

VIDEO DISPLAY PROCESSOR

2.1 Introduction

The 9929 VDP contains all the circuitry necessary to generate the video display. It appears to the Z80 as two I/O ports called the Data Port and the Command Port. Although the VDP has its own 16 KB of VRAM (Video RAM), the contents of which define the screen image, this cannot be directly accessed by the Z80. Instead it must use the two I/O ports to modify the VRAM and to set the various VDP operating conditions.

2.2 Data Port (I/O Port 98H)

The Data Port is used to read or write single bytes to the VRAM. The VDP possesses an internal address register pointing to a location in the VRAM. Reading the Data Port will input the byte from this VRAM location while writing to the Data Port will store a byte there. After a read or write the address register is automatically incremented to point to the next VRAM location. Sequential bytes can be accessed simply by continuous reads or writes to the Data Port.

2.3 Command Port (I/O Port 99H)

The Command Port is used for three purposes: (1) To set up the Data Port address register. (2) To read the VDP Status Register. (3) To write to one of the VDP Mode Registers.

2.3.1 Address Register

The Data Port address register must be set up in different ways depending on whether the subsequent access is to be a read or a write. The address register can be set to any value from 0000H to 3FFFH by first writing the LSB (Least Significant Byte) and then the MSB (Most Significant Byte) to the Command Port. Bits 6 and 7 of the MSB are used by the VDP to determine whether the address register is being set up for subsequent reads or writes as follows:

| | | |
|-------|---------|---------|
| Read | xxxxxxx | 00xxxxx |
| Write | xxxxxxx | 01xxxxx |

Figure 7: VDP Address Setup.

It is important that no other accesses are made to the VDP in between writing the LSB and the MSB as this will upset its synchronization. The MSX ROM interrupt handler is continuously reading the VDP Status Register as a background task so interrupts should be disabled as necessary.

2.3.2 VDP Status Register

Reading the Command Port will input the contents of the VDP Status Register. This contains various flags as below:

| | | | | | | | |
|-----------|------------|-----------|---------------------|---|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| F Flag | 5S Flag | C Flag | Fifth Sprite Number | | | | |

Figure 8: VDP Status Register.

The **Fifth Sprite Number** bits contain the number (0 to 31) of the sprite triggering the Fifth Sprite Flag.

The **Coincidence Flag** is normally 0 but is set to 1 if any sprites have one or more overlapping pixels. Reading the Status Register will reset this flag to a 0. Note that coincidence is only checked as each pixel is generated during a video frame, on a UK machine this is every 20 ms. If fast moving sprites pass over each other between checks then no coincidence will be flagged.

The **Fifth Sprite Flag** is normally 0 but is set to 1 when there are more than four sprites on any pixel line. Reading the Status Register will reset this flag to a 0.

The **Frame Flag** is normally 0 but is set to a 1 at the end of the last active line of the video frame. For UK machines with a 50 Hz frame rate this will occur every 20 ms. Reading the Status register will reset this flag to a 0. There is an associated output signal from the VDP which generates Z80 interrupts at the same rate, this drives the MSX ROM interrupt handler.

2.3.3 VDP Mode Registers

The VDP has eight write-only registers, numbered 0 to 7, which control its general operation. A particular register is set by first writing a data byte then a register selection byte to the Command Port. The register selection byte contains the register number in the lower three bits: 10000RRR. As the Mode Registers are write-only, and cannot be read, the MSX ROM maintains an exact copy of the eight registers in the Workspace Area of RAM (Chapter 6). Using the MSX ROM standard routines for VDP functions ensures that this register image is correctly updated.

| Mode Register 0 | | | | | | | |
|-----------------|---|---|---|---|---|----|----|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | M3 | EV |

Figure 9: Mode Register 0.

The **External VDP** bit determines whether external VDP input is to be enabled or disabled: 0=Disabled, 1=Enabled.
The **M3** bit is one of the three VDP mode selection bits, see Mode Register 1.

| Mode Register 1 | | | | | | | |
|-----------------|-------|----|----|----|---|------|-----|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 4/16K | Blank | IE | M1 | M2 | 0 | Size | Mag |

Figure 10: Mode Register 1.

The **Magnification** bit determines whether sprites will be normal or doubled in size: 0=Normal, 1=Doubled.
The **Size** bit determines whether each sprite pattern will be 8x8 bits or 16x16 bits: 0=8x8, 1=16x16.
The **M1** and **M2** bits determine the VDP operating mode in conjunction with the M3 bit from Mode Register 0:

| M1 | M2 | M3 | |
|----|----|----|------------------|
| 0 | 0 | 0 | 32x24 Text Mode |
| 0 | 0 | 1 | Graphics Mode |
| 0 | 1 | 0 | Multicolour Mode |
| 1 | 0 | 0 | 40x24 Text Mode |

The **Interrupt Enable** bit enables or disables the interrupt output signal from the VDP: 0=Disable, 1=Enable.
The **Blank** bit is used to enable or disable the entire video display: 0=Disable, 1=Enable. When the display is blanked it will be the same colour as the border.
The **4/16K** bit alters the VDP VRAM addressing characteristics to suit either 4 KB or 16 KB chips: 0=4 KB, 1=16 KB.

| Mode Register 2 | | | | | | | |
|-----------------|---|---|---|-----------------|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 0 | 0 | 0 | 0 | Name Table Base | | | |

Figure 11: Mode Register 2.

Mode Register 2 defines the starting address of the Name Table in the VDP VRAM. The four available bits only specify positions 00BB BB00 0000 0000 of the full address so register contents of 0FH would result in a base address of 3C00H.

| Mode Register 3 | | | | | | | |
|-------------------|---|---|---|---|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Colour Table Base | | | | | | | |

Figure 12: Mode Register 3.

Mode Register 3 defines the starting address of the Colour Table in the VDP VRAM. The eight available bits only specify positions 00BB BBBB BB00 0000 of the full address so register contents of FFH would result in a base address of 3FC0H. In Graphics Mode only bit 7 is effective thus offering a base of 0000H or 2000H. Bits 0 to 6 must be 1.

| Mode Register 4 | | | | | | | |
|-----------------|---|---|---|---|-------------------|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 | Character Pattern | | |

Figure 13: Mode Register 4.

Mode Register 4 defines the starting address of the Character Pattern Table in the VDP VRAM. The three available bits only specify positions 00BB B000 0000 0000 of the full address so register contents of 07H would result in a base address of 3800H. In Graphics Mode only bit 2 is effective thus offering a base of 0000H or 2000H. Bits 0 and 1 must be 1.

| Mode Register 5 | | | | | | | |
|-----------------|-----------------------|---|---|---|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 0 | Sprite Attribute base | | | | | | |

Figure 14: Mode Register 5.

Mode Register 5 defines the starting address of the Sprite Attribute Table in the VDP VRAM. The seven available bits only specify positions 00BB BBBB B000 0000 of the full address so register contents of 7FH would result in a base address of 3F80H.

| Mode Register 6 | | | | | | | |
|-----------------|---|---|---|---|----------------|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 | Sprite Pattern | | |

Figure 15: Mode Register 6.

Mode Register 6 defines the starting address of the Sprite Pattern Table in the VDP VRAM. The three available bits only specify positions 00BB B000 0000 0000 of the full address so register contents of 07H would result in a base address of 3800H.

| Mode Register 7 | | | | | | | |
|-----------------|---|---|---|---------------|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Text Colour 1 | | | | Border Colour | | | |

Figure 16: Mode Register 7.

The **Border Colour** bits determine the colour of the region surrounding the active video area in all four VDP modes. They also determine the colour of all 0 pixels on the screen in 40x24 Text Mode. Note that the border region actually extends across the entire screen but will only become visible in the active area if the overlying pixel is transparent.

The **Text Colour 1** bits determine the colour of all 1 pixels in 40x24 Text Mode. They have no effect in the other three modes where greater flexibility is provided through the use of the Colour Table. The VDP colour codes are:

| | | | | | | | |
|---|-------------|---|------------|----|--------------|----|------------|
| 0 | Transparent | 4 | Dark Blue | 8 | Red | 12 | Dark Green |
| 1 | Black | 5 | Light Blue | 9 | Bright Red | 13 | Purple |
| 2 | Green | 6 | Dark Red | 10 | Yellow | 14 | Grey |
| 3 | Light Green | 7 | Sky Blue | 11 | Light Yellow | 15 | White |

2.4 Screen Modes

The VDP has four operating modes, each one offering a slightly different set of capabilities. Generally speaking, as the resolution goes up the price to be paid in VRAM size and updating complexity also increases. In a dedicated application these associated hardware and software costs are important considerations. For an MSX machine they are irrelevant, it therefore seems a pity that a greater attempt was not made to standardize on one particular mode. The Graphics Mode is capable of adequately performing all the functions of the other modes with only minor reservations. An added difficulty in using the VDP arises because insufficient allowance was made in its design for the overscanning used by most televisions. The resulting loss of characters at the screen edges has forced all the video-related MSX software into being based on peculiar screen sizes. UK machines normally use only the central thirty-seven characters available in 40x24 Text Mode. Japanese machines, with NTSC (National Television Standards Committee) video outputs, use the central thirty-nine characters. The central element in the VDP, from the programmer's point of view, is the Name Table. This is a simple list of single-byte character codes held in VRAM. It is 960 bytes long in 40x24 Text Mode, 768 bytes long in 32x24 Text Mode, Graphics Mode and Multicolour Mode. Each position in the Name Table corresponds to a particular location on the screen. During a video frame the VDP will sequentially read every character code from the Name Table, starting at the base. As each character code is read the corresponding 8x8 pattern of pixels is looked up in the Character Pattern Table and displayed on the screen. The appearance of the screen can thus be modified by either changing the character codes in the Name Table or the pixel patterns in the Character Pattern Table. Note that the VDP has no hardware cursor facility, if one is required it must be software generated.

2.4.1 40x24 Text Mode

The Name Table occupies 960 bytes of VRAM from 0000H to 03BFH:

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|----|
| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | | |
| 0000H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 0 |
| 0028H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 1 |
| 0050H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 2 |
| 0078H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 3 |
| 00A0H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 4 |
| 00C8H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 5 |
| 00F0H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 6 |
| 0118H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 7 |
| 0140H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 8 |
| 0168H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 9 |
| 0190H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 10 |
| 01B8H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 11 |
| 01E0H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 12 |
| 0208H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 13 |
| 0230H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 14 |
| 0258H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 15 |
| 0280H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 16 |
| 02A8H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 17 |
| 02D0H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 18 |
| 02F8H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 19 |
| 0320H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 20 |
| 0348H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 21 |
| 0370H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 22 |
| 0398H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 23 |
| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | | |

Figure 17: 40x24 Name Table.

Pattern Table occupies 2 KB of VRAM from 0800H to 0FFFH. Each eight byte block contains the pixel pattern for a character code:

| | | | | | | | | |
|---|---|---|---|---|---|---|---|--------|
| 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | Byte 0 |
| 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | Byte 1 |
| 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | Byte 2 |
| 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | Byte 3 |
| 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | Byte 4 |
| 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | Byte 5 |
| 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | Byte 6 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Byte 7 |

Figure 18: Character Pattern Block (No. 65 shown = `A').

The first block contains the pattern for character code 0, the second the pattern for character code 1 and so on to character code 255. Note that only the leftmost six pixels are actually displayed in this mode. The colours of the 0 and 1 pixels in this mode are defined by VDP Mode Register 7, initially they are blue and white.

2.4.2 32x24 Text Mode

The Name Table occupies 768 bytes of VRAM from 1800H to 1AFFH. As in 40x24 Text Mode normal operation involves placing character codes in the required position in the table. The "VPOKE" statement may be used to attain familiarity with the screen layout:

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | |
|-------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|----|
| 1800H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 0 |
| 1820H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 1 |
| 1840H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 2 |
| 1860H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 3 |
| 1880H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 4 |
| 18A0H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 5 |
| 18C0H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 6 |
| 18E0H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 7 |
| 1900H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 8 |
| 1920H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 9 |
| 1940H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 10 |
| 1960H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 11 |
| 1980H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 12 |
| 19A0H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 13 |
| 19C0H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 14 |
| 19E0H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 15 |
| 1A00H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 16 |
| 1A20H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 17 |
| 1A40H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 18 |
| 1A60H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 19 |
| 1A80H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 20 |
| 1AA0H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 21 |
| 1AC0H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 22 |
| 1AE0H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 23 |
| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | |

Figure 19: 32x24 Name Table.

The Character Pattern Table occupies 2 KB of VRAM from 0000H to 07FFH. Its structure is the same as in 40x24 Text Mode, all eight pixels of an 8x8 pattern are now displayed. The border colour is defined by VDP Mode Register 7 and is initially blue. An additional table, the Colour Table, determines the colour of the 0 and 1 pixels. This occupies thirty-two bytes of VRAM from 2000H to 201FH. Each entry in the Colour Table defines the 0 and 1 pixel colours for a group of eight character codes, the lower four bits defining the 0 pixel colour, the upper four bits the 1 pixel colour. The first entry in the table defines the colours for character codes 0 to 7, the second for character codes 8 to 15 and so on for thirty-two entries. The MSX ROM initializes all entries to the same value, blue and white, and provides no facilities for changing individual ones.

2.4.3 Graphics Mode

The Name Table occupies 768 bytes of VRAM from 1800H to 1AFFH, the same as in 32x24 Text Mode. The table is initialized with the character code sequence 0 to 255 repeated three times and is then left untouched, in this mode it is the Character Pattern Table which is modified during normal operation. The Character Pattern Table occupies 6 KB of VRAM from 0000H to 17FFH. While its structure is the same as in the text modes it does not contain a character set but is initialized to all 0 pixels. The first 2 KB of the Character Pattern Table is addressed by the character codes from the first third of the Name Table, the second 2 KB by the central third of the Name Table and the last 2 KB by the final third of the Name Table. Because of the sequential pattern in the Name Table the entire Character Pattern Table is read out linearly during a video frame. Setting a point on the screen involves working out where the corresponding bit is in the Character Pattern Table and turning it on. For a BASIC program to convert X,Y coordinates to an address see the MAPXYC standard routine in Chapter 4.

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | |
|-------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|----|
| 0000H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 0 |
| 0100H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 1 |
| 0200H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 2 |
| 0300H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 3 |
| 0400H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 4 |
| 0500H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 5 |
| 0600H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 6 |
| 0700H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 7 |
| 0800H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 8 |
| 0900H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 9 |
| 0A00H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 10 |
| 0B00H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 11 |
| 0C00H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 12 |
| 0D00H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 13 |
| 0E00H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 14 |
| 0F00H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 15 |
| 1000H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 16 |
| 1100H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 17 |
| 1200H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 18 |
| 1300H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 19 |
| 1400H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 20 |
| 1500H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 21 |
| 1600H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 22 |
| 1700H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 23 |
| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | |

Figure 20: Graphics Character Pattern Table.

The border colour is defined by VDP Mode Register 7 and is initially blue. The Colour Table occupies 6 KB of VRAM from 000H to 37FFH. There is an exact byte-to-byte mapping from the Character Pattern Table to the Colour Table but, because it takes a whole byte to define the 0 pixel and 1 pixel colours, there is a lower resolution for colours than for pixels. The lower four bits of a Colour Table entry define the colour of all the 0 pixels on the corresponding eight pixel line. The upper four bits define the colour of the 1 pixels. The Colour Table is initialized so that the 0 pixel colour and the 1 pixel colour are blue for the entire table. Because both colours are the same it will be necessary to alter one colour when a bit is set in the Character Pattern Table.

2.4.4 Multicolour Mode

The Name Table occupies 768 bytes of VRAM from 0800H to 0AFFH, the screen mapping is the same as in 32x24 Text Mode. The table is initialized with the following character code pattern:

00H to 1FH (Repeated four times)
 20H to 3FH (Repeated four times)
 40H to 5FH (Repeated four times)
 60H to 7FH (Repeated four times)
 80H to 9FH (Repeated four times)
 A0H to BFH (Repeated four times)

As with Graphics Mode this is just a character code "driver" pattern, it is the Character Pattern Table which is modified during normal operation. The Character Pattern table occupies 1536 bytes of VRAM from 0000H to 05FFH. As in the other modes each character code maps onto an eight byte block in the Character Pattern Table. Because of the lower resolution in this mode only two bytes of the pattern block are actually needed to define an 8x8 pattern:

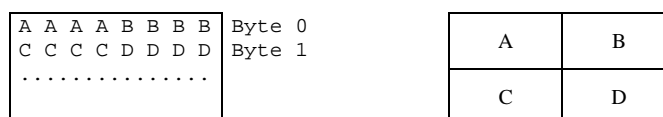


Figure 21: Multicolour Pattern Block.

As can be seen from Figure 21 each four bit section of the two byte block contains a colour code and thus defines the COLOUR of a quadrant of the 8x8 pixel pattern. So that the entire eight bytes of the pattern block can be utilized a given character code will use a different two byte section depending upon the character code's screen location (i.e. its position in the Name Table):

| | |
|--------------------------------|--------------------|
| Video row 0, 4, 8, 12, 16, 20 | Uses bytes 0 and 1 |
| Video row 1, 5, 9, 13, 17, 21 | Uses bytes 2 and 3 |
| Video row 2, 6, 10, 14, 18, 22 | Uses bytes 4 and 5 |
| Video row 3, 7, 11, 15, 19, 23 | Uses bytes 6 and 7 |

When the Name Table is filled with the special driver sequence of character codes shown above the Character Pattern Table will be read out linearly during a video frame:

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|----|
| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | |
| 0000H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 0 |
| 0002H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 1 |
| 0004H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 2 |
| 0006H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 3 |
| 0100H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 4 |
| 0102H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 5 |
| 0104H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 6 |
| 0106H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 7 |
| 0200H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 8 |
| 0202H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 9 |
| 0204H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 10 |
| 0206H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 11 |
| 0300H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 12 |
| 0302H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 13 |
| 0304H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 14 |
| 0306H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 15 |
| 0400H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 16 |
| 0402H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 17 |
| 0404H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 18 |
| 0406H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 19 |
| 0500H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 20 |
| 0502H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 21 |
| 0504H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 22 |
| 0506H | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 23 |
| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | |

Figure 22: Multicolour Character Pattern Table.

The border colour is defined by VDP Mode Register 7 and is initially blue. There is no separate Colour Table as the colours are defined directly by the contents of the Character Pattern Table, this is initially filled with blue.

2.5 Sprites

The VDP can control thirty-two sprites in all modes except 40X24 Text Mode. Their treatment is identical in all modes and independent of any character-orientated activity. The Sprite Attribute Table occupies 128 bytes of VRAM from 1B00H to 1B7FH. The table contains thirty-two four byte blocks, one for each sprite. The first block controls sprite 0 (the "top" sprite), the second controls sprite 1 and so on to sprite 31. The format of each block is as below:

| | | | | | | | | |
|---------------------|---|---|---|---|-------------|---|---|--------|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
| Vertical Position | | | | | | | | Byte 0 |
| Horizontal Position | | | | | | | | Byte 1 |
| Pattern Number | | | | | | | | Byte 2 |
| EC | 0 | 0 | 0 | 0 | Colour Code | | | Byte 3 |

Figure 23: Sprite Attribute Block.

Byte 0 specifies the vertical (Y) coordinate of the top-left pixel of the sprite. The coordinate system runs from -1 (FFH) for the top pixel line on the screen down to 190 (BEH) for the bottom line. Values less than -1 can be used to slide the sprite in from the top of the screen. The exact values needed will obviously depend upon the size of the sprite. Curiously there has been no attempt in MSX BASIC to reconcile this coordinate system with the normal graphics range of Y=0 to 191. As a consequence a sprite will always be one pixel lower on the screen than the equivalent graphic point. Note that the special vertical coordinate value of 208 (D0H) placed in a sprite attribute block will cause the VDP to ignore all subsequent blocks in the Sprite Attribute Table. Effectively this means that any lower sprites will disappear from the screen.

Byte 1 specifies the horizontal (X) coordinate of the top-left pixel of the sprite. The coordinate system runs from 0 for the leftmost pixel to 255 (FFH) for the rightmost. As this coordinate system provides no mechanism for sliding a sprite in from the left a special bit in byte 3 is used for this purpose, see below.

Byte 2 selects one of the two hundred and fifty-six 8x8 bit patterns available in the Sprite Pattern Table. If the Size bit is set in VDP Mode Register 1, resulting in 16x16 bit patterns occupying thirty-two bytes each, the two least significant bits of the pattern number are ignored. Thus pattern numbers 0, 1, 2 and 3 would all select pattern number 0.

In **Byte 3** the four Colour Code bits define the colour of the 1 pixels in the sprite patterns, 0 pixels are always transparent. The Early Clock bit is normally 0 but will shift the sprite thirty-two pixels to the left when set to 1. This is so that sprites can slide in from the left of the screen, there being no spare coordinates in the horizontal direction.

The Sprite Pattern Table occupies 2 KB of VRAM from 3800H to 3FFFH. It contains two hundred and fifty-six 8x8 pixel patterns, numbered from 0 to 255. If the Size bit in VDP Mode Register 1 is 0, resulting in 8x8 sprites, then each eight byte sprite pattern block is structured in the same way as the character pattern block shown in Figure 18. If the Size bit is 1, resulting in 16x16 sprites, then four eight byte blocks are needed to define the pattern as below:

| | | |
|-----------------|---|---|
| 8 Bytes Block A | A | C |
| 8 Bytes Block B | | |
| 8 Bytes Block C | B | D |
| 8 Bytes Block D | | |

Figure 24: 16x16 Sprite Pattern Block.