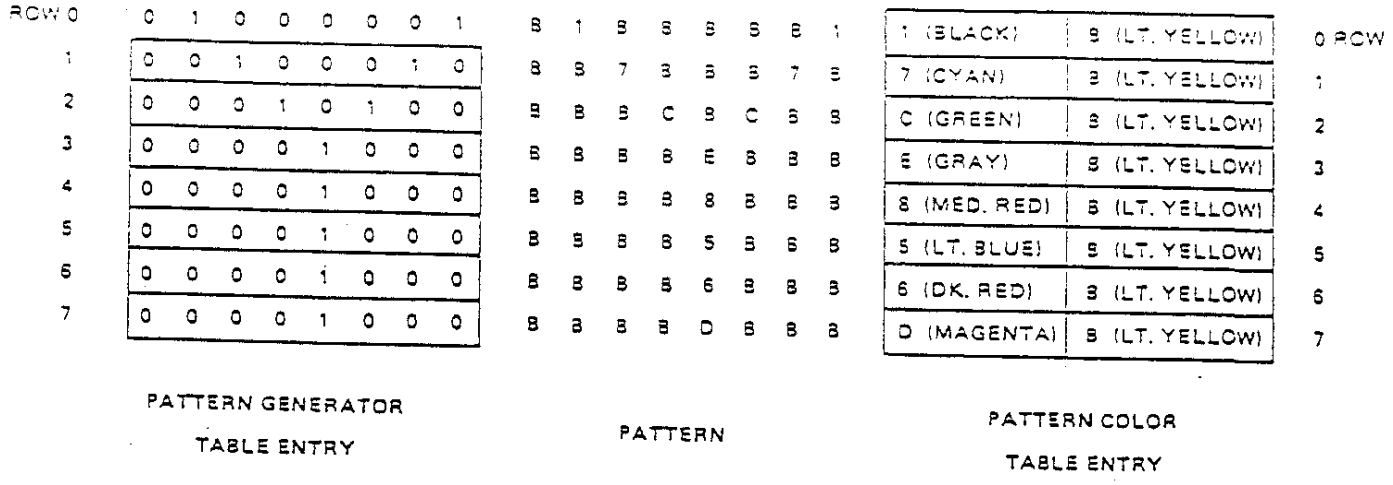

FIGURE 2-11 – PATTERN DISPLAY MAPPING

2.7.3 Multicolor Mode

The VDP is in Multicolor mode when mode bits M1 = 0, M2 = 1, and M3 = 0. Multicolor mode provides an unrestricted 64 X 48 color square display. Each color square contains a 4 X 4 block of pixels. The color of each of the color squares can be any one of the 15 video display colors plus transparent. Consequently, all 15 colors can be used simultaneously in the Multicolor mode. The Backdrop and Sprite Planes are still active in the Multicolor mode.

The Multicolor Name Table is the same as that for the graphics modes, consisting of 768 name entries. The name no longer points to a color list; rather color is now derived from the Pattern Generator Table. The name points to an eight-byte segment of VRAM in the Pattern Generator Table.

Only two bytes of the eight-byte segment are used to specify the screen image. These two bytes specify four colors, each color occupying a 4 X 4 pixel area. The four MSB's of the first byte define the color of the upper left quarter of the multicolor pattern; the LSB's define the color of the upper right quarter. The second byte similarly defines the lower left and right quarters of the multicolor pattern. The two bytes thus map into a 8 X 8 pixel multicolor pattern. (See Figure 2-12).

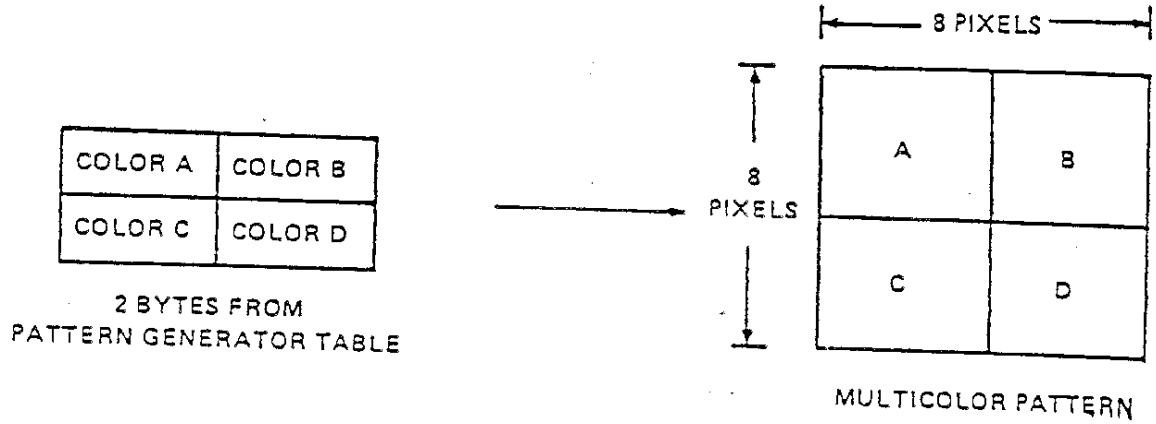

FIGURE 2-12 – MULTICOLOR LIST MAPPING

The location of the two bytes within the eight-byte segment pointed to by the name is dependent upon the screen position where the name is mapped. For names in the top row (names 0-31), the two bytes are the first two within the groups of eight-byte segments pointed to by the names. The next row of names (32-63) uses the third and fourth bytes within the eight-byte segments. The next row of names uses the fifth and sixth bytes while the last row of names uses the seventh and eighth. This series repeats for the remainder of the screen.

For example, referring to Figure 2-13, if Name Table entry 0 (pattern position 0) multicolor block =N (name = N), the multicolor pattern displayed will be an 8 X 8 pixel block consisting of colors A, B, C, D comprising the first two bytes of the Multicolor Table. If, however, name =N is located in Name Table entry 33, (Pattern position 33), the colors displayed will be colors E, F, G, H as specified by the third and fourth bytes of the multicolor block pointed to by the name. Likewise pattern positions which lie in rows 2 and 3 would cause colors I, J, K, L and colors M, N, O, P respectively to be displayed. Thus it can be seen that the color displayed from the multicolor generator block is dependent upon pattern position on the screen. Figure 2-14 illustrates the multicolor mode mapping scheme.

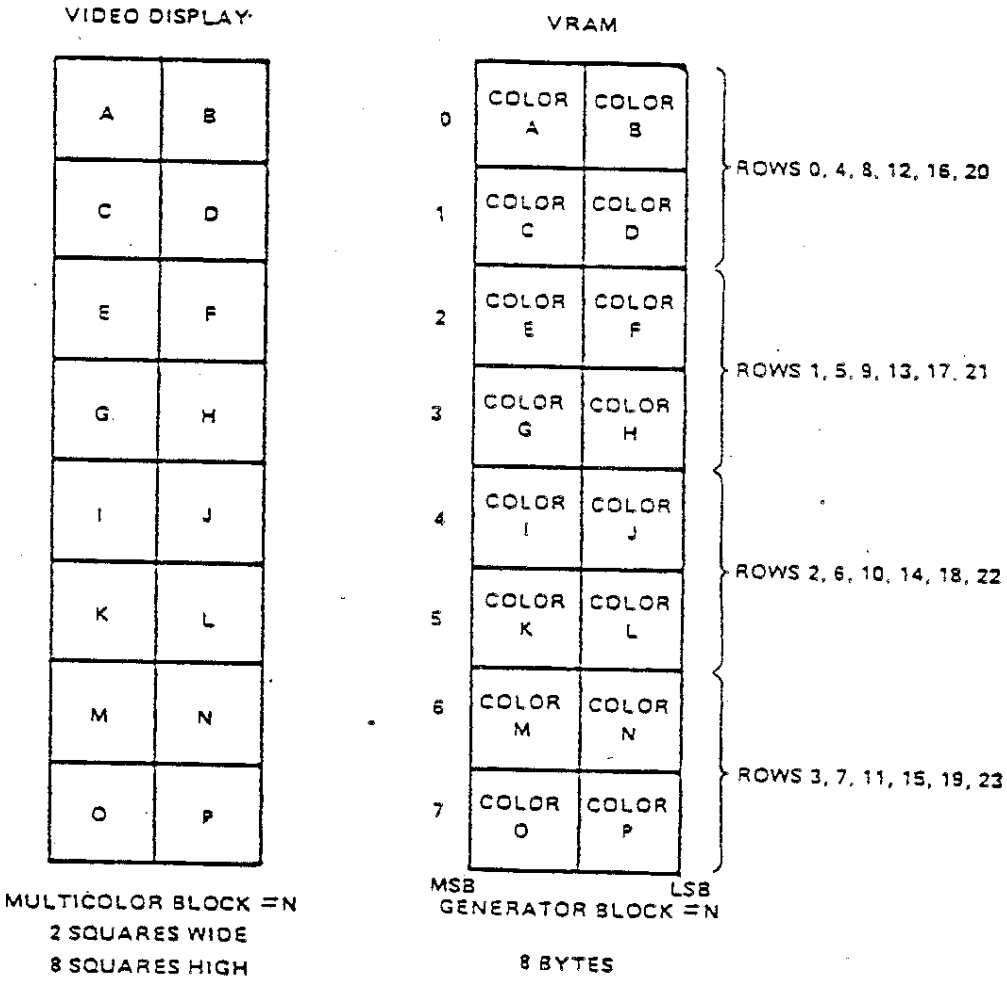

FIGURE 2-13 — MULTICOLOR BLOCK DISPLAY

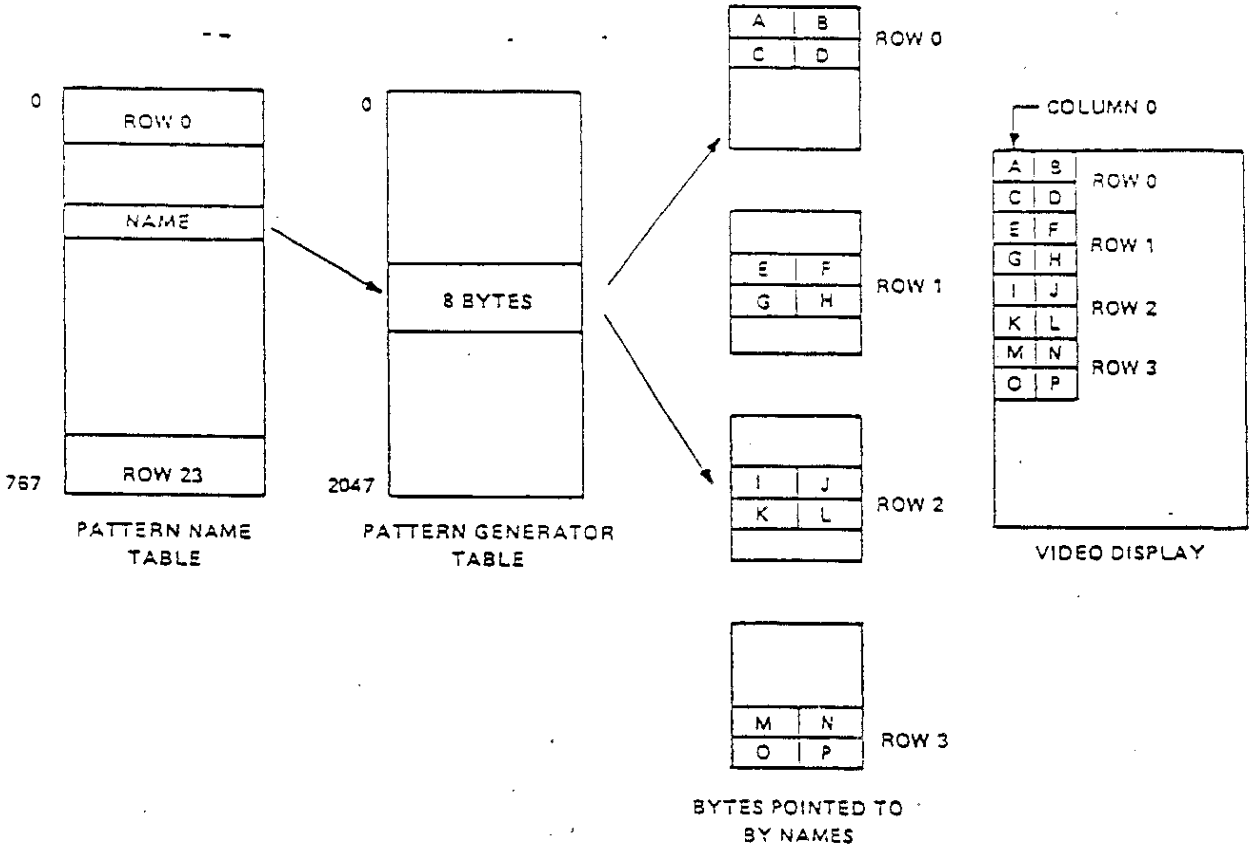

FIGURE 2-14 – MULTICOLOR MODE MAPPING

The mapping of VRAM contents to screen image is simplified by using duplicate names in the Name Table. Since the series of bytes used within the eight-byte segment repeats every four rows, the four rows in the same column can use the same name. Then the eight-byte segment specifies a 2 X 8 color square pattern on the screen as a straightforward translation from the eight-byte segment in VRAM pointed to by the common name.

When used in this manner, 768 bytes are still used for the Name Table and 1536 bytes are used for the color information in the Pattern Generator Table (24 rows X 32 columns X 8 bytes/pattern position). Thus a total of 1728 bytes in VRAM are required. It should be noted that the tables begin on even 1K and 2K boundaries and are therefore not contiguous. An example of multicolor VRAM memory allocation is provided in Section 3.

2.7.4 Text Mode

The VDP is in Text mode when mode bits M1 = 1, M2 = 0, and M3 = 0. In the Text mode, the screen is divided into a grid of 40 text positions across and 24 down. (See Figure 2-15). Each of the text positions contains six pixels across and eight pixels down. The tables used to generate the Pattern Plane are the Pattern Name Table and the Pattern Generator Table. There can be up to 256 unique patterns defined at any time. The pattern definitions are stored in the Pattern Generator Table in VRAM and can be dynamically changed. The VRAM contains a Pattern Name Table which maps the pattern definitions into each of the 960 pattern cells on the Pattern Plane (Figure 2-16). Sprites are not available in Text mode.

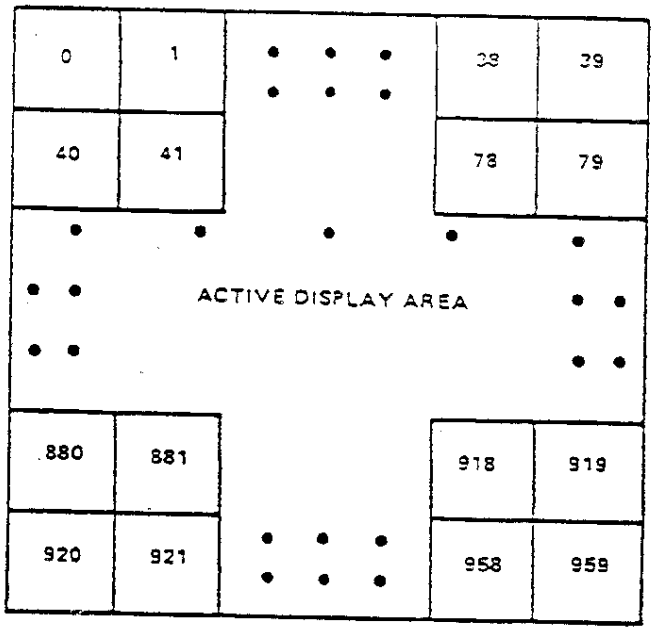

FIGURE 2-15 – TEXT MODE NAME TABLE PATTERN POSITIONS

As in the case of the Graphics modes, the Pattern Generator Table contains a library of text patterns that can be displayed in the text positions. It is 2048 bytes long, and is arranged in 256 text patterns, each of which is eight bytes long. Since each text position on the screen is only six pixels across, the least-significant 2 bits of each text pattern are ignored, yielding 6 X 8 bits in each text pattern. Each block of eight bytes defines a text pattern in which all the '1's in the text pattern take on one color when displayed on the screen, while all the '0's take on another color. These colors are chosen by loading VDP register 7 with the color 1 and color 0 in the left and right nibbles respectively (see Section 2.4).

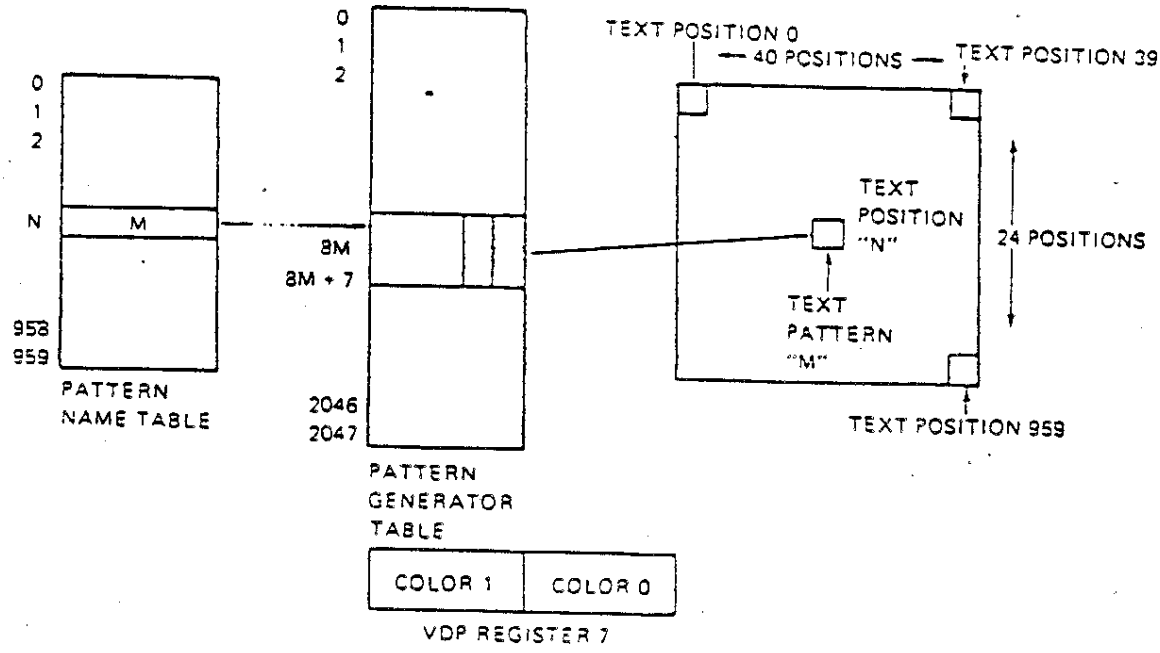

FIGURE 2-16 – MAPPING OF VRAM INTO THE PATTERN PLANE IN TEXT MODE

In the Text mode, the Pattern Name Table determines the position of the text pattern on the screen. There are 960 entries in the Pattern Name Table, each one byte long. There is a one-to-one correspondence between text pattern positions on the screen and entries in the Pattern Name Table (40*24 = 960). The first 40 entries corresponds to the top row of text pattern positions on the screen, the next forty to the second row, and so on. The value of an entry in the Pattern Name Table indicates which of the 256 text patterns is to be placed at that spot on the Pattern plane. The Pattern Name Table is located in a contiguous 960-byte block in VRAM beginning on a 1 kilobyte boundary. The starting address of the name table is determined by the 4-bit Name Table base address field in VDP register 2. The base address forms the upper 4 bits of the 14-bit VRAM address. The lower 10 bits of the VRAM address point to one of 960 pattern cells. The name table is organized by rows. An example of Pattern Name Table addressing is given in Section 3. Each byte entry in the name table is the pointer to a pattern definition in the Pattern Generator Table. The same two colors are used for all 256 patterns; the color codes are stored in VDP register 7.

As its name implies, the Text mode is intended mainly for textual applications, especially those in which the 32 patterns-per-line in Graphics modes is insufficient. The advantage is that eight more patterns can be fitted onto one line; the disadvantages are that sprites cannot be used, and only two colors are available for the entire screen. With care, the same text pattern set that is used in Text mode can be also used in Graphics I mode. This is done by ensuring that the least-significant 2 bits of all the character patterns are '0'. A switch from Text mode to Pattern mode, then, results in a stretching of the space between characters, and a reduction of the number of characters per line from 40 to 32. As with the Graphics Modes, once a character set has been defined and placed into the Pattern Generator, updating the Pattern Name Table will produce and manipulate textual material on the screen.

The full 8-bit pattern name is used to select one of the 256 pattern definitions in the pattern generator table. The table is a 2048-byte block in VRAM beginning on a 2 kilobyte boundary. The starting address of the table is determined by the generator base address in VDP register 4. The base address forms the 3 most-significant bits of the 14-bit VRAM address for each Pattern Generator Table entry. The next 8 bits are equal to the 8-bit name of the selected pattern definition. The lowest 3 bits of the VRAM address are equal to the row number within the pattern definition.

Eight bytes are required for each of the 256 possible unique 6 X 8 pattern definitions. The first byte defines the first row of the pattern, and the second byte defines the second row. The two least-significant bits in each byte are not used. It is, however, strongly recommended that these bits be '0's. Each bit entry in the pattern definition selects one of the two colors for that pattern. A '1' bit selects the color code (color 1) contained in the most-significant 4 bits of VDP register 7. A '0' bit selects the other color code (color 0) which is in the least-significant 4 bits of the same VDP Register. An example of pattern definition mapping is provided in Figure 2-16.

A total of 3005 VRAM bytes are required for the Pattern Name and Generator Tables. Less memory is required if all 256 possible pattern definitions are not required; the tables can be overlapped to reduce the amount of VRAM needed for pattern generation. Examples of VRAM memory allocation are provided in Appendix B.

2.7.5 Sprites

The video display can have up to 32 *sprites on the highest priority video planes*. The sprites are special animation patterns which provide smooth motion and multilevel pattern overlaying. The location of a sprite is defined by the top left-hand corner of the sprite pattern. The sprite can be easily moved pixel-by-pixel by redefining the sprite origin. This provides a simple but powerful method of quickly and smoothly moving special patterns. The sprites are not active in the Text mode. The 32 Sprite Planes are fully transparent outside of the sprite itself.

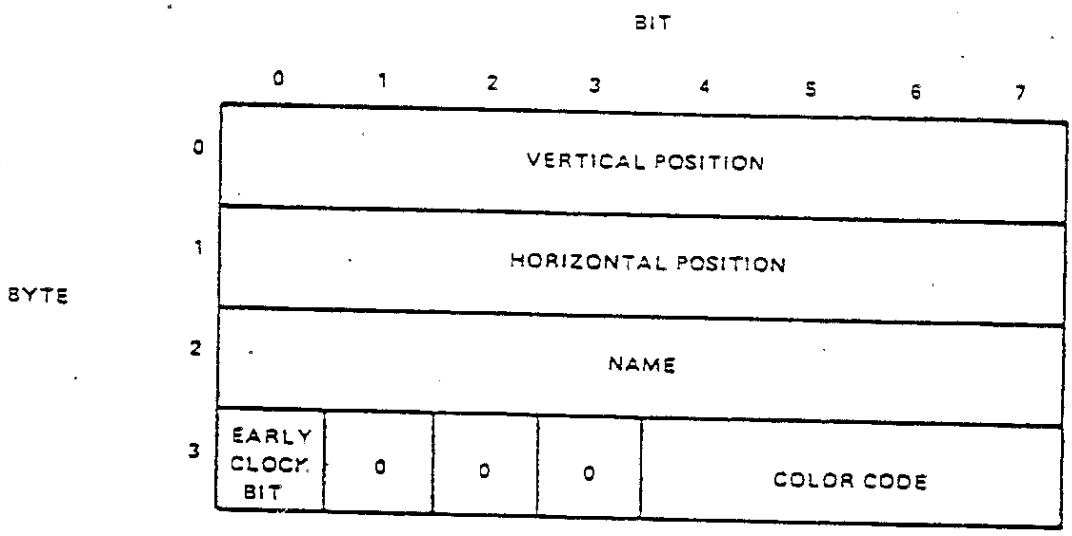

FIGURE 2-17 – SPRITE ATTRIBUTE TABLE ENTRY

The sub-blocks in VRAM that define sprites are the Sprite Attribute Table (see example of entry in Figure 2-17) and the Sprite Generator Table. These tables are similar to their equivalents in the pattern realm in that the Sprite Attribute Table specifies where the sprite goes on the screen, while the Sprite Generator Table describes what the sprite looks like. Sprite Pattern formats are given in Table 5.

Figure 2-18 illustrates the manner in which the VRAM tables map into the existence of sprites on the display. Since there are 32 sprites available for display, there are 32 entries in the Sprite Attribute Table. Each entry consists of four bytes. The entries are ordered so that the first entry corresponds to the sprite on the sprite 0 plane, the next to the sprite on the sprite 1 plane, and so on. The Sprite Attribute Table is 4*32 = 128 bytes long. The Sprite Attribute Table is located in a contiguous 128-byte block in VRAM beginning on a 128-byte boundary. The starting address of the Attribute Table is determined by the 7-bit Sprite Attribute Table base address in VDP register 5. The base address forms the upper seven bits of the 14-bit VRAM address. The next 5 bits of the VRAM address are equal to the sprite number. The lowest 2 bits select one of the four bytes in the Attribute Table entry for each sprite. Each Sprite Attribute Table entry contains four bytes which specify the sprite position, sprite pattern name, and color as shown in Figure 2-17.

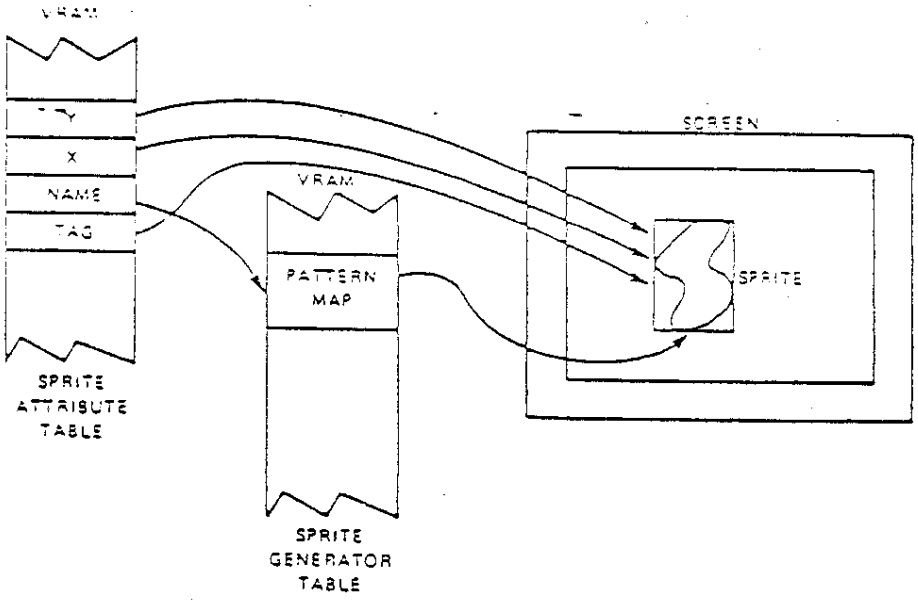

FIGURE 2-18 — SPRITE MAPPING

TABLE 5. SPRITE PATTERN FORMATS

SIZE	MAG	AREA	RESOLUTION	BYTES/PATTERN
0	0	8 x 8	single pixel	8
1	0	16 x 16	single pixel	32
0	1	16 x 16	2 x 2 pixels	8
1	1	32 x 32	2 x 2 pixels	32

The first two bytes of each entry of the Sprite Attribute Table determine the position of the sprite on the display. The first byte indicates the vertical distance of the sprite from the top of the screen, in pixels. It is defined such that a value of −1 puts the sprite butted up at the top of the screen, touching the backdrop area. The second bytes describes the horizontal displacement of the sprite from the left edge of the display. A value of 0 butts the sprite up against the left edge of the backdrop. Note that it is from the upper left pixel of the sprite that all measurement are taken.

When the first two bytes of an entry position a sprite overlapping the backdrop, the part of the sprite that is within the backdrop is displayed normally. The part of the sprite that overlaps the backdrop is hidden from view by the backdrop. This allows the animator to move a sprite into the display from behind the backdrop. The displacement in the first byte is partially signed, in that values for vertical displacement between −31 and 0 ($E1_{16}$ to 0) allow a sprite to "bleed in" from the top edge of the backdrop. Likewise, values in the range of 207 to 191 allow the sprite to bleed in from the bottom edge of the backdrop. Similarly, horizontal displacement values in the vicinity of 255 allow a sprite to bleed-in from the right side of the screen. To allow sprites to bleed-in from the left edge of the backdrop, a special bit in the third byte of the Sprite Attribute Table entry is used, as described in a later paragraph.

Byte 3 of the Sprite Attribute Table entry contains the pointer to the Sprite Generator Table that specifies what the sprite should look like. This is an 8-bit pointer to the sprite patterns definition, the Sprite Generator Table. The sprite name is similar to that in the Patterns Graphics mode.

Byte 4 of the Sprite Attribute Table entry contains the color of the sprite in its lower 4 bits (see Table 2 for color codes). The most-significant bit is the Early Clock bit (EC). This bit, when set to a '0', does nothing. When set to '1', the horizontal poisition of the sprite is shifted to the left by 32 pixels. This allows a sprite to bleed-in from the left edge of the backdrop. Values for horizontal displacement (byte 2 in the entry) in the range 0 to 32 cause the sprite to overlap with the left-hand border of the backdrop.

The Sprite Generator Table is a maximum of 2048 bytes long beginning on the 2 kilobyte boundaries. It is arranged into 256 blocks of 8 bytes each. The third byte of the Sprite Attribute Table entry, then, specifies which eight byte block to use to specify a sprite's shape. The '1's in the Sprite Generator cause the sprite to be defined at that point; '0's cause the transparent color to be used. The starting address of the table is determined by the sprite generator base address in VDP register 6. The base address forms the 3 most-significant bits of the 14-bit VRAM address. The next 8 bits of the address are equal to sprite name, and the last 3 bits are equal to the row number within the sprite pattern. The address formation is slightly modified for SIZE$_1$ sprites.

There is a maximum limit of four sprites that can be displayed on one horizontal line. If this rule is violated, the four highest-priority sprites on the line are displayed normally. The fifth and subsequent sprites are not displayed on that line. Furthermore, the fifth-sprite bit in the VDP status register is set to a '1', and the number of the violating fifth sprite is loaded into the status register (see Section 2.5).

Larger sprites than 8 X 8 pixels can be used if desired. The MAG and SIZE bits in VDP register 1 are used to select the various options. The options are described here:

MAG = 0, SIZE = 0: No options chosen;

MAG = 1, SIZE = 0: Eight bytes are still used in the Sprite Generator Table to describe the sprite; however, each bit in the Sprite Generator maps into 2 X 2 pixels on the TV screen, effectively doubling the size of the sprite to 16 X 16.

MAG = 0, SIZE = 1: 32 bytes are used in the Sprite Generator Table to define the sprite shape; the result is a 16 X 16 pixel sprite. The mapping of the 32 bytes into the sprite image is as shown in Figure 2-19. Mapping is still one bit-to-one pixel.

MAG = 1, SIZE = 1: Same as MAG = 0, SIZE = 1 except each bit now maps into a 2 X 2 pixel area, yielding a 32 X 32 sprite.

The VDP provides sprite coincidence checking. The coincidence status flag in the VDP status register is set to a '1' whenever two active sprites have '1' bits at the same screen location.

Sprite processing is terminated if the VDP finds a value of 208 ($D0_{16}$) in the vertical position field of any entry in the Sprite Attribute Table. This permits the Sprite Attribute Table to be shortened to the minimum size required; it also permits the user to blank out part or all of the sprites by simply changing one byte in VRAM.

A total of 2176 VRAM bytes are required for the Sprite Name and Pattern Generator Tables. Significantly less memory is required if all 256 possible sprite pattern definitions are not required. The Sprite Attribute Table can also be shortened as described above. The tables can be overlapped to reduce the amount of VRAM required for sprite generation. Examples of VRAM memory allocation are provided in Section 3.

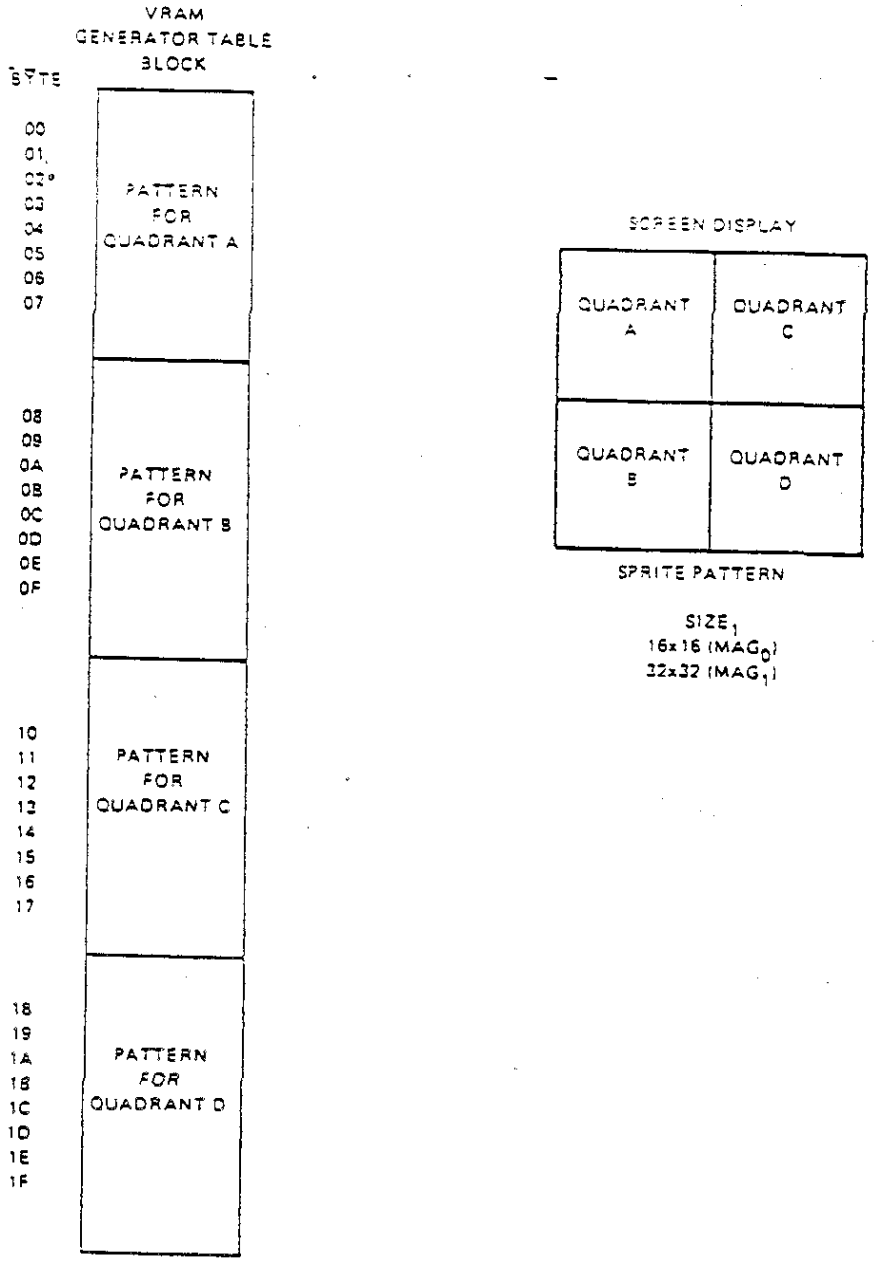

FIGURE 2-19 – SIZE 1 SPRITE MAPPING

2.8 EXTERNAL VIDEO

The external video interface allows mixing an external video source and VDP generated video under software control. As shown in Figure 2-20, composite video signals that are input to the 9918A through the external video input pin appear on the television screen in the plane behind the backdrop when the transparent color is programmed. Thus VDP-generated images on the Pattern Plane and the Sprite Planes can be superimposed upon an incoming signal. The source of the signal may be a standard NTSC broadcast signal, the output from an NTSC-compatible video-tape recorder, another VDP chip output, or any other NTSC-compatible composite video signal.