TECHNICAL NOTE

DESIGNING WITH THE MT42C4256/8128 VRAM

INTRODUCTION

Memory buffers designed with video RAM (VRAM) will outperform similar, DRAM-based designs for a variety of applications, from graphics frame buffers to data communications and networking. Furthermore, the enhanced feature sets offered by Micron's 1 Meg VRAM can simplify and improve the performance of these designs. This note highlights the functional details of the Micron Semiconductor MT42C4256/8128 family of 1 Meg density VRAMs. All references to "VRAM" refer to the MT42C4256/8128 VRAM except where noted. The 1 Meg Triple-Port DRAMs (MT43C4257/8; 8128/9) also share common functionality with the MT42C4256/8128, all modes common between these devices are covered by this technical note.

VIDEO RAM BASICS

A VRAM is created by adding an independent static memory to a dynamic RAM (DRAM) core array. To address the bandwidth limitations of standard DRAMs, the static memory is accessed via a separate 4- or 8-bit-wide port.

The second memory buffer is referred to as a serial access memory, or SAM. It is a fully static memory equal in size to one row of the DRAM array. It is addressed in an incremental manner by an address counter/pointer that is incremented by a special clock pin, see Figure 2. This makes it well suited for high speed sequential data streams, as present in raster display graphics subsystems, network data buffers and communications.

To use the SAM as an output port, data must move internally from the DRAM array to the SAM I/O. Data written into the DRAM array from the DRAM I/O is internally moved a row at a time to the SAM by a TRANSFER cycle. The TRANSFER is facilitated by pass (transfer) gates between the DRAM column sense amplifiers and the SAM storage cells. A TRANSFER begins as any DRAM

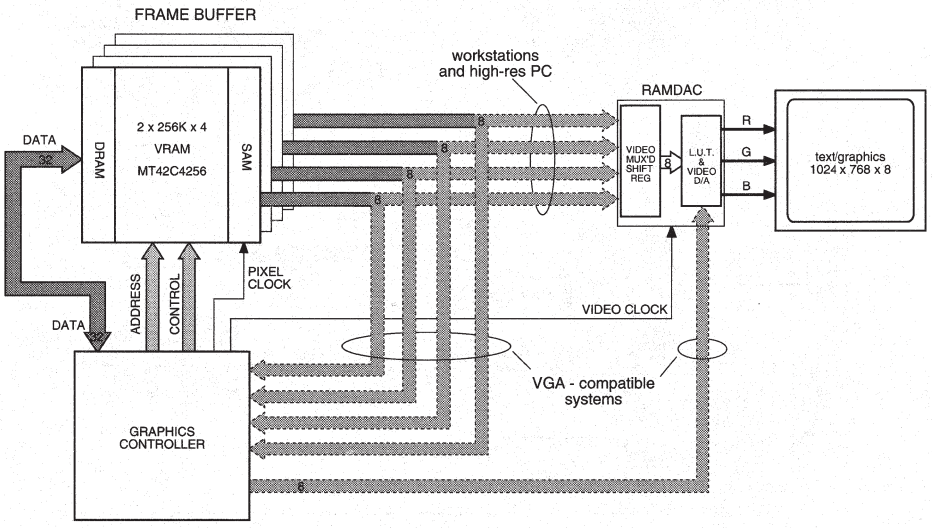

Figure 1
TYPICAL GRAPHICS APPLICATIONS FOR VRAM

access would: a row-address is selected by strobing the $\overline{RAS}$ pin. When the row is selected, the contents of all its capacitive storage elements are "read" by the column sense amplifiers. Once the data from the row is amplified ("opened"), it is ready to be sent to the output pins or overwritten from the input pins of the DRAM. A TRANSFER begins by opening a row in the same manner, but there the similarity ends. During a transfer, the data is passed (as an entire row) between the sense amps and the SAM static latches residing at the "end" of each column. Micron VRAMs incorporate bidirectional SAMs that can be configured to output data, as described above, or input data and then "write" it into a DRAM row.

Once the row is open, the DRAM access cycle selects a column-address by strobing $\overline{CAS}$. This selects the specific column sense amps that will drive a unique word to the outputs. A VRAM TRANSFER cycle does not input or output data at the DRAM I/O. Instead, the column-address is latched into a SAM "Tap" address buffer. The Tap address defines the column-address at which SAM input or output will begin.

The type of TRANSFER cycle performed determines the direction of the data flow of the SAM. Transfers are referenced from the DRAM. A READ TRANSFER means that data is read from the DRAM to the SAM and sets the SAM I/O as an output. A WRITE TRANSFER writes information from the SAM to the DRAM and sets the SAM I/O to the input direction.

Figure 1 illustrates a basic graphics frame buffer using VRAM devices. The DRAM port is accessed and controlled by a graphics controller or coprocessor. This port is used to update the display pixel data, perform DRAM refresh and control TRANSFERS from the DRAM array and the SAM. The SAM supplies the pixel data to the raster-video display circuitry. This consists of a video digital-to-analog converter (DAC) and an SRAM-based look-up table (LUT). Most systems also utilize a high-speed multiplexed shift register, or a DAC incorporating one, to increase the effective pixel rate without requiring the SAM clock to run at the video pixel, or dot clock, rate (80 MHz or more). Many video DACs also incorporate the LUT and are referred to as palette DACs or RAMDAC™s.

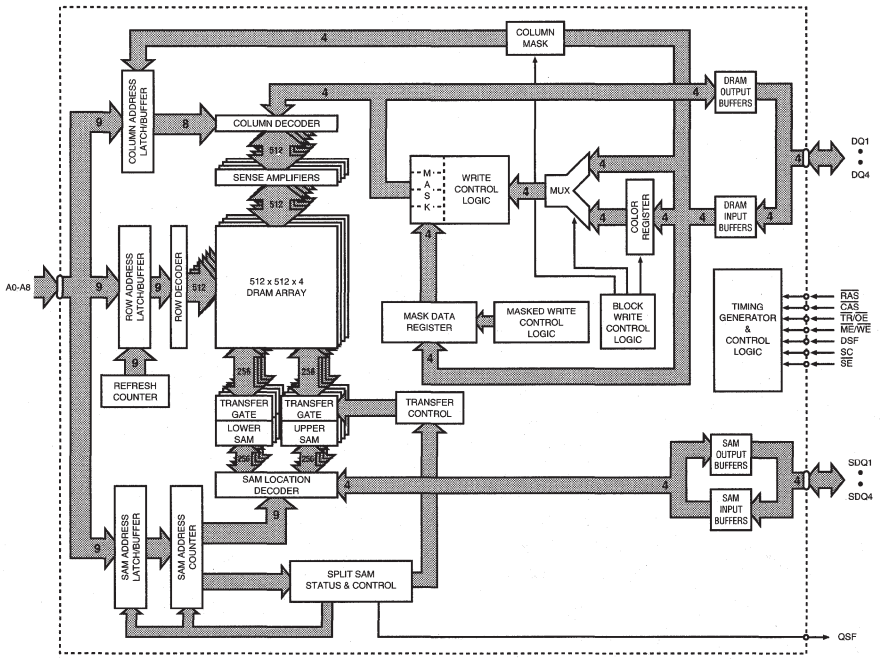

Figure 2
MT42C4256 FUNCTIONAL BLOCK DIAGRAM

In systems with VGA-compatible controllers, the SAM data is routed back through the controller. In these cases, the video multiplexer is incorporated in the controller. This reduces the cost and complexity of the DAC required. To get full advantage of the VRAM bandwidth, the VGA controller should have separate I/O for the DRAM and SAM ports.

The serial clock for the SAM is derived by dividing the pixel clock by the number of pixels accessed per SAM clock cycle. It may be generated by the controller or by the palette DAC depending on the implementation and components chosen.

ADVANTAGE OF VRAM OVER DRAM

VRAMs were developed to increase the bandwidth of raster graphic display frame buffers. If DRAM is used to build the frame buffer, it must allow access by both the host/graphics controller and the CRT refresh circuitry. The nature of raster graphics displays requires that a constant, uninterrupted flow of pixel data be available to the CRT driver circuitry. Because of this, it is the host or graphics processor that must be interrupted when a request is made by the CRT driver for a new line of pixel data (scan line). DRAM-based frame buffers suffer from data contention at the DRAM port that reduces screen redraw speed. A 1 Megabyte frame buffer designed with 70ns, single-port DRAMs would use 90 percent or more of the total memory bandwidth to perform screen refresh. This leaves 10 percent or less of the bandwidth for drawing new pixels to the display. Obviously, this will result in a noticeably slower display time when new information is to be displayed. When designed with 70ns VRAM, a similar frame buffer would only utilize two percent of the total bandwidth for screen refresh, resulting in screen redraw performance improvement of roughly an order of magnitude.

VRAM-based frame buffers are designed to give the CRT driver circuitry access to pixel data from the SAM port, thus alleviating the DRAM-port contention problem as seen with DRAM-based buffers (see Figure 1). This results in improved screen redraw performance over DRAM-based buffers when changing display information. DRAM-port accesses to the VRAM must be interrupted for READ TRANSFERs from the DRAM array to the SAM, but they occur infrequently since an entire row (page) of data is transferred per cycle. DRAM-port access is available to the graphics controller or coprocessor while screen refresh data is constantly supplied, independently, by the SAM port.

Other features of the VRAM further enhance graphics performance. These "special features" of Micron's 1 Meg VRAM include: MASKED WRITE, BLOCK WRITE, and SPLIT READ TRANSFER. These features, and the TRANSFER cycles, are detailed in the following sections, along with other features of Micron's 1 Meg VRAM.

SAM TRANSFERS

Data TRANSFERs move data between the two VRAM memory arrays. These cycles are variations of a normal DRAM $\overline{RAS}$-$\overline{CAS}$ cycle. To incorporate TRANSFERs, the VRAM's $\overline{OE}$ pin is made a dual-function pin and called Transfer Enable/Output Enable ($\overline{TR}/\overline{OE}$). At the falling edge of $\overline{RAS}$, $\overline{TR}/\overline{OE}$ is sampled; if it is LOW, the cycle will be a TRANSFER. Once a TRANSFER is selected, its direction must be defined. This also occurs at the falling edge of $\overline{RAS}$ by the level of the $\overline{ME}/\overline{WE}$ pin. When $\overline{ME}/\overline{WE}$ is LOW, a WRITE TRANSFER is selected; when $\overline{ME}/\overline{WE}$ is HIGH, a READ TRANSFER is selected. A mode-select pin called DSF is used during TRANSFERs to define variants of the standard READ and WRITE TRANSFERs. This pin is also latched at the falling edge of $\overline{RAS}$. The following sections describe the detail of all the TRANSFER variations.

Note: *All VRAM operations are defined at the falling edge of $\overline{RAS}$, with the exception of BLOCK WRITE.*

CRT REFRESH USING SERIAL OUTPUT MODE

A design using VRAM for a graphics frame buffer uses the SAM port, in the output mode, to update the CRT with pixel information. When using a VRAM in this manner, the pixel data is transferred a page at a time by READ TRANSFER cycles to the SAM. Then it is sequentially clocked out of the SAM to the DAC driving the CRT, see Figure 1.

There are three kinds of READ TRANSFERs: normal READ TRANSFER (RT), which is not synchronized with the serial clock (SC); a REAL-TIME RT (RTRT), which is synchronized to an SC cycle; and SPLIT RT (SRT), in which data is asynchronously transferred while the SAM is active.

Pixel data is stored as scan lines in the DRAM array. Each row of the array may contain all, part or many video scan lines depending on the organization and size of the frame buffer and display. Different organizations require different methods of moving data to the SAM in order to optimize performance and reduce timing requirements of the transfer cycles.

If the SAM data is loaded only during retrace time of the CRT, a normal READ TRANSFER may be performed. This will be the case when a VRAM row contains one or more complete scan lines, no partial lines. An RT allows the VRAM to time the movement of screen data from the DRAM array to the SAM internally. The pixel clock will be idle during this TRANSFER. In this case the $\overline{TR}/\overline{OE}$ pin

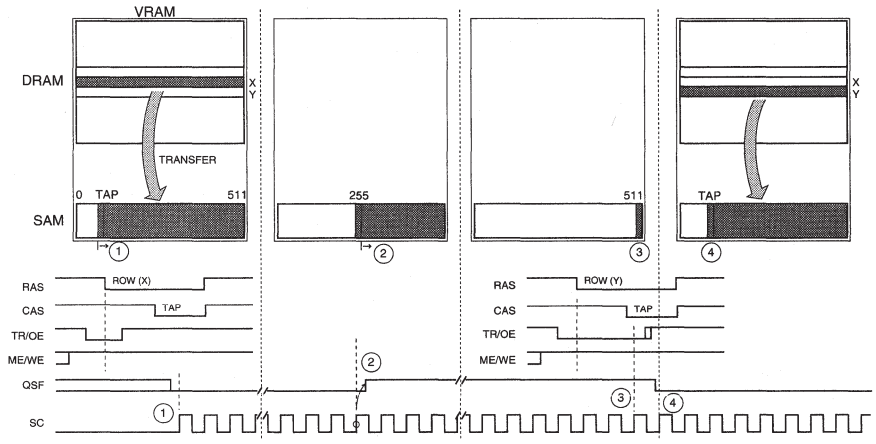

Figure 3
REAL-TIME READ TRANSFER EXAMPLE

may be taken from LOW to HIGH any time during the $\overline{RAS}$-$\overline{CAS}$ cycle. The exact timing of the internal transfer is not important, except that it be complete before the serial clock is restarted. Figures 3 and 4 illustrate this operation as the first TRANSFER cycle at state ①; here, $\overline{TR}/\overline{OE}$ is taken HIGH "early" (before $\overline{CAS}$ falls).

Note: *To implement an all-purpose VRAM design, $\overline{TR}/\overline{OE}$ should be taken HIGH before or at the same time $\overline{RAS}$ and $\overline{CAS}$ go HIGH to terminate the TRANSFER cycle.*

When screen data must be loaded into the SAM during active display time, a REAL-TIME READ TRANSFER may be performed. This allows the data output to proceed in an uninterrupted fashion required when the VRAM rows contain partial scan lines. RTRTs must be synchronized with the serial clock so that the active pixel data is not disrupted when the next portion of the scan line is moved to the SAM. The $\overline{TR}/\overline{OE}$ pin is used to control the exact moment of the transfer of data from the DRAM row to the SAM. By holding $\overline{TR}/\overline{OE}$ LOW until after $\overline{CAS}$ falls (meeting ^{t}RTH and ^{t}CTH), the transfer is "held off" until the LOW-to-HIGH transition. When $\overline{TR}/\overline{OE}$ goes HIGH, the transfer gates open and let the new row move from the sense amps to the SAM, all within a single SC cycle. This is shown from state ③ to state ④ of Figure 3.

Micron's 1 Meg VRAM introduces a transfer feature that greatly reduces the timing demands of REAL-TIME READ TRANSFERs while allowing continuous data streams out of the SAM port. By splitting the SAM and the transfer circuitry into two separately controlled halves, it is possible to transfer screen data to one half of the SAM while the other half is actively supplying screen data. This cycle is called SPLIT READ TRANSFER (SRT), and it eliminates the restriction of having to synchronize the rising $\overline{TR}/\overline{OE}$ edge

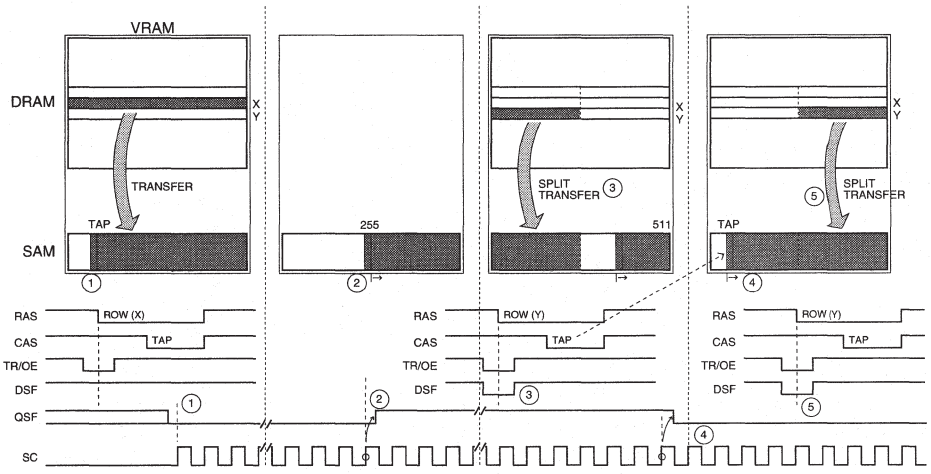

Figure 4
SPLIT READ TRANSFER EXAMPLE

with the serial clock during an RTRT cycle. Figure 4 illustrates a state sequence for split SAM operation. Before split operation can begin, the SAM must be initialized. A normal RT is used to ensure the SAM I/O is in the output state and that a known Tap address is loaded into the SAM address counter/pointer. There is no restriction regarding the initial Tap address location for the Micron VRAM. Once the RT is performed, any number of SRTs may follow. Most 1 Meg VRAMs, including Micron's, will do the SRT automatically to the "idle" half of the SAM. The most significant column-address bit latched at $\overline{CAS}$ LOW is a "don't care."

Note: *Repetitive TRANSFERs may be done regardless of the occurrence of a serial clock cycle. A READ TRANSFER may be immediately followed by a SPLIT READ TRANSFER, provided $\overline{RAS}$ precharge time is met per the specification.*

When using SRT, the system is given a window of time equal to the time needed to read a SAM-half (about 7.6µs for a 256K x 4, 80ns VRAM) to update the idle half and maintain an uninterrupted flow of data from the SAM. This is illustrated from state ② to ④ of Figure 4. The SRT is performed after the address count of the SAM moves from the "old" half (lower half, as shown at state ②) to the "new" half (upper half). State ③ of Figure 4 shows the SRT being executed a finite time after the first clock, but before the last clock, of the upper half. This delay ensures that the SAM counter has entered the new half of the SAM, ensuring that the SRT is invoked to the old half. After address 511, the SAM address counter jumps to the lower half, state ④, and to whatever Tap address was loaded during the SRT performed at state ③. Now the lower half is the active half again, and a SRT to the upper half may be done, ⑤.

The Split SAM Status (QSF) output pin is provided to indicate which half of the SAM is active. By monitoring the QSF pin, it is possible to update data in the SAM without having to externally track the SAM address. At the transition of the SAM boundary, QSF will change state (LOW for lower half, HIGH for upper). When this occurs, the controller performs an SRT to the idle half and the new screen data will be available when the next portion of the scan line is required. Using SRT will simplify the graphics controller design by reducing the control logic and critical timing associated with REAL-TIME READ TRANSFER.

USING THE SAM FOR SERIAL INPUT

An increasing number of VRAM applications require the serial port to be used as an input port. These include video capture, network data buffers and data communications (satellite, telecom, etc.). In these applications, the SAM is used to input data to be moved to the DRAM with a WRITE TRANSFER (WT) operation. All of these applications deal with sequential data, as is the case with computer graphics data. However, these applications put as much emphasis on serial input streams as graphics does on output streams.

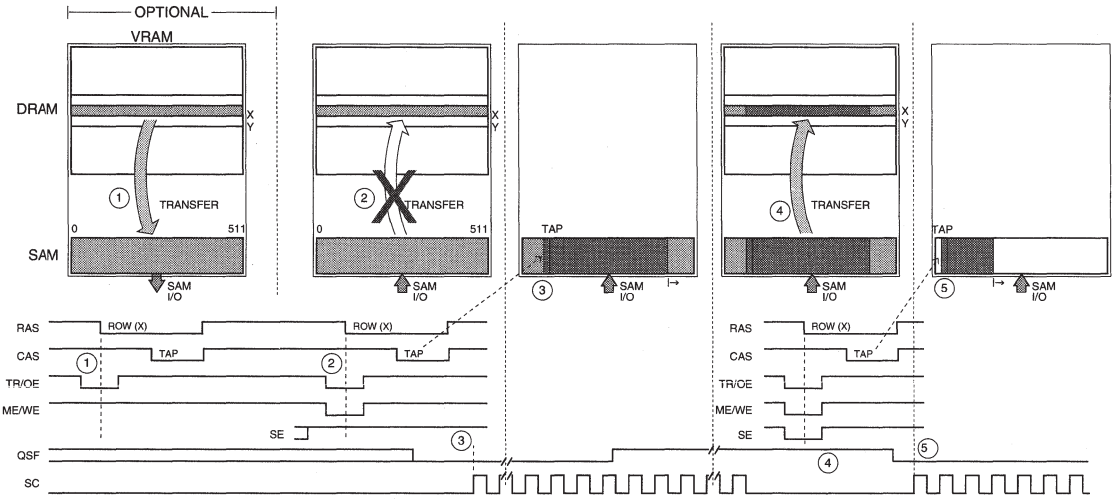

Figure 5
WRITE TRANSFER EXAMPLE

WRITE TRANSFERs are selected in a similar manner to READ TRANSFERs. To perform a WT, the $\overline{ME}/\overline{WE}$ pin must be LOW at the falling edge of $\overline{RAS}$ during the TRANSFER cycle (indicated by $\overline{TR}/\overline{OE}$ = LOW at $\overline{RAS}$), see Figure 5. Before data can be input, the SAM I/O port must be conditioned as an input port. The initial PSEUDO WRITE TRANSFER conditions the SAM I/O port as an input without actually writing data to the DRAM, this is shown in Figure 5 at state ②.

It is possible to preload a row from the DRAM array into the SAM before the PWT (see Figure 5, state ①). This will allow new serial input data to be written over a portion of an existing row in the DRAM array. To perform this operation, an RT is invoked before the PWT, followed by the normal serial input sequence.

After the PSEUDO WRITE TRANSFER has completed, the serial-input data is loaded. This starts at state ③ and progresses until ④. At this time, all the serial data has been loaded and the WRITE TRANSFER is done; the serial clock must be idle. The Tap loaded at the falling edge of $\overline{CAS}$ defines the location where the next string of serial data begins loading, ⑤. An RT and PWT may be inserted between the WT at ④ and the new serial-input at state ⑤.

Note: *The serial clock, SC, must be idle during WRITE TRANSFERs. If it is not, the Tap address will be corrupted.*

WRITE TRANSFERs may also be used to replicate rows within the DRAM. In this operation, a READ TRANSFER is done from the source row to the SAM. Then, without clocking the SAM, a WRITE TRANSFER is done to any row(s) of the DRAM. Once "full," the SAM may be transferred to any row or multiple rows by performing single or multiple WTs. This is useful in accelerating full screen clears. BLOCK WRITE may be used to "color" the source row, further enhancing the performance of this operation.

When selecting a WRITE TRANSFER, the Micron 1 Meg VRAM uses the DSF pin to invoke the ALTERNATE WRITE TRANSFER (AWT) cycle. AWT is a variation of the standard WT, differing in that the $\overline{SE}$ pin, which selects between WT and PWT, is a "don't care" when $\overline{RAS}$ falls. This will change the SAM I/O direction to input, and transfer the contents of the SAM to the addressed row of the DRAM array. The advantage of this cycle is that the SAM I/O control pins (e.g. SC and $\overline{SE}$) are not used to invoke the write transfer, separating the SAM control logic and the WT control logic.

Any of the three WRITE TRANSFER cycles will set the SAM I/O to the input direction so it is not necessary to do a PWT between successive WTs or AWTs. Once the port is set as an input, it will stay in that state until an RT is performed. The PWT cycle is only provided to give designers a way to change the I/O direction without corrupting the DRAM array data when initialing the SAM.

Some 1 Meg VRAM manufacturers use the DSF to select SPLIT WRITE TRANSFER in lieu of AWT. Micron provides SPLIT WRITE TRANSFERS on the MT42C8256/7, 2 Meg VRAM. For a description of the 2 Meg VRAM features, refer to Technical Note TN-42-01.

SAM TIMING CONSIDERATIONS

SAM timing is straightforward. Data is output from, or input to, the SAM port under the control of the serial clock (SC) and serial enable ($\overline{SE}$) pins. The serial clock performs two operations per cycle. First, on the rising edge, it outputs serial data when reading from the SAM, or it latches the serial input data when writing to the SAM. Second, shortly after the rising edge, SC increments the serial address counter/pointer, which holds the Tap address of the SAM. The address counter will increment regardless of the state of $\overline{SE}$, which is purely a data enable/disable pin.

When the SAM is conditioned as an output, the data is accessed by the rising edge of the SC pin. The incremental data will appear on the SDQ pins an access time after the rising edge. The $\overline{SE}$ pin may be used to disable the output, allowing the serial ports of multiple VRAM banks to be tied to a common data bus. QSF is always active on the Micron 1 Meg VRAM and will indicate when the address crosses the SAM-half boundaries (e.g. 255 and 511 for the 42C4256, or 127 and 255 for the 42C8128) during all SAM READ and WRITE operations. Some 1 Meg VRAMs only drive an active QSF during split SAM operation. When the device is in "full" SAM operation, QSF will be High-Z.

When data is written into the SAM, it must be stable for the setup and hold window specified by the data sheets to ensure it is latched accurately by the rising edge of the SC pin. The $\overline{SE}$ pin may be used to disable a WRITE to the SAM for any given clock cycle. When $\overline{SE}$ is disabled (HIGH) the WRITE will be inhibited but the SAM address count will increment. As the SAM is written, the QSF pin may be used as a full or half-full flag similar to those used for first-in-first-out memory devices.

Note: *The Micron 1 Meg VRAM does not use the falling edge of the SC pulse for any internal operations. This allows the clock to be idled in either state without affecting the address pointer location. This is not true of all VRAM manufacturers. To ensure compatibility, the SC should be idled in the LOW state.*

The $\overline{SE}$ pin provides a faster access time than SC and may be used to bank select. When $\overline{SE}$ goes HIGH, the SAM output drivers are designed to turn off as fast as, or two or three nanoseconds faster than, they turn on. This helps to reduce bus contention when switching banks of VRAM with $\overline{SE}$. The actual delta between turn-off and turn-on will vary lot by lot, but any lot variation will be of minor impact. Bus contention will be insignificant, if it exists all, even when using opposite polarities of a single serial enable to select between two banks of VRAM. However, systems that require a guaranteed skew between turn-on and turn-off will require additional logic to ensure that every vendor and device will not cause contention. In this case, an asynchronous state machine may be used to skew the rising and falling edges of $\overline{SE}$ to guarantee the desired separation. A skew of 5ns should yield adequate guardband between turn-off and turn-on time.

HARDWARE PIXEL MASKING

The 1 Meg VRAM supports hardware-level pixel or pixel-bit masking with the MASKED WRITE cycle. This feature can ease system design or improve performance by allowing overlay/underlay planes to be written and operations such as window clipping to be done with single WRITE, or continuous PAGE MODE WRITE cycles.

Standard DRAMs require a READ-MODIFIED-WRITE (RMW) cycle to perform a "write-per-bit" operation. During a RMW, data is read from a location in memory, changed by the processor and written back to the same memory location within the same $\overline{CAS}$ cycle.

MASKED WRITE allows individual DQ bits to be selectively overwritten during a WRITE to the DRAM port. This is done by preloading a write-mask bit for each corresponding DQ bit plane. A mask data register (4 or 8 bits for x4 or x8, respectively) inside the VRAM stores the DQ mask data. A logic 1 in the mask register will enable that bit's corresponding DQ bit (plane) when the data is written to the DRAM after $\overline{CAS}$ goes LOW. A logic 0 will mask data for that DQ plane when the DRAM WRITE is performed. See Figure 2. During non-PAGE MODE operation, MASKED WRITE is approximately 25 to 30 percent faster than an RMW; when operating in FAST-PAGE-MODE, it is at least 50 percent faster.

Two types of MASKED WRITE cycles are supported by the 1 Meg VRAM; nonpersistent and persistent. When nonpersistent MASKED WRITE is selected, mask data must be presented on the DQ pins for every new $\overline{RAS}$ cycle. When $\overline{RAS}$ falls, the data present on the DQ pins is loaded into the mask register. When the WRITE occurs at the falling edge of $\overline{CAS}$ or $\overline{WE}$, the mask is applied to the incoming write data. After $\overline{RAS}$ goes HIGH, terminating the WRITE cycle, the mask register data is lost and must be reloaded for any subsequent $\overline{RAS}$ cycles.

Some graphics processors and controllers are not able to provide mask data at the falling edge of $\overline{RAS}$. To allow these controllers to perform hardware pixel plane protect directly in the VRAM, a persistent MASKED WRITE (read/write old mask; RWOM) cycle is provided. When using this cycle, the mask data must be loaded before the WRITE cycle is done. However, once loaded, the mask register will not be

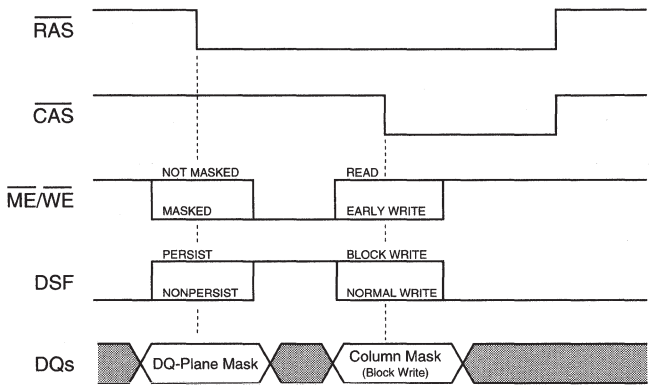

Figure 6
SPECIAL FUNCTION MODE SELECTION

erased at the end of the WRITE cycle. A LOAD MASK REGISTER cycle is performed to permanently load the mask register. During this cycle, data present on the DQ pins will be written to the mask register instead of the DRAM. This mask data will stay in the register until overwritten or until power is lost. The register is fully static and requires no refresh. When the RWOM cycle is performed, the mask is supplied by the mask register. The DQ inputs are "don't care" when $\overline{RAS}$ falls.

The control levels for the special-function DRAM access modes are shown, in a simplified form, in Figure 6. MASKED WRITE and BLOCK WRITE select options are shown. A write mask may be applied to a BLOCK WRITE in the same manner as it is to a normal WRITE. See the MT42C4256 and MT42C8128 data sheets for detailed timing of these cycles.

There is some functional variation between VRAM suppliers regarding these cycles. Some 1 Meg VRAM manufacturers support FLASH WRITE cycles in lieu of persistent MASKED WRITE. The Micron MT42C8256/7, 2 Meg VRAM includes the FLASH WRITE function, see the data sheet and TN-42-01 for more detail.

ACCELERATING WINDOW AND POLYGON FILLS WITH BLOCK WRITE

As the operating systems of workstations and PCs move toward graphical, windowed environments, the ability to quickly "fill" a window with color or text will become more important to display performance. In addition to windowed environments, demands for real-time imaging require improvement of the solid-color fill rate of drawn polygons. To satisfy these requirements, BLOCK WRITE has been added to the VRAM. This feature provides a four-times improvement of pixel drawing speed to the DRAM port of the VRAM. However, there is a penalty for this speed improvement; the pixel data is restricted to a single, preloaded "color" for each cycle. BLOCK WRITE can be used to quickly fill simple background patterns on the screen when the VRAM frame buffer contains more than one pixel per pixel-word.

The color data used during a BW is stored in the color register of the VRAM by the LOAD COLOR REGISTER cycle. This cycle must precede any BLOCK WRITE cycle

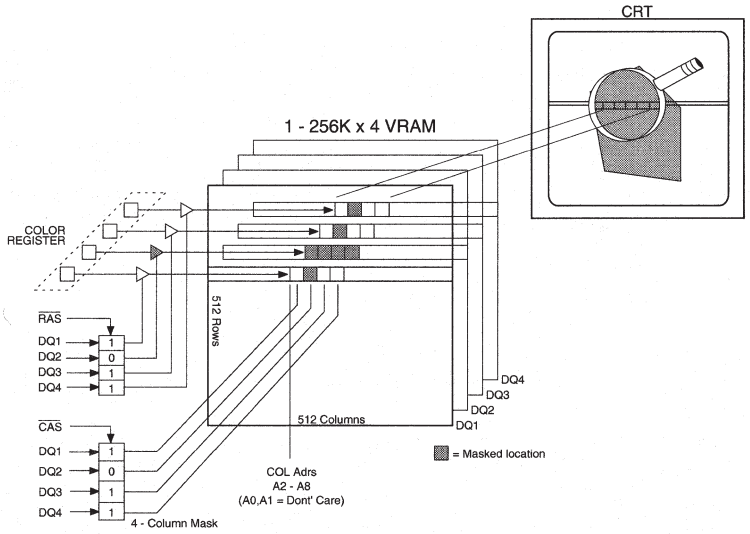

Figure 7
Block Write

because the write data is supplied by the register instead of the DQ pins.

BLOCK WRITE alters the column addressing of the DRAM array. When a BW is performed, the least significant two column-address bits (A0, A1) are ignored. This addressing selects the four-column "block" that will be written to the value stored in the color register. Each bit of the color register corresponds to its associated bit plane and is simultaneously written to all four column locations during the BW, see Figure 7.

BLOCK WRITE is selected by the DSF pin at the falling edge of $\overline{CAS}$ and is the only VRAM special function that is not selected at the falling edge of $\overline{RAS}$. It is important to note that all manufacturers of the 1 Meg VRAM do not perform BW the same way. Some manufacturers (including Micron) perform the BW only at the falling edge of $\overline{CAS}$ and allow $\overline{WE}$ to be a "don't care." The BW control and column mask (DQ pins) are latched only at the falling edge of $\overline{CAS}$. Some manufacturers allow BW to be either $\overline{CAS}$ or $\overline{WE}$ controlled (late BLOCK WRITE). For a robust design, it is recommended that all BWs be done as "early" BW ($\overline{WE}$ LOW prior to $\overline{CAS}$ LOW). A design done in this manner will accept both BW methods.

There is no need to input data at $\overline{CAS}$ time because the pixel data is already stored in the color register. Therefore, a column mask may be applied on the DQ pins when $\overline{CAS}$ falls. Data present on these pins act as separate write enables to each column location within the block, see Figure 7. The least significant DQ corresponds to the least significant column location, the second least significant to the second least column, and so on. This allows column by column addressibility of the window or polygon to be filled during the BW. For example, if the left side of the window starts at the second column location of the "block", the first column may be masked off and columns 2, 3 and 4 of the block will be written with the new color.

Note: *When using the Micron VRAM in "minimum" function applications (no MASKED WRITE, BLOCK WRITE or SPLIT READ TRANSFER), ensure that the DSF pin is grounded or held a logic 0 level.*

CONCLUSION

VRAM function sets and system implementations are very diverse. Your function set and system design may differ from what was presented here. These were typical implementation examples and are not exhaustive. The intent is to show the bandwidth advantages of the VRAM over DRAM and how to use them in a memory design. The VRAM bandwidth advantage exists when dealing with sequential data streams in particular, and longer streams in general. The special features, such as MASKED WRITE and BLOCK WRITE further advance the performance of VRAM in graphics systems. Features such as SPLIT READ TRANSFER simplify high-bandwidth continuous sequential data streams such as those seen in communications and high-bandwidth networks.

If your design question was not answered here, feel free to call Micron Application Engineering at 208-368-3905.