4. TriMatrix Embedded Memory Blocks in Stratix III Devices

SIII51004-1.8

Introduction

TriMatrix embedded memory blocks provide three different sizes of embedded SRAM to efficiently address the needs of Stratix® III FPGA designs. TriMatrix memory includes 640- (in ROM mode only) or 320-bit memory logic array blocks (MLABs), 9-Kbit M9K blocks, and 144-Kbit M144K blocks. The MLABs have been optimized to implement filter delay lines, small first-in first-out (FIFO) buffers, and shift registers. You can use the M9K blocks for general purpose memory applications, and the M144K blocks are ideal for processor code storage, packet buffering, and video frame buffering.

You can independently configure each embedded memory block to be a single- or dual-port RAM, FIFO, ROM, or shift register via the Quartus® II MegaWizard™ Plug-In Manager. You can stitch together multiple blocks of the same type to produce larger memories with minimal timing penalty. TriMatrix memory provides up to 20,491 Kbits of embedded SRAM at up to 600 MHz operation. This chapter describes TriMatrix memory blocks, modes, features, and design considerations.

Overview

Table 4–1 summarizes the features supported by the three sizes of TriMatrix memory.

Table 4–1. Summary of TriMatrix Memory Features (Part 1 of 2)

Feature	MLABs	M9K Blocks	M144K Blocks
Maximum performance	600 MHz	580 MHz	580 MHz
Total memory bits (including parity bits)	640 (in ROM mode) or 320 (in other modes)	9,216	147,456
Configurations (depth × width) *(1)*	16 × 8 16 × 9 16 × 10 16 × 16 16 × 18 16 × 20	8 K × 1 4 K × 2 2 K × 4 1 K × 8 1 K × 9 512 × 16 512 × 18 256 × 32 256 × 36	16 K × 8 16 K × 9 8 K × 16 8 K × 18 4 K × 32 4 K × 36 2 K × 64 2 K × 72
Parity bits	✓	✓	✓

Table 4–1. Summary of TriMatrix Memory Features (Part 2 of 2)

Feature	MLABs	M9K Blocks	M144K Blocks
Byte-enable	✓	✓	✓
Packed mode	—	✓	✓
Address clock enable	✓	✓	✓
Single-port memory	✓	✓	✓
Simple dual-port memory	✓	✓	✓
True dual-port memory	—	✓	✓
Embedded shift register	✓	✓	✓
ROM	✓	✓	✓
FIFO buffer	✓	✓	✓
Simple dual-port mixed width support	—	✓	✓
True dual-port mixed width support	—	✓	✓
Memory initialization file (.mif)	✓	✓	✓
Mixed-clock mode	✓	✓	✓
Power-up condition	Outputs cleared if registered, otherwise reads memory contents	Outputs cleared	Outputs cleared
Register clears	Output registers	Output registers	Output registers
Asynchromous clear on output latch	—	✓	✓
Write/Read operation triggering	Write: Falling clock edges Read: Rising clock edges	Write and Read: Rising clock edges	Write and Read: Rising clock edges
Same-port read-during-write	Outputs set to don't care	Outputs set to old or new data	Outputs set to old or new data
Mixed-port read-during-write	Outputs set to old data or don't care	Outputs set to old data or don't care	Outputs set to old data or don't care
ECC Support	Soft IP support via Quartus II software	Soft IP support via Quartus II software	Built-in support in ×64 wide SDP mode or soft IP support via Quartus II software

Notes to Table 4–1:

(1) In ROM mode, MLABs support the (depth × width) configurations of 64×8, 64×9, 64×10, 32×16, 32×18, or 32× 20.

(2) MLABs support byte-enable via emulation.

Table 4–2 shows the capacity and distribution of the TriMatrix memory blocks for each Stratix III family member

Table 4–2. TriMatrix Memory Capacity and Distribution in Stratix III Devices

Device	MLABs	M9K Blocks	M144K Blocks	Total Dedicated RAM Bits (dedicated memory blocks only)	Total RAM Bits (including MLABs) *(1)*
EP3SL50	950	108	6	1,836 Kb	2,133 Kb
EP3SL70	1,350	150	6	2,214 Kb	2,636 Kb
EP3SL110	2,150	275	12	4,203 Kb	4,875 Kb
EP3SL150	2,850	355	16	5,499 Kb	6,390 Kb
EP3SL200	4,000	468	36	9,396 Kb	10,646 Kb
EP3SL340	6,750	1,040	48	16,272 Kb	18,381 Kb
EP3SE50	950	400	12	5,328 Kb	5,625 Kb
EP3SE80	1,600	495	12	6,183 Kb	6,683 Kb
EP3SE110	2,150	639	16	8,055 Kb	8,727 Kb
EP3SE260	5,100	864	48	14,688 Kb	16,282 Kb

Note to Table 4–2:

(1) For total ROM Kbits, use this equation to calculate:
Total ROM Kbits = Total Embedded RAM Kbits + [(number of MLAB blocks × 640)/1024]

TriMatrix Memory Block Types

While the M9K and M144K memory blocks are dedicated resources, the MLABs are dual-purpose blocks. They can be configured as regular logic array blocks (LABs) or as memory logic array blocks (MLABs). Ten adaptive logic modules (ALMs) make up one MLAB. Each ALM in an MLAB can be configured as a 16 × 2 block, resulting in a 16 × 20 simple dual-port SRAM block in a single MLAB. In ROM mode, each ALM in an MLAB can be configured as either a 64 × 1 or a 32 × 2 block, resulting in a 64 × 10 or 32 × 20 ROM block in a single MLAB.

All the ALMs share the same address bits. Therefore, you cannot combine multiple memories with different address bits and implement them in a single MLAB.

When you are using an MLAB as memory, you will not be able to use the unused ALMs in the MLAB even if you do not use the full capacity of an MLAB.

Parity Bit Support

All TriMatrix memory blocks have built-in parity-bit support. The ninth bit associated with each byte can store a parity bit or serve as an additional data bit. No parity function is actually performed on the ninth bit.

Byte-Enable Support

All TriMatrix memory blocks support byte-enables that mask the input data so that only specific bytes of data are written. The unwritten bytes retain the previous written value. The write enable (`wren`) signals, along with the byte-enable (`byteena`) signals, control the RAM blocks' write operations.

MLABs support byte-enable via emulation. There will be increased logic utilization when the byte-enables are emulated.

The default value for the byte-enable signals is high (enabled), in which case writing is controlled only by the write enable signals. The byte-enable registers have no clear port. When using parity bits on the M9K and M144K blocks, the byte-enable controls all nine bits (eight bits of data plus one parity bit). When using parity bits on the MLAB, the byte-enable controls all 10 bits in the widest mode.

Byte-enables operate in a one-hot fashion, with the LSB of the `byteena` signal corresponding to the least significant byte of the data bus. For example, if you are using a RAM block in ×18 mode, with `byteena = 01`, `data[8..0]` is enabled and `data[17..9]` is disabled. Similarly, if `byteena = 11`, both `data[8..0]` and `data[17..9]` are enabled. Byte-enables are active high.

You cannot use the byte-enable feature when using the ECC feature on M144K blocks.

Figure 4–1 shows how the write enable (`wren`) and byte-enable (`byteena`) signals control the operations of the M9K and M144K.

When a byte-enable bit is de-asserted during a write cycle, the corresponding data byte output can appear as either a "don't care" value or the current data at that location. The output value for the masked byte is controllable via the Quartus II software. When a byte-enable bit is asserted during a write cycle, the corresponding data byte output also depends on the setting chosen in the Quartus II software.

Figure 4–1. Stratix III Byte-Enable Functional Waveform for M9K and M144K

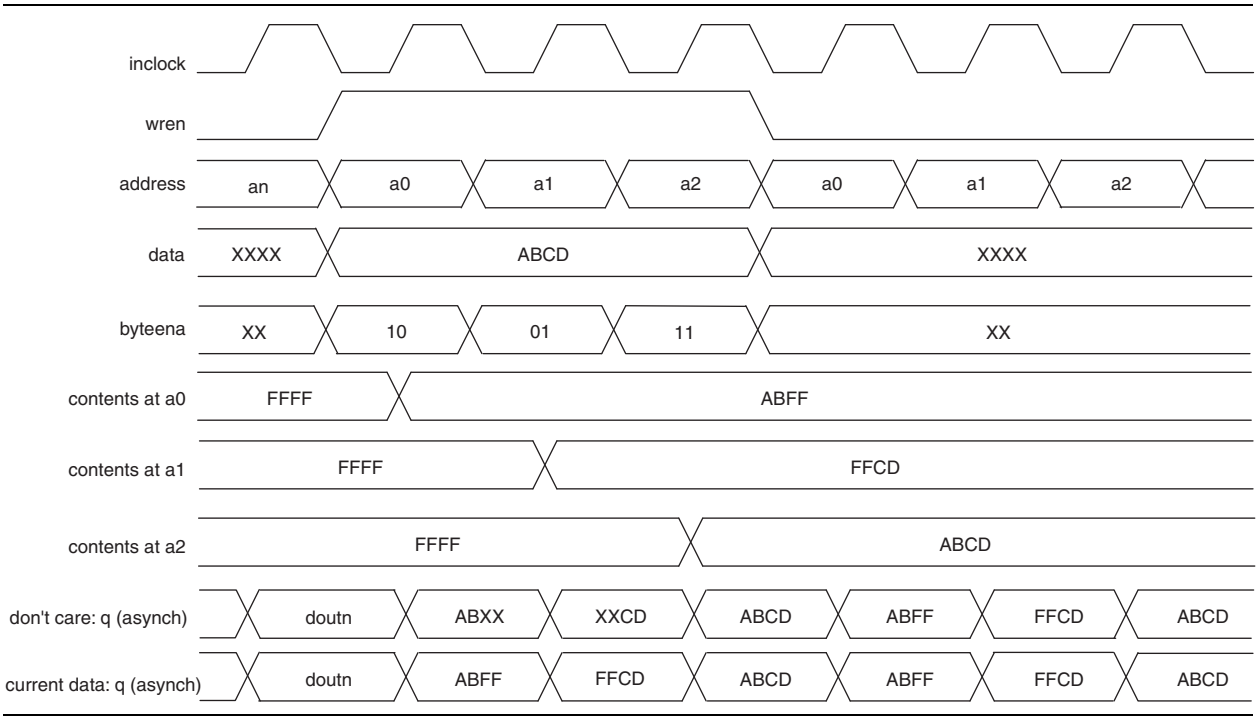
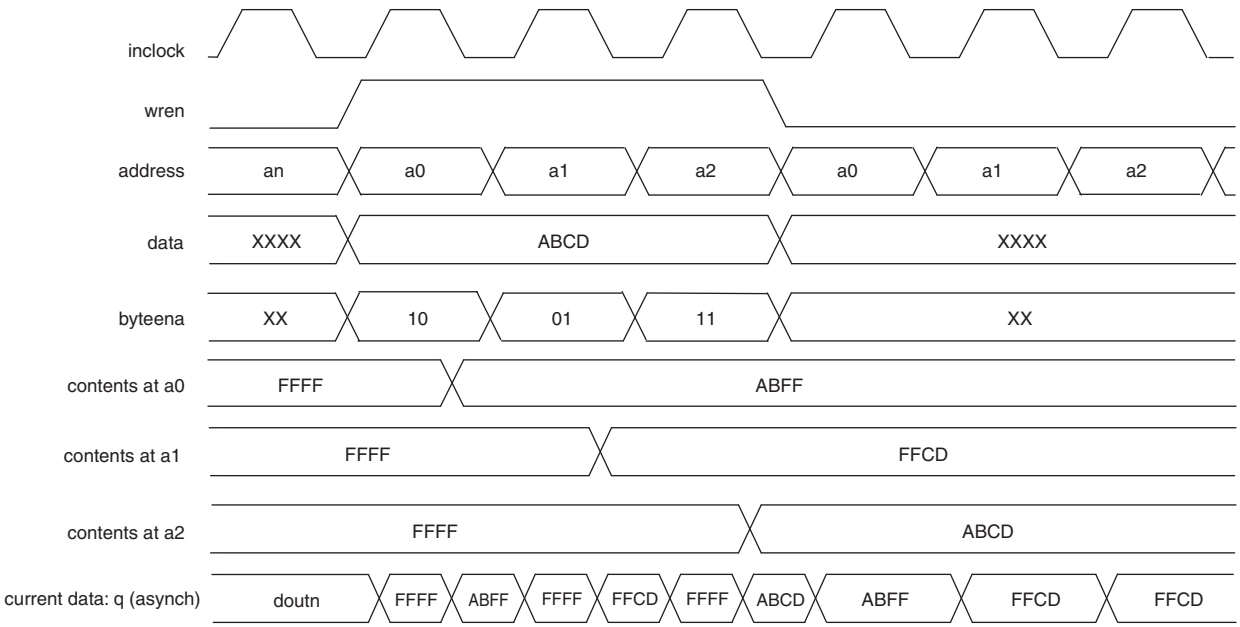


Figure 4–2 shows how the write enable (`wren`) and byte-enable (`byteena`) signals control the operations of the MLABs. The write operation in MLABs is triggered by failing clock edges.

Figure 4–2. Stratix III Byte-Enable Functional Waveform for MLABs



Packed Mode Support

Stratix III M9K and M144K blocks support packed mode. The packed mode feature packs two independent single-port RAMs into one memory block. The Quartus II software automatically implements packed mode where appropriate by placing the physical RAM block into true dual-port mode and using the MSB of the address to distinguish between the two logical RAMs. The size of each independent single-port RAM must not exceed half of the target block size.

Address Clock Enable Support

All Stratix III memory blocks support address clock enable, which holds the previous address value for as long as the signal is enabled (`addressstall = 1`). When the memory blocks are configured in dual-port mode, each port has its own independent address clock enable. The default value for the address clock enable signals is low (disabled).

Figure 4–3 shows an address clock enable block diagram. The address clock enable is referred to by the port name `addressstall`.

Figure 4–3. Stratix III Address Clock Enable Block Diagram

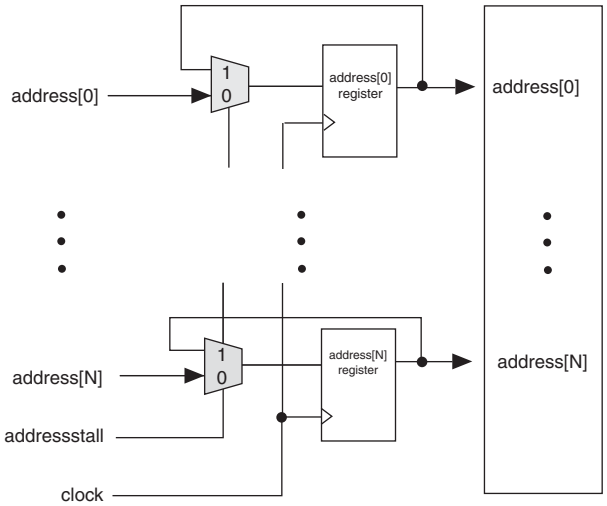


Figure 4–4 shows the address clock enable waveform during the read cycle.

Figure 4–4. Stratix III Address Clock Enable during Read Cycle Waveform

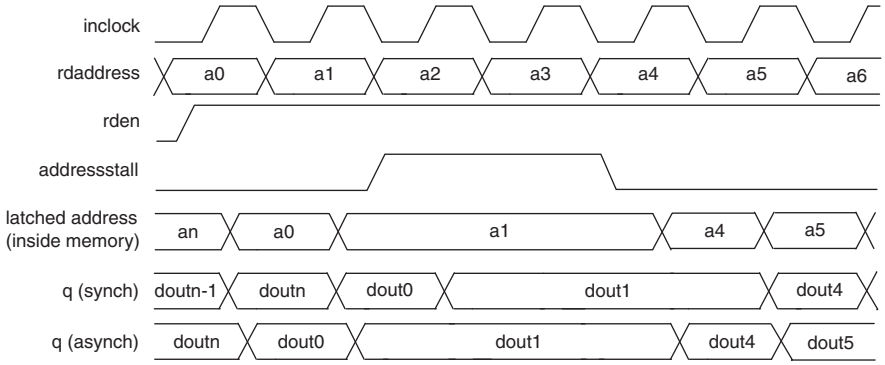


Figure 4–5 shows the address clock enable waveform during the write cycle for M9K and M144K.

Figure 4–5. Stratix III Address Clock Enable during Write Cycle Waveform for M9K and M144K

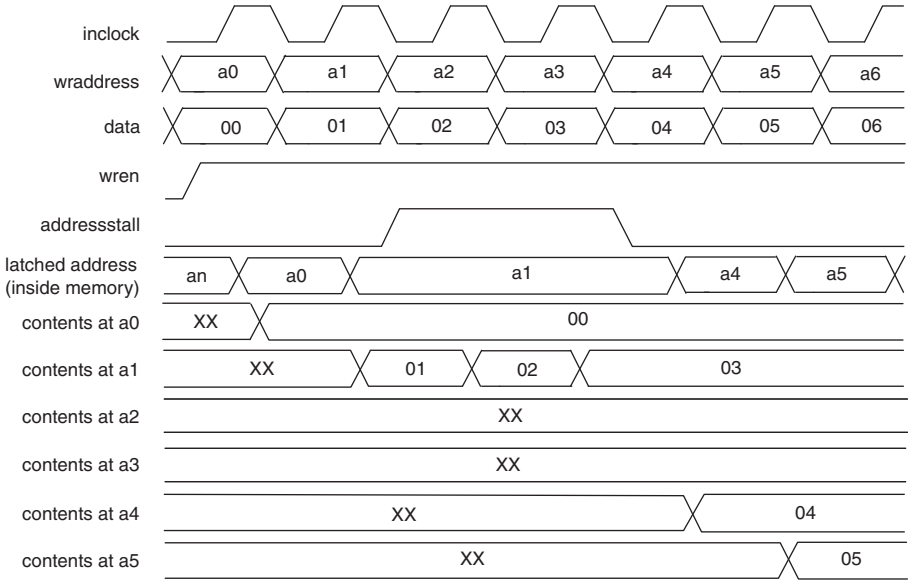
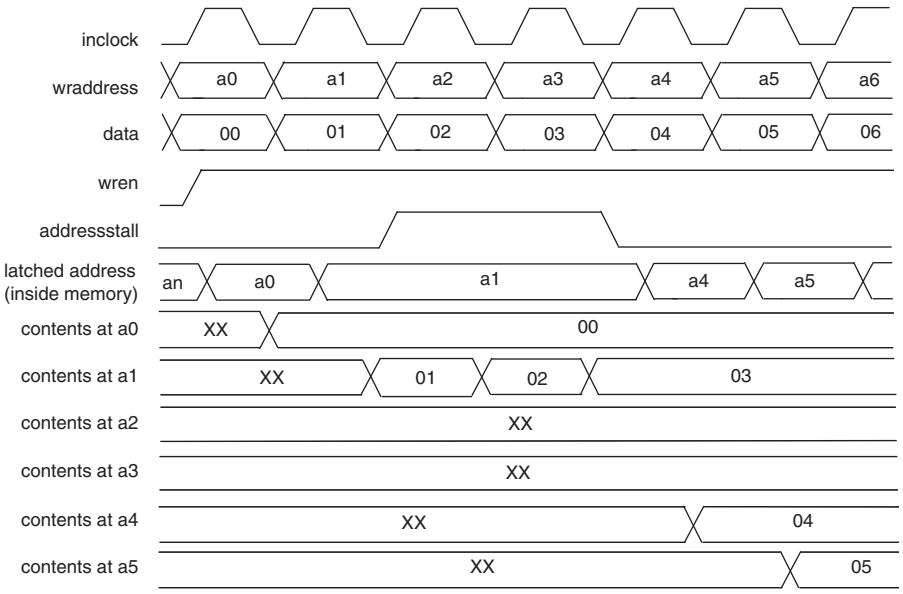


Figure 4–6 shows the address clock enable waveform during the write cycle for MLABs.

Figure 4–6. Stratix III Address Clock Enable during Write Cycle Waveform for MLABs



Mixed Width Support

M9K and M144K memory blocks inherently support mixed data widths. MLABs can support mixed data widths through emulation via the Quartus II software. When using simple dual-port or true dual-port mixed width support allows you to read and write different data widths to a memory block. Refer to "Memory Modes" on page 4–10 for details on the different widths supported per memory mode.

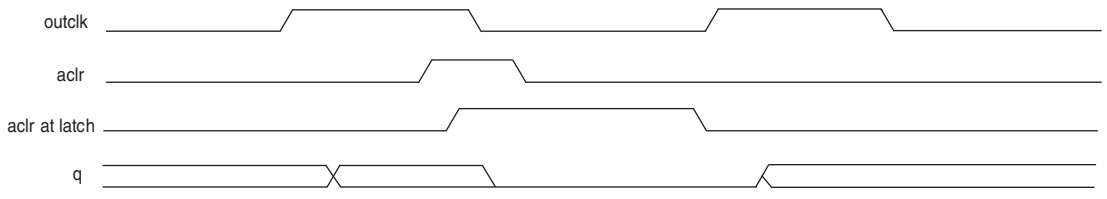
You cannot use the ECC on M144 memory blocks when using the mixed width support.

MLABs do not support mixed-width FIFO mode.

Asynchronous Clear

Stratix III M9K and M144K memory blocks support asynchronous clears on the output latches and output registers. MLABs supports asynchronous clear on the output registers only as the output is not latched. Therefore, if your M9K and M144K are not using the output registers, you can still clear the RAM outputs via the output latch asynchronous clear. The functional waveform in Figure 4–7 shows this functionality.

Figure 4–7. Output Latch Asynchronous Clear Waveform



You can selectively enable asynchronous clears per logical memory via the Quartus II RAM MegaWizard Plug-In Manager.

For more information, refer to the *RAM Megafunction User Guide*.

Error Correction Code Support

Stratix III M144K blocks have built-in support for error correction code (ECC) when in ×64-wide simple dual-port mode. ECC allows you to detect and correct data errors in the memory array. The M144K blocks have a single-error-correction double-error-detection (SECDED) implementation. SECDED can detect and fix a single-bit error in a 64-bit word or detect two-bit errors in a 64-bit word. It cannot detect three or more errors.

The M144K ECC status is communicated via a three-bit status flag `eccstatus[2..0]`. The status flag can be either registered or unregistered. When registered, it uses the same clock and asynchronous clear signals as the output registers. When not registered, it cannot be asynchronously cleared.

Table 4–3 shows the truth table for the ECC status flags.

Table 4–3. Truth Table for ECC Status Flags

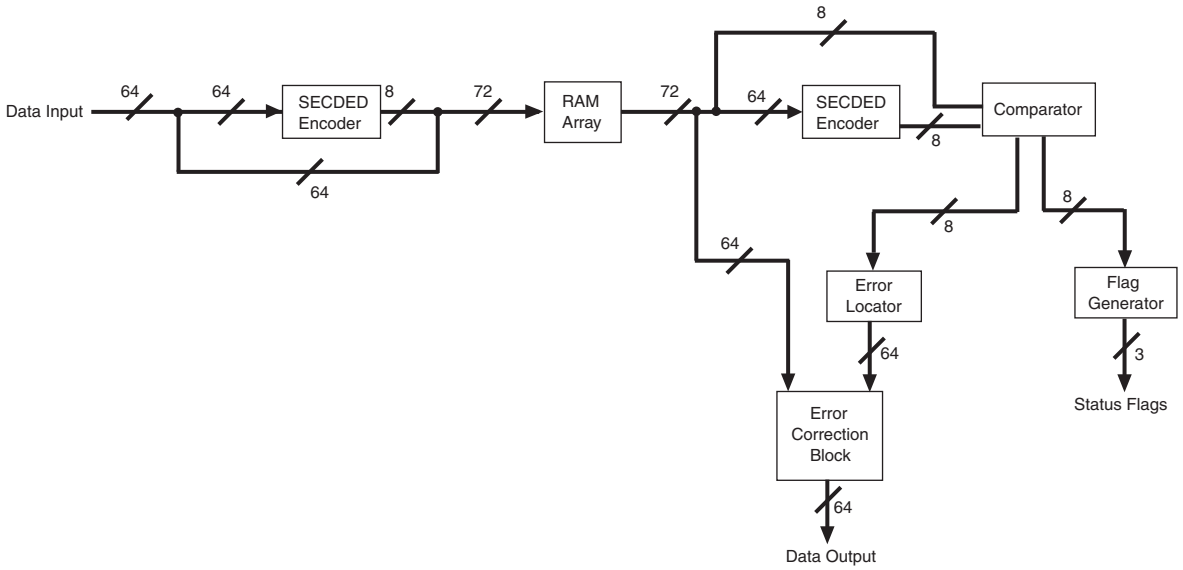
Status	eccstatus[2]	eccstatus[1]	eccstatus[0]
No error	0	0	0
Single error and fixed	0	1	1
Double error and no fix	1	0	1
Illegal	0	0	1
Illegal	0	1	0
Illegal	1	0	0
Illegal	1	1	X

You cannot use the byte-enable feature when ECC is engaged.

Read during write “old data” mode is not supported when ECC is engaged.

Figure 4–8 shows a block diagram of the ECC block of the M144K.

Figure 4–8. ECC Block Diagram of the M144K



Memory Modes

Stratix III TriMatrix memory blocks allow you to implement fully synchronous SRAM memory in multiple modes of operation. M9K and M144K blocks do not support asynchronous memory (unregistered inputs). MLABs support asynchronous (flow-through) read operations.

Depending on which TriMatrix memory block you target, the following modes may be used:

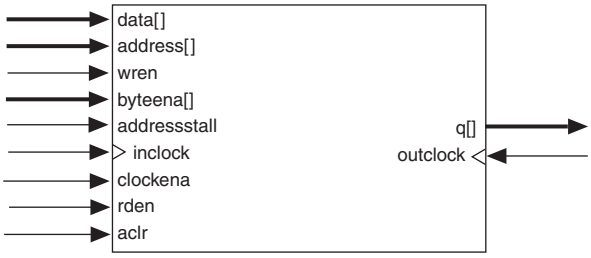
- Single-port
- Simple dual-port
- True dual-port
- Shift-register
- ROM
- FIFO

When using the memory blocks in ROM, single-port, simple dual-port, or true dual-port mode, you can corrupt the memory contents if you violate the setup or hold-time on any of the memory block input registers. This applies to both read and write operations.

Single Port RAM

All TriMatrix memory blocks support single-port mode. Single-port mode allows you to do either one read or one write operation at a time. Simultaneous reads and writes are not supported in single-port mode. Figure 4–9 shows the single-port RAM configuration.

Figure 4–9. Single-Port Memory *(Note 1)*



Note to Figure 4–9:

(1) You can implement two single-port memory blocks in a single M9K or M144K block. See "Packed Mode Support" on page 4–5 for more details.

During a write operation, behavior of the RAM outputs is configurable. If you use the read-enable signal and perform a write operation with the read enable deactivated, the RAM outputs retain the values they held during the most recent active read enable. If you activate read enable during a write operation, or if you are not using the read-enable signal at all, the RAM outputs either show the new data being written, the old data at that address, or a don't care value. To choose the desired behavior, set the read-during-write behavior to either new data, old data, or don't care in the RAM MegaWizard Plug-In Manager in the Quartus II software. See "Read During Write" on page 4–21 for more details on this behavior.

Table 4–4 shows the possible port width configurations for TriMatrix memory blocks in single-port mode.

Table 4–4. Stratix III Port Width Configurations for MLABs, M9K Blocks, and M144K Blocks (Single-Port Mode)

Port Width	MLABs *(1)*	M9K Blocks	M144K Blocks
Port Width Configurations	16 × 8	8 K × 1	16 K × 8
	16 × 9	4 K × 2	16 K × 9
	16 × 10	2 K × 4	8 K × 16
	16 × 16	1 K × 8	8 K × 18
	16 × 18	1 K × 9	4 K × 32
	16 × 20	512 × 16	4 K × 36
		512 × 18	2 K × 64
		256 × 32	2 K × 72
		256 × 36	

Note to Table 4–4:

(1) Configurations of 64 × 8, 64 × 9, 64 × 10, 32 × 16, 32 × 18, and 32 × 20 are supported by stitching multiple MLAB blocks.

Figure 4–10 shows the timing waveforms for read and write operations in single-port mode with unregistered outputs for M9K and M144K. In M9K and M144K registering the RAM's outputs would simply delay the q output by one clock cycle.

Figure 4–10. Timing Waveform for Read-Write Operations (Single-Port Mode) for M9K and M144K

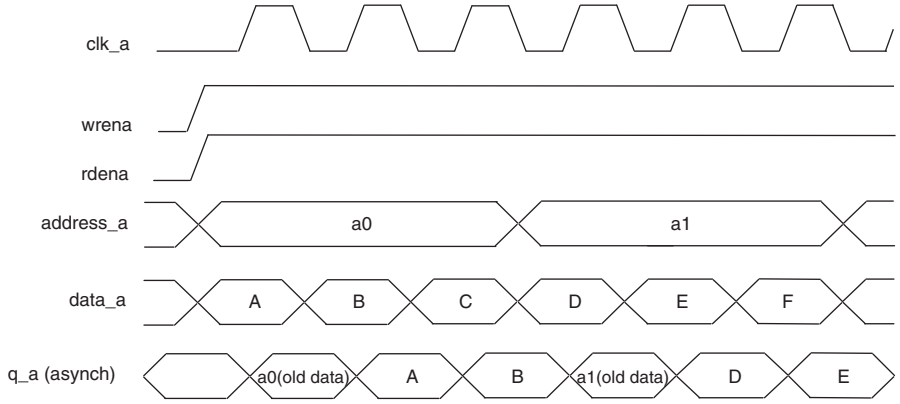
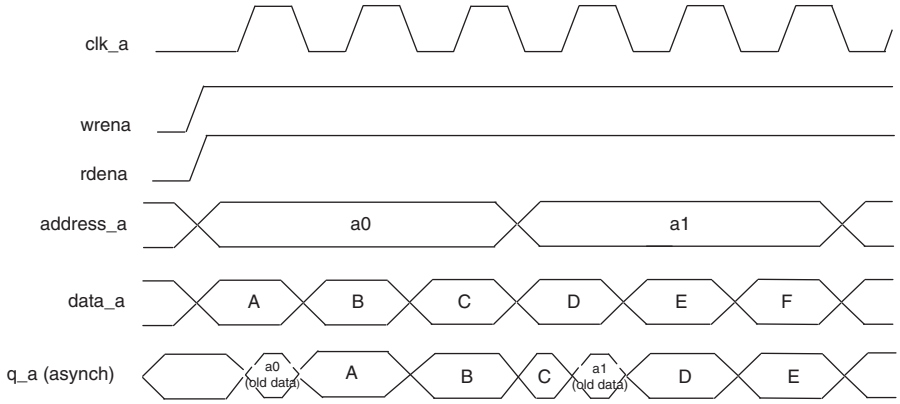


Figure 4–11 shows the timing waveforms for read and write operations in single-port mode with unregistered outputs for MLABs. For MLABs, the read operation is triggered by the rising clock edges whereas the write operation is triggered by the falling clock edges.

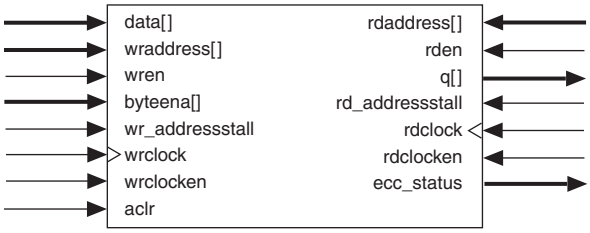
Figure 4–11. Timing Waveform for Read-Write Operations (Single-Port Mode) for MLABs



Simple Dual-Port Mode

All TriMatrix memory blocks support simple dual-port mode. Simple dual-port mode allows you to perform one-read and one-write operation to different locations at the same time. Figure 4–12 shows the simple dual-port configuration.

Figure 4–12. Stratix III Simple Dual-Port Memory *(Note 1)*



Note to Figure 4–12:

(1) Simple dual-port RAM supports input/output clock mode in addition to the read/write clock mode shown.

Simple dual-port mode supports different read and write data widths (mixed width support). Table 4–5 shows the mixed width configurations for the M9K blocks in simple dual-port mode. MLABs do not have native support for mixed width operation. The Quartus II software can implement mixed width memories in MLABs by using more than one MLAB.

Table 4–5. Stratix III M9K Block Mixed-Width Configurations (Simple Dual-Port Mode)

Read Port	Write Port								
	8K×1	**4K×2**	**2K×4**	**1K×8**	**512×16**	**256×32**	**1K×9**	**512×18**	**256×36**
8K×1	✓	✓	✓	✓	✓	✓	—	—	—
4K×2	✓	✓	✓	✓	✓	✓	—	—	—
2K×4	✓	✓	✓	✓	✓	✓	—	—	—
1K×8	✓	✓	✓	✓	✓	✓	—	—	—
512×16	✓	✓	✓	✓	✓	✓	—	—	—
256×32	✓	✓	✓	✓	✓	✓	—	—	—
1K×9	—	—	—	—	—	—	✓	✓	✓
512×18	—	—	—	—	—	—	✓	✓	✓
256×36	—	—	—	—	—	—	✓	✓	✓

Table 4–6 shows the mixed width configurations for the M144K blocks in simple dual-port mode.

Table 4–6. Stratix III M144K Block Mixed-Width Configurations (Simple Dual-Port Mode)

Read Port	Write Port							
	16K×8	**8K×16**	**4K×32**	**2K×64**	**16K×9**	**8K×18**	**4K×36**	**2K×72**
16K×8	✓	✓	✓	✓	—	—	—	—
8K×16	✓	✓	✓	✓	—	—	—	—
4K×32	✓	✓	✓	✓	—	—	—	—
2K×64	✓	✓	✓	✓	—	—	—	—
16K×9	—	—	—	—	✓	✓	✓	✓
8K×18	—	—	—	—	✓	✓	✓	✓
4K×36	—	—	—	—	✓	✓	✓	✓
2K×72	—	—	—	—	✓	✓	✓	✓

In simple dual-port mode, M9K and M144K blocks support separate write-enable and read-enable signals. You can save power by keeping the read-enable signal low (inactive) when not reading. Read-during-write operations to the same address can either output a don't care value or old data. To choose the desired behavior, set the read-during-write behavior to either don't care or old data in the RAM MegaWizard Plug-In Manager in the Quartus II software. See "Read During Write" on page 4–21 for more details about this behavior.

MLABs only support a write-enable signal. Read-during-write behavior for the MLABs can be either don't care, new data, or old data. The available choices depend on the configuration of the MLAB.

Figure 4–13 shows the timing waveforms for read and write operations in simple dual-port mode with unregistered outputs in M9K and M144K. Registering the RAM's outputs would simply delay the q output by one clock cycle in M9k and M144K.

Figure 4–13. Stratix III Simple Dual-Port Timing Waveforms for M9K and M144K

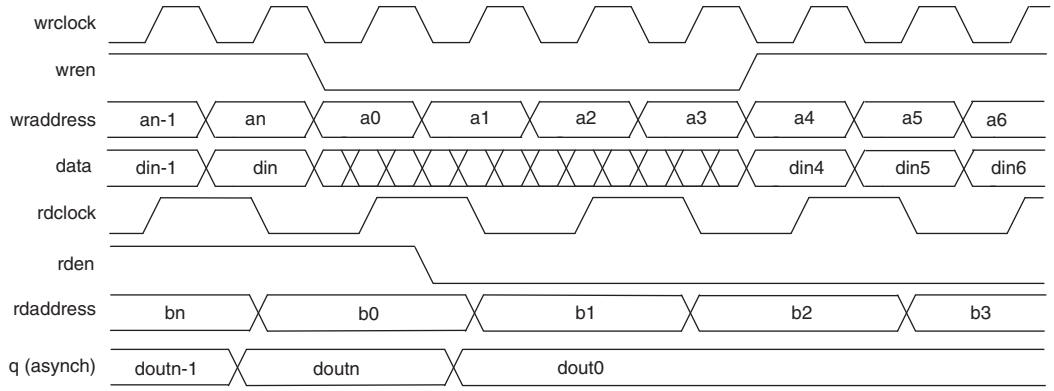
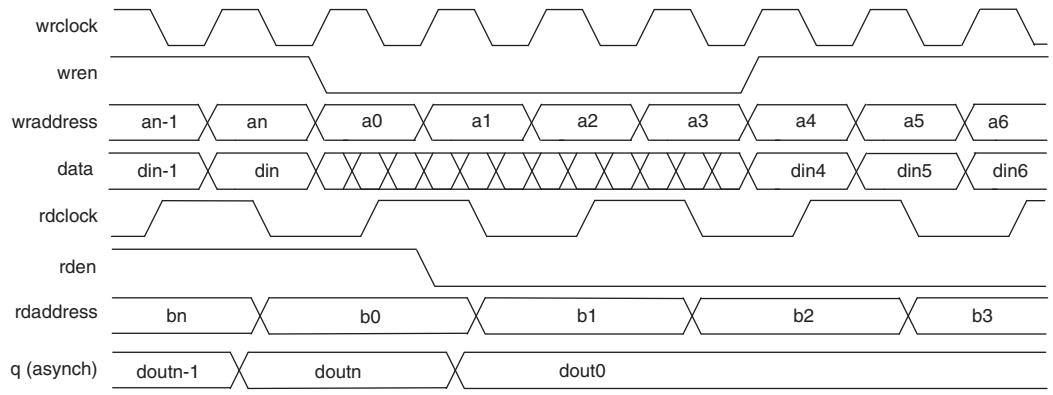


Figure 4–14 shows the timing waveforms for read and write operations in simple dual-port mode with unregistered outputs in MLABs. In MLABs, the write operation is triggered by the falling clock edges.

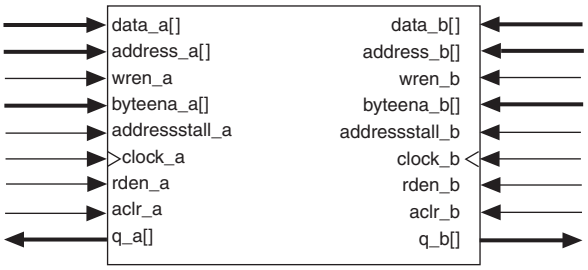
Figure 4–14. Stratix III Simple Dual-Port Timing Waveforms for MLABs



True Dual-Port Mode

Stratix III M9K and M144K blocks support true dual-port mode. Sometimes called bi-directional dual-port, this mode allows you to perform any combination of two port operations: two reads, two writes, or one read and one write at two different clock frequencies. Figure 4–15 shows the true dual-port RAM configuration.

Figure 4–15. Stratix III True Dual-Port Memory *(Note 1)*



Note to Figure 4–15:

(1) True dual-port memory supports input/output clock mode in addition to the independent clock mode shown.

The widest bit configuration of the M9K and M144K blocks in true dual-port mode is as follows:

- 512 × 16-bit (×18-bit with parity) (M9K)
- 4K × 32-bit (×36-bit with parity) (M144K)

Wider configurations are unavailable because the number of output drivers is equivalent to the maximum bit width of the respective memory block. Because true dual-port RAM has outputs on two ports, its maximum width equals half of the total number of output drivers. Table 4–7 lists the possible M9K block mixed-port width configurations in true dual-port mode.

Table 4–7. Stratix III M9K Block Mixed-Width Configuration (True Dual-Port Mode)

Read Port	Write Port						
	8K×1	4K×2	2K×4	1K×8	512×16	1K×9	512×18
8K×1	✓	✓	✓	✓	✓	—	—
4K×2	✓	✓	✓	✓	✓	—	—
2K×4	✓	✓	✓	✓	✓	—	—
1K×8	✓	✓	✓	✓	✓	—	—
512×16	✓	✓	✓	✓	✓	—	—
1K×9	—	—	—	—	—	✓	✓
512×18	—	—	—	—	—	✓	✓

Table 4–8 lists the possible M144K block mixed-port width configurations in true dual-port mode.

Table 4–8. Stratix III M144K Block Mixed-Width Configurations (True Dual-Port Mode)

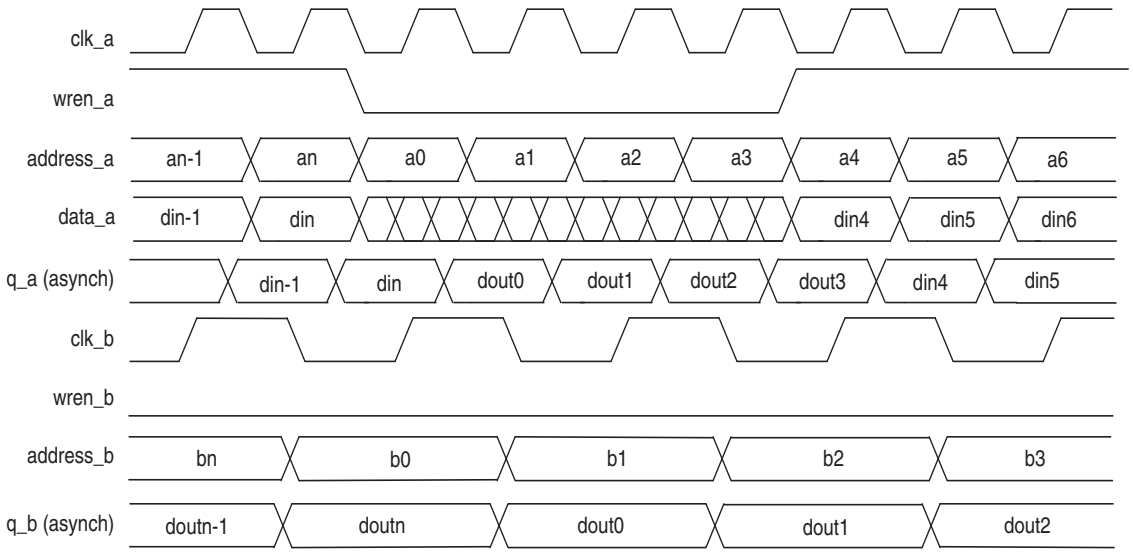
Read Port	Write Port					
	16K×8	8K×16	4K×32	16K×9	8K×18	4K×36
16K×8	✓	✓	✓	—	—	—
8K×16	✓	✓	✓	—	—	—
4K×32	✓	✓	✓	—	—	—
16K×9	—	—	—	✓	✓	✓
8K×18	—	—	—	✓	✓	✓
4K×36	—	—	—	✓	✓	✓

In true dual-port mode, M9K and M144K blocks support separate write-enable and read-enable signals. You can save power by keeping the read-enable signal low (inactive) when not reading. Read-during-write operations to the same address can either output new data at that location or old data. To choose the desired behavior, set the read-during-write behavior to either new data or old data in the RAM MegaWizard Plug-In Manager in the Quartus II software. See "Read During Write" on page 4–21 for more details about this behavior.

In true dual-port mode you can access any memory location at any time from either port. When accessing the same memory location from both ports, you must avoid possible write conflicts. A write conflict happens when you attempt to write to the same address location from both ports at the same time. This results in unknown data being stored to that address location. No conflict resolution circuitry is built into the Stratix III TriMatrix memory blocks. You must handle address conflicts external to the RAM block.

Figure 4–16 shows the true dual-port timing waveforms for the write operation at port A and read operation at port B with the Read-During-Write behavior set to new data. Registering the RAM's outputs would simply delay the q outputs by one clock cycle.

Figure 4–16. Stratix III True Dual-Port Timing Waveform



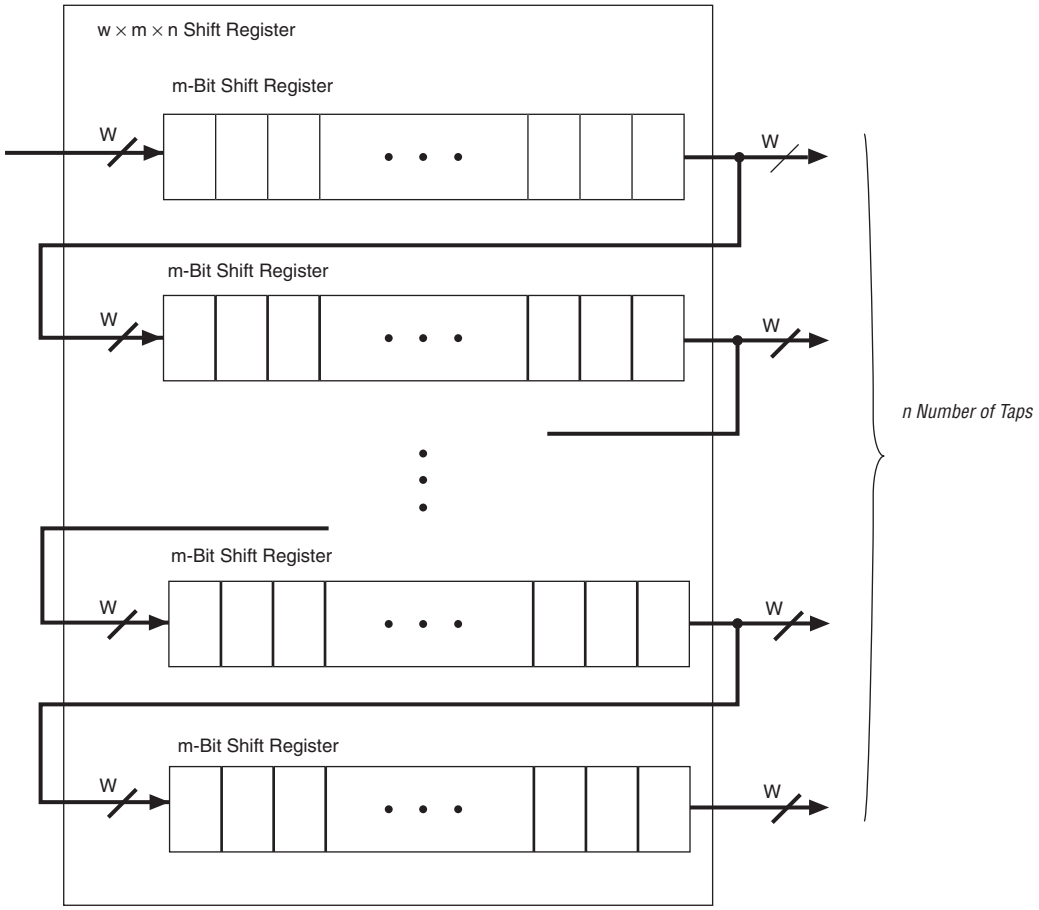
Shift-Register Mode

All Stratix III memory blocks support shift register mode. Embedded memory block configurations can implement shift registers for digital signal processing (DSP) applications, such as finite impulse response (FIR) filters, pseudo-random number generators, multi-channel filtering, and auto- and cross-correlation functions. These and other DSP applications require local data storage, traditionally implemented with standard flipflops that quickly exhaust many logic cells for large shift registers. A more efficient alternative is to use embedded memory as a shift-register block, which saves logic cell and routing resources.

The size of a shift register (`w × m × n`) is determined by the input data width (`w`), the length of the taps (`m`), and the number of taps (`n`). You can cascade memory blocks to implement larger shift registers.

Figure 4–17 shows the TriMatrix memory block in shift-register mode.

Figure 4–17. Stratix III Shift-Register Memory Configuration



ROM Mode

All Stratix III TriMatrix memory blocks support ROM mode. A **.mif** file initializes the ROM content of these blocks. The address lines of the ROM are registered on M9K and M144K blocks, but can be unregistered on MLABs. The outputs can be registered or unregistered. Output registers can be asynchronously cleared. The ROM read operation is identical to the read operation in the single-port RAM configuration.

FIFO Mode

All TriMatrix memory blocks support FIFO mode. MLABs are ideal for designs with many small, shallow FIFO buffers. To implement FIFO buffers in your design, use the Quartus II software FIFO MegaWizard Plug-In Manager. Both single and dual-clock (asynchronous) FIFOs are supported.

For more information about implementing FIFO buffers, refer to the *Single- and Dual-Clock FIFO Megafunctions User Guide*.

MLABs do not support mixed-width FIFO mode.

Clocking Modes

Stratix III TriMatrix memory blocks support the following clocking modes:

- Independent
- Input/output
- Read/write
- Single clock

Violating the setup or hold time on the memory block address registers could corrupt the memory contents. This applies to both read and write operations.

Altera recommends using a memory block clock that comes through global clock routing from an on-chip PLL set to 50 % output duty cycle to achieve the maximum memory block performance. Use Quartus II to report timing for this and other memory block clocking schemes.

For more information refer to the *Stratix III Device Family Errata Sheet*.

Table 4–9 shows the clocking mode versus memory mode support matrix.

Table 4–9. Stratix III TriMatrix Memory Clock Modes

Clocking Mode	True Dual-Port Mode	Simple Dual-Port Mode	Single-Port Mode	ROM Mode	FIFO Mode
Independent	✓	—	—	✓	—
Input/output	✓	✓	✓	✓	—
Read/write	—	✓	—	—	✓
Single clock	✓	✓	✓	✓	✓

Independent Clock Mode

Stratix III TriMatrix memory blocks can implement independent clock mode for true dual-port memories. In this mode, a separate clock is available for each port (A and B). Clock A controls all registers on the port A side, while clock B controls all registers on the port B side. Each port also supports independent clock enables for port A and port B registers. Asynchronous clears are supported only for output latches and output registers on both ports.

Input/Output Clock Mode

Stratix III TriMatrix memory blocks can implement input/output clock mode for true and simple dual-port memories. In this mode, an input clock controls all registers related to the data input to the memory block, including data, address, byte-enables, read enables, and write enables. An output clock controls the data output registers. Asynchronous clears are available on output latches and output registers only.

Read/Write Clock Mode

Stratix III TriMatrix memory blocks can implement read/write clock mode for simple dual-port memories. In this mode, a write clock controls the data-input, write-address, and write-enable registers. Similarly, a read clock control the data-output, read-address, and read-enable registers. The memory blocks support independent clock enables for both the read and write clocks. Asynchronous clears are available on data output latches and registers only.

When using read/write mode, if you perform a simultaneous read/write to the same address location, the output read data will be unknown. If you require the output data to be a known value in this case, use either single-clock mode or input/output clock mode and choose the appropriate read-during-write behavior in the Megawizard.

Single Clock Mode

Stratix III TriMatrix memory blocks can implement single-clock mode for true dual-port, simple dual-port, and single-port memories. In this mode, a single clock, together with a clock enable, is used to control all registers of the memory block. Asynchronous clears are available on output latches and output registers only.

Design Considerations

This section describes guidelines for designing with TriMatrix memory blocks.

Selecting TriMatrix Memory Blocks

The Quartus II software automatically partitions user-defined memory into embedded memory blocks by taking into account both speed and size constraints placed on your design. For example, the Quartus II software may spread out a memory across multiple memory blocks when resources are available to increase the performance of the design. You can manually assign the memory to a specific block size via the RAM MegaWizard Plug-In Manager.

MLABs can implement single-port SRAM through emulation via the Quartus II software. Emulation results in minimal additional logic resources being used. Because of the dual-purpose architecture of the MLAB, it only has data input registers and output registers in the block. MLABs gain input address registers and additional optional data output registers from adjacent ALMs by using register packing.

For more information about register packing, refer to the *Logic Array Blocks and Adaptive Logic Modules in Stratix III Devices* chapter in volume 1 of the *Stratix III Device Handbook*.

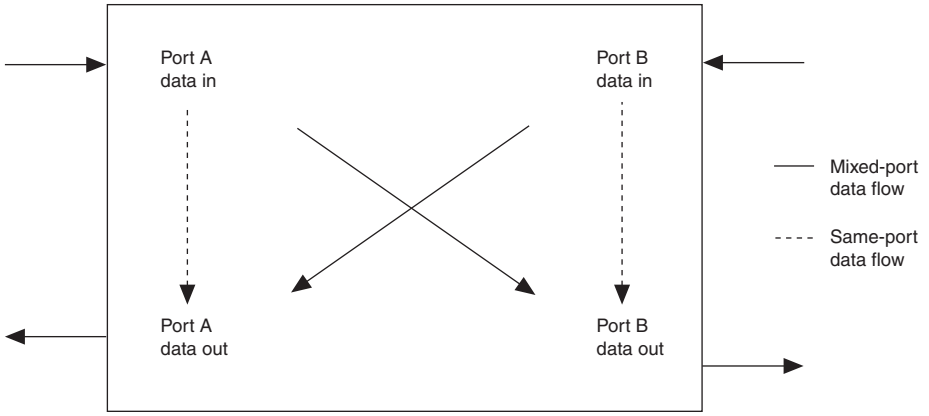
Conflict Resolution

When using the memory blocks in true dual-port mode, it is possible to attempt two write operations to the same memory location (address). Since no conflict resolution circuitry is built into the memory blocks, this results in unknown data being written to that location. Therefore, you must implement conflict resolution logic external to the memory block to avoid address conflicts.

Read During Write

You can customize the read-during-write behavior of the Stratix III TriMatrix memory blocks to suit your design needs. Two types of read-during-write operations are available: same port and mixed port. Figure 4–18 shows the difference between the two types.

Figure 4–18. Stratix III Read-During-Write Data Flow



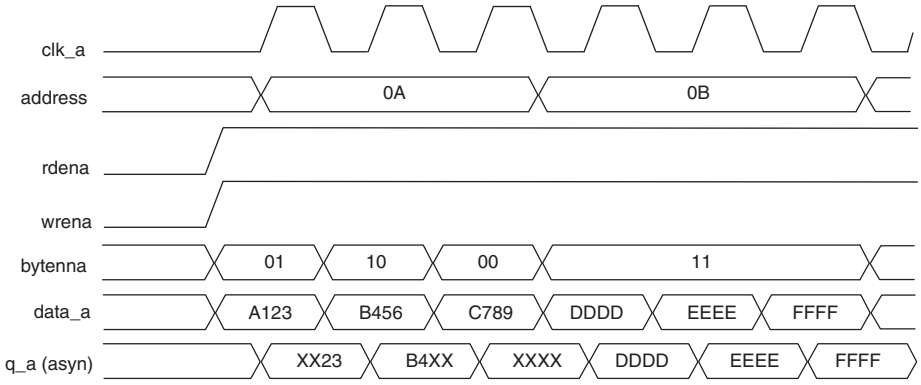
Same-Port Read-During-Write Mode

This mode applies to either a single-port RAM or the same port of a true dual-port RAM. In same-port read-during-write mode, three output choices are available: new data mode (or flow-through), old data mode, or don't care mode. In new data mode, the new data is available on the rising edge of the same clock cycle on which it was written. In old data mode, the RAM outputs reflect the old data at that address before the write operation proceeds. In don't care mode, the RAM outputs don't care values for a read-during-write operation.

If you are not using the new data mode or old data mode, you should select the don't care mode. Using the don't care mode increases the flexibility in the type of memory block used, provided you do not assign block type when instantiating a memory block. You may also get potential performance gain by selecting the don't care mode.

Figure 4–19 shows the sample functional waveforms of same-port read-during-write behavior with new data.

Figure 4–19. Same Port Read-During Write: New Data Mode *(Note 1)*

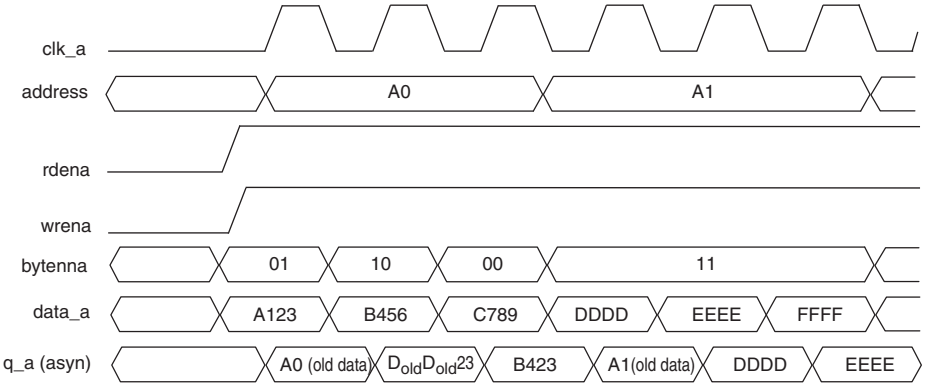


Note to Figure 4–19:

(1) "X" can be a don't care value or current data at that location, depending on the setting chosen in the Quartus II software.

Figure 4–20 shows the sample functional waveforms of same-port read-during-write behavior with old data mode.

Figure 4–20. Same Port Read-During-Write: Old Data Mode *(Note 1)*



Note to Figure 4–20:

(1) D_{old} is the old data bit at address A0, A0 (old data) is the old data at address A0, and A1 (old data) is the old data at address A1.

Mixed-Port Read-During-Write Mode

This mode applies to a RAM in simple or true dual-port mode which has one port reading and the other port writing to the same address location with the same clock.

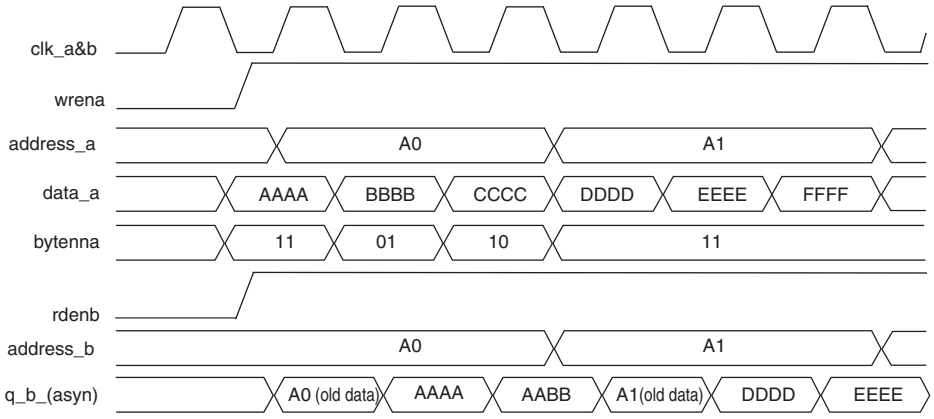
In this mode you also have two output choices: old data or don't care. In old data mode, a read-during-write operation to different ports causes the RAM outputs to reflect the old data at that address location. In don't care mode, the same operation results in a "don't care" or "unknown" value on the RAM outputs.

For more details about how to implement the desired behavior, read-during-write behavior is controlled via the RAM MegaWizard Plug-In Manager refer to the *RAM Megafunction User Guide*.

You should select don't care mode if you do not use old data mode. This increases the flexibility in the type of memory block used, if you do not assign block type when instantiating a memory block. You may also get potential performance gain by selecting don't care mode.

Figure 4–21 shows a sample functional waveform of mixed-port read-during-write behavior for the old data mode. In don't care mode, the old data shown in the figure is simply replaced with "**don't cares**".

Figure 4–21. Mixed Port Read During Write: Old Data Mode *(Note 1)*



Note to Figure 4–21:

(1) A0 (old data) is the old data at address A0 and A1 (old data) is the old data at address A1.

Mixed-port read-during-write using two different clocks in simple-dual port RAM with old data output is supported via emulation. The Quartus II software takes two memory blocks to implement the widest width mode.

Power-Up Conditions and Memory Initialization

M9K and M144K memory block outputs power up to zero (cleared), regardless of whether the output registers are used or bypassed. MLABs power up to zero if output registers are used and power up reading the memory contents if output registers are not used. However, the actual RAM cells power up to an unknown state. Therefore, after power-up, if an address is read before being written, the output from the read operation is undefined because the contents are not initialized.

All memory blocks support initialization via **.mif** file. You can create **.mif** files in the Quartus II software and specify their use with the RAM MegaWizard Plug-In Manager when instantiating a memory in your design. Even if a memory is pre-initialized (for example, by a **.mif** file), it still powers up with its outputs cleared.

For more information about **.mif** files, refer to the *RAM Megafunction User Guide* and the *Quartus II Handbook*.

Power Management

Stratix III memory block clock-enables allow you to control clocking of each memory block to reduce AC power consumption. Use the read-enable signal to ensure that read operations only occur when you need them to. If your design does not require read-during-write, you can reduce your power consumption by de-asserting the read-enable signal during write operations, or any period when no memory operations occur.

The Quartus II software automatically places any unused memory blocks in low power mode to reduce static power.

Programming File Compatibility

Beginning with version 8.1, the Quartus II software supports the logic option **STRATIXIII_MRAM_COMPATIBILITY**. When this option is set to on, the Quartus II software will generate programming files compatible with both affected and fixed silicons (for write speed decrease in M144K blocks). The default setting for this option is on.

For the list of devices that is affected by the write speed decrease for M144K blocks refer to the *Stratix III Device Family Errata Sheet*.

To set the **STRATIXIII_MRAM_COMPATIBILITY** variable, enter the following line in the Quartus Settings File:

```
set_global_assignment -name STRATIXIII_MRAM_COMPATIBILITY ON
```

When targeting fixed silicon devices, set the **STRATIXIII_MRAM_COMPATIBILITY** variable to OFF. When the **STRATIXIII_MRAM_COMPATIBILITY** option is set to OFF, you will be able to achieve the higher Fmax that is published for M144K blocks in fixed silicons and the programming files will only be compatible with fixed silicons. These programming files will not configure other silicon revisions. The `nSTATUS` pin will drive out low and configuration will fail.

Conclusion

The Stratix III TriMatrix embedded memory structure provides three different on-chip RAM block sizes to address your design needs. All memory blocks are fully customizable and can be cascaded to implement wider or deeper memories with minimal speed penalty.

You can independently configure each embedded memory block to be a single- or dual-port RAM, FIFO, ROM, or shift register via the Quartus II MegaWizard Plug-In Manager software.

Chapter Revision History

Table 4–10 shows the revision history for this chapter.

Table 4–10. Chapter Revision History

Date and Revision	Changes Made	Summary of Changes
May 2009, version 1.8	■ Updated Table 4–1. ■ Updated "Read/Write Clock Mode" and "Simple Dual-Port Mode" sections.	—
February 2009, version 1.7	■ Updated Figure 4–2, Figure 4–4, and Figure 4–5. ■ Removed "Referenced Documents" section.	—
November 2008, Version 1.6	■ Updated "Byte-Enable Support", "Address Clock Enable Support", "Asynchronous Clear", "Single Port RAM", and "Simple Dual-Port Mode" sections. ■ Updated Figure 4–1, Figure 4–5, Figure 4–8, Figure 4–10, and Figure 4–15. ■ Added Figure 4–2, Figure 4–6, Figure 4–11, Figure 4–14, and Figure 4–16.	—
October 2008, version 1.5	■ Updated Table 4–1. ■ Updated "Asynchronous Clear" and "Clocking Modes" section. ■ Added "Programming File Compatibility" section. ■ Updated New Document Format.	—
May 2008, version 1.4	■ Updated "Introduction" section. ■ Updated "TriMatrix Memory Block Types" section. ■ Updated "Byte-Enable Support" section. ■ Updated "Mixed Width Support" section. ■ Updated "Same-Port Read-During-Write Mode" section. ■ Updated Figure 4–16, Figure 4–17, and Figure 4–18. ■ Updated "Mixed-Port Read-During-Write Mode" section. ■ Updated Table 4–1, Table 4–2, and Table 4–4.	—
November 2007, version 1.3	Updated Table 4–2.	—
October 2007, version 1.2	■ Updated Table 4–1. ■ Added section "Referenced Documents". ■ Added live links for references.	—
May 2007, version 1.1	Updated Table 4–2, Table 4–9.	—
November 2006, version 1.0	Initial Release.	—