

### Introduction

Arria™ GX devices feature the TriMatrix memory structure, consisting of three sizes of embedded RAM blocks that efficiently address the memory needs of FPGA designs.

TriMatrix memory includes 512-bit M512 blocks, 4-Kbit M4K blocks, and 512-Kbit M-RAM blocks, which are each configurable to support many features. TriMatrix memory provides up to 4,477,824 bits of RAM at up to 380 MHz operation.

This chapter contains the following sections:

- [“TriMatrix Memory Overview” on page 6–1](#)
- [“Memory Modes” on page 6–9](#)
- [“Clock Modes” on page 6–19](#)
- [“Designing With TriMatrix Memory” on page 6–30](#)
- [“Read-During-Write Operation at the Same Address” on page 6–32](#)
- [“Conclusion” on page 6–34](#)

### TriMatrix Memory Overview

TriMatrix architecture provides complex memory functions for different applications in FPGA designs. For example, M512 blocks are used for first-in first-out (FIFO) functions and clock domain buffering where memory bandwidth is critical; M4K blocks are ideal for applications requiring medium-sized memory, such as asynchronous transfer mode (ATM) cell processing; and M-RAM blocks are suitable for large buffering applications, such as internet protocol (IP) packet buffering and system cache.

TriMatrix memory blocks support various memory configurations, including single-port, simple dual-port, true dual-port (also known as bidirectional dual-port), shift register, and ROM modes. TriMatrix memory architecture also includes advanced features and capabilities, such as parity-bit support, byte enable support, pack mode support, address clock enable support, mixed port width support, and mixed clock mode support.

When applied to output registers, the asynchronous clear signal clears the output registers and the effects are seen immediately. Register clears are only supported on output registers.

Table 6–1 summarizes the features supported by the three sizes of TriMatrix memory.

<b>Feature</b>	<b>M512 Blocks</b>	<b>M4K Blocks</b>	<b>M-RAM Blocks</b>
Maximum performance	345 MHz	380 MHz	290 MHz
Total RAM bits (including parity bits)	576	4,608	589,824
Configurations	512 × 1 256 × 2 128 × 4 64 × 8 64 × 9 32 × 16 32 × 18	4K × 1 2K × 2 1K × 4 512 × 8 512 × 9 256 × 16 256 × 18 128 × 32 128 × 36	64K × 8 64K × 9 32K × 16 32K × 18 16K × 32 8K × 64 8K × 72 4K × 128 4K × 144
Parity bits	✓	✓	✓
Byte enable	✓	✓	✓
Pack 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	Outputs cleared	Outputs unknown
Register clears	Output registers only	Output registers only	Output registers only
Same-port read-during-write	New data available at positive clock edge	New data available at positive clock edge	New data available at positive clock edge
Mixed-port read-during-write	Outputs set to unknown or old data	Outputs set to unknown or old data	Unknown output

Table 6–2 shows the capacity of the TriMatrix memory blocks for each device in the Arria GX family.

<b>Device</b>	<b>M512 Blocks</b>	<b>M4K Blocks</b>	<b>M-RAM Blocks</b>	<b>Total RAM Bits</b>
EP1AGX20	166	118	1	1,229,184
EP1AGX35	197	140	1	1,348,416
EP1AGX50	313	242	2	2,475,072
EP1AGX60	326	252	2	2,528,640
EP1AGX90	478	400	4	4,477,824

## Parity Bit Support

All TriMatrix memory blocks (M512, M4K, and M-RAM) support one parity bit for each byte.

Parity bits add to the amount of memory in each RAM block. For example, the M512 block has 576 bits, 64 of which are optionally used for parity bit storage. The parity bit, along with logic implemented in adaptive logic modules (ALMs), implements parity checking for error detection to ensure data integrity. Parity-size data words can also be used for other purposes such as storing user-specified control bits.

## Byte Enable Support

All TriMatrix memory blocks support byte enables that mask the input data so that only specific bytes, nibbles, or bits of data are written. The unwritten bytes or bits retain the previous written value. The write enable (*wren*) signals, along with the byte enable (*byteena*) signals, control the RAM blocks' write operations. The default value for the byte enable signals is high (enabled), in which case writing is controlled only by the write enable signals. There is no clear port to the byte enable registers.

### *M512 Blocks*

M512 blocks support byte enables for data widths of 16 and 18 bits only. The byte-enable feature for memory block configurations with widths of less than two bytes ( $\times 16/\times 18$ ) is not supported. For memory configurations less than two bytes wide, the write enable or clock enable signals can optionally be used to control the write operation.

Table 6–3 summarizes the byte selection.

<b>Table 6–3. Byte Enable for Arria GX M512 Blocks</b> <i>Note (1)</i>		
<b>byteena[1..0]</b>	<b>data ×16</b>	<b>data ×18</b>
[0] = 1	[7..0]	[8..0]
[1] = 1	[15..8]	[17..9]

**Note to Table 6–3:**

(1) Any combination of byte enables is possible.

### M4K Blocks

M4K blocks support byte enables for any combination of data widths of 16, 18, 32, and 36 bits only. The byte-enable feature for memory block configurations with widths of less than two bytes (×16/×18) is not supported. For memory configurations less than two bytes wide, the write enable or clock enable signals can optionally be used to control the write operation. Table 6–4 summarizes the byte selection.

<b>Table 6–4. Byte Enable for Arria GX M4K Blocks</b> <i>Note (1)</i>				
<b>byteena[3..0]</b>	<b>data ×16</b>	<b>data ×18</b>	<b>data ×32</b>	<b>data ×36</b>
[0] = 1	[7..0]	[8..0]	[7..0]	[8..0]
[1] = 1	[15..8]	[17..9]	[15..8]	[17..9]
[2] = 1	-	-	[23..16]	[26..18]
[3] = 1	-	-	[31..24]	[35..27]

**Note to Table 6–4:**

(1) Any combination of byte enables is possible.

### M-RAM Blocks

M-RAM blocks support byte enables for any combination of data widths of 16, 18, 32, 36, 64, and 72 bits. The byte-enable feature for memory block configurations with widths of less than two bytes (×16/×18) is not supported. In the ×128 and ×144 simple dual-port modes, the two sets of byte enable signals (byteena\_a and byteena\_b) combine to form the necessary 16 byte enables. In ×128 and ×144 modes, byte enables are only supported when using single clock mode. However, the Quartus® II software can implement byte enable in other clocking modes for ×128 or ×144 widths but will use twice as many M-RAM resources.

If clock enables are used in  $\times 128$  or  $\times 144$  mode, you must use the same clock enable setting for both A and B ports. Table 6–5 summarizes the byte selection for M-RAM blocks.

**Table 6–5. Byte Enable for Arria GX M-RAM Blocks** *Note (1)*

byteena	data $\times 16$	data $\times 18$	data $\times 32$	data $\times 36$	data $\times 64$	data $\times 72$
[0] = 1	[7..0]	[8..0]	[7..0]	[8..0]	[7..0]	[8..0]
[1] = 1	[15..8]	[17..9]	[15..8]	[17..9]	[15..8]	[17..9]
[2] = 1	-	-	[23..16]	[26..18]	[23..16]	[26..18]
[3] = 1	-	-	[31..24]	[35..27]	[31..24]	[35..27]
[4] = 1	-	-	-	-	[39..32]	[44..36]
[5] = 1	-	-	-	-	[47..40]	[53..45]
[6] = 1	-	-	-	-	[55..48]	[62..54]
[7] = 1	-	-	-	-	[63..56]	[71..63]

**Note to Table 6–5:**

(1) Any combination of byte enables is possible.

Table 6–6 summarizes the byte selection for  $\times 144$  mode.

**Table 6–6. Arria GX M-RAM Combined Byte for  $\times 144$  Mode (Part 1 of 2)** *Note (1)*

byteena	data $\times 128$	data $\times 144$
[0] = 1	[7..0]	[8..0]
[1] = 1	[15..8]	[17..9]
[2] = 1	[23..16]	[26..18]
[3] = 1	[31..24]	[35..27]
[4] = 1	[39..32]	[44..36]
[5] = 1	[47..40]	[53..45]
[6] = 1	[55..48]	[62..54]
[7] = 1	[63..56]	[71..63]
[8] = 1	[71..64]	[80..72]
[9] = 1	[79..72]	[89..73]
[10] = 1	[87..80]	[98..90]
[11] = 1	[95..88]	[107..99]
[12] = 1	[103..96]	[116..108]
[13] = 1	[111..104]	[125..117]

**Table 6-6. Arria GX M-RAM Combined Byte for ×144 Mode (Part 2 of 2)**  
*Note (1)*

byteena	data ×128	data ×144
[14] = 1	[119..112]	[134..126]
[15] = 1	[127..120]	[143..135]

**Note to Table 6-6:**

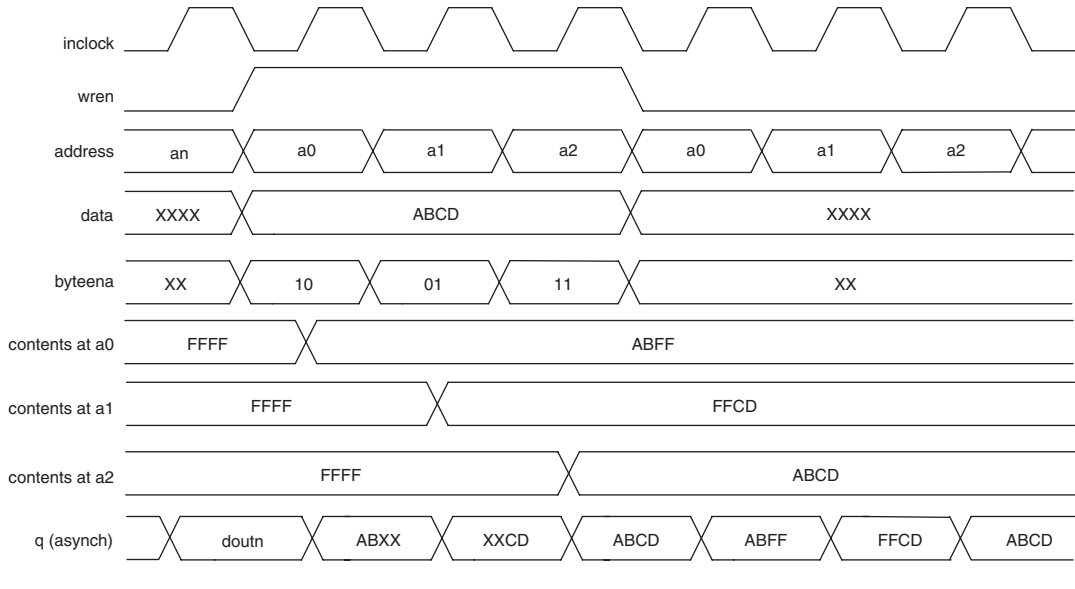
- (1) Any combination of byte enables is possible.

**Byte Enable Functional Waveform**

Figure 6-1 shows how the write enable (wren) and byte enable (byteena) signals control the operations of the RAM.

When a byte enable bit is de-asserted during a write cycle, the corresponding data byte output appears as a "don't care" or unknown value. When a byte enable bit is asserted during a write cycle, the corresponding data byte output will be the newly written data.

**Figure 6-1. Arria GX Byte Enable Functional Waveform**




## Pack Mode Support

Arria GX M4K and M-RAM memory blocks support pack mode. In M4K and M-RAM memory blocks, you can implement two single-port memory blocks in a single block under the following conditions:

- Each of the two independent block sizes is equal to or less than half of the M4K or M-RAM block size.
- Each of the single-port memory blocks is configured in single-clock mode.

Thus, each of the single-port memory blocks access up to half of the M4K or M-RAM memory resources such as clock, clock enables, and asynchronous clear signals.

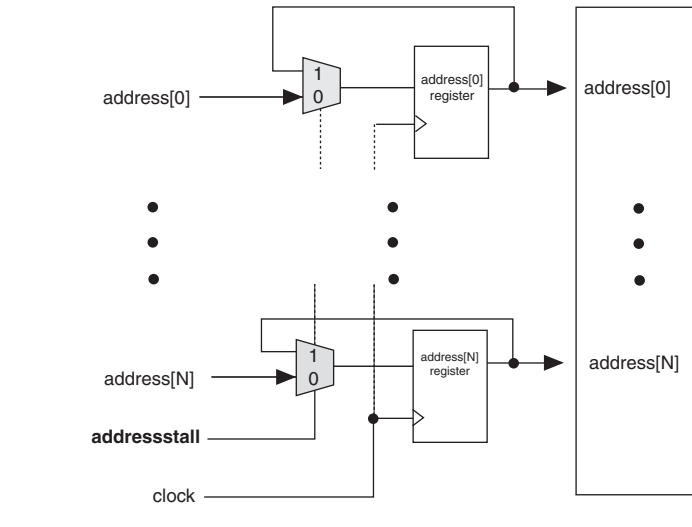
 For more information, refer to “Single-Port Mode” on page 6–10 and “Single-Clock Mode” on page 6–27.

## Address Clock Enable Support

Arria GX M4K and M-RAM memory blocks support address clock enable, which is used to hold the previous address value for as long as the signal is enabled. When the memory blocks are configured in dual-port mode, each port has its own independent address clock enable.

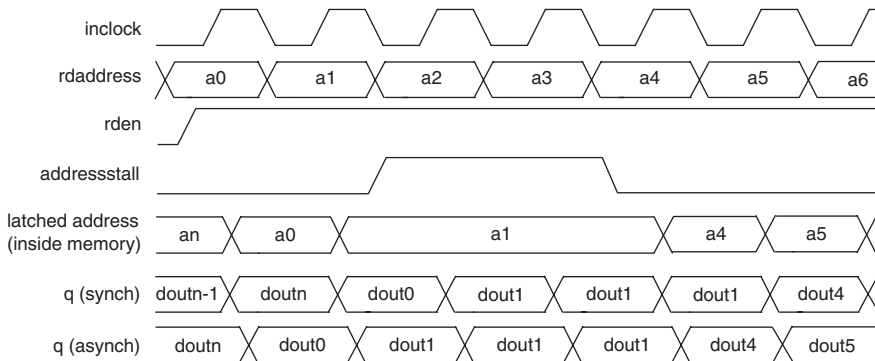
Figure 6–2 shows an address clock enable block diagram. Placed in the address register, the address signal output by the address register is fed back to the input of the register via a multiplexer. The multiplexer output is selected by the address clock enable (`addressstall`) signal. Address latching is enabled when the `addressstall` signal turns high. The output of the address register is then continuously fed into the input of the register; therefore, the address value can be held until the `addressstall` signal turns low.

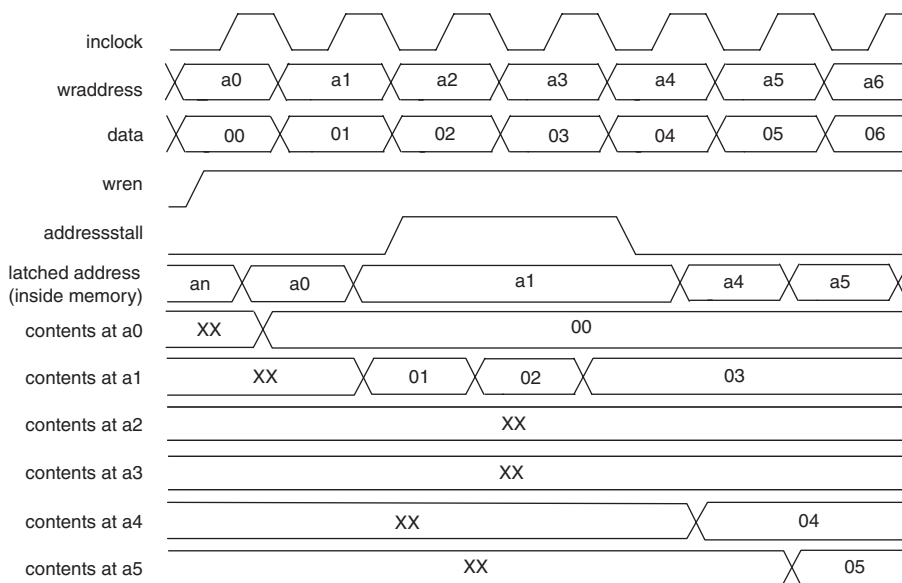
**Figure 6–2. Arria GX Address Clock Enable Block Diagram**



Address clock enable is typically used for cache memory applications, which require one port for read and another port for write. The default value for the address clock enable signals is low (disabled). Figures 6–3 and 6–4 show the address clock enable waveform during read and write cycles, respectively.

**Figure 6–3. Arria GX Address Clock Enable During Read Cycle Waveform**



**Figure 6–4. Arria GX Address Clock Enable During Write Cycle Waveform**

## Memory Modes

Arria GX TriMatrix memory blocks include input registers that synchronize writes and output registers to pipeline data to improve system performance. All TriMatrix memory blocks are fully synchronous, meaning that all inputs are registered, but outputs can be either registered or unregistered.



TriMatrix memory does not support asynchronous memory (unregistered inputs).

Depending on which TriMatrix memory block you use, the memory has various modes, including:

- Single-port
- Simple dual-port
- True dual-port (bidirectional dual-port)
- Shift-register
- ROM
- FIFO

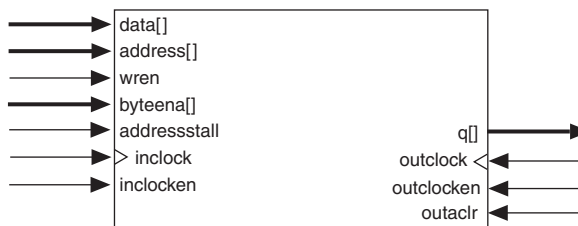


Violating the setup or hold time on the memory block address registers could corrupt memory contents. This applies to both read and write operations.

## Single-Port Mode

All TriMatrix memory blocks support single-port mode that supports non-simultaneous read and write operations. Figure 6-5 shows the single-port memory configuration for TriMatrix memory.

**Figure 6-5. Single-Port Memory Note (1)**



**Note to Figure 6-5:**

- (1) Two single-port memory blocks can be implemented in a single M4K or M-RAM block.

M4K and M-RAM memory blocks can also be halved and used for two independent single-port RAM blocks. The Altera® Quartus II software automatically uses this single-port memory packing when running low on memory resources. To force two single-port memories into one M4K or M-RAM block, first ensure that each of the two independent RAM blocks is equal to or less than half the size of the M4K or M-RAM block. Second, assign both single-port RAMs to the same M4K or M-RAM block.

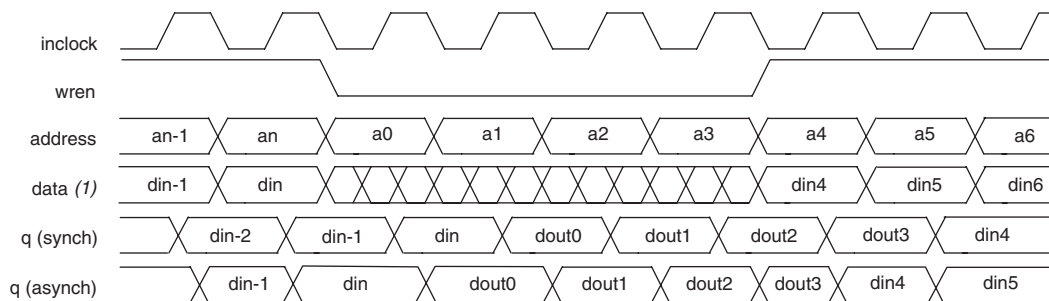
In single-port RAM configuration, the outputs can only be in read-during-write mode, which means that during the write operation, data written to the RAM flows through to the RAM outputs. When the output registers are bypassed, the new data is available on the rising edge of the same clock cycle on which it was written. For more information about read-during-write mode, refer to [“Read-During-Write Operation at the Same Address”](#) on page 6-32.

Table 6-7 shows the port width configurations for TriMatrix blocks in single-port mode.

	M512 Blocks	M4K Blocks	M-RAM Blocks
Port Width	512 × 1	4K × 1	64K × 8
Configurations	256 × 2	2K × 2	64K × 9
	128 × 4	1K × 4	32K × 16
	64 × 8	512 × 8	32K × 18
	64 × 9	512 × 9	16K × 32
	32 × 16	256 × 16	16K × 36
	32 × 18	256 × 18	8K × 64
		128 × 32	8K × 72
		128 × 36	4K × 128
			4K × 144

Figure 6-6 shows timing waveforms for read and write operations in single-port mode.

**Figure 6-6. Arria GX Single-Port Timing Waveforms** Note (1)



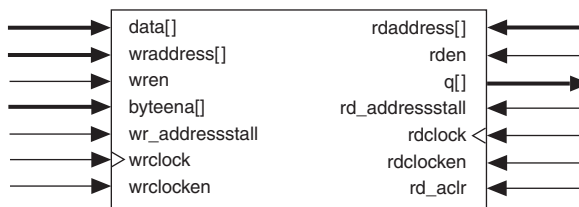
**Note to Figure 6-6:**

(1) The crosses in the data waveform during read mean "don't care."

## Simple Dual-Port Mode

All TriMatrix memory blocks support simple dual-port mode which supports a simultaneous read and write operation. Figure 6-7 shows the simple dual-port memory configuration for TriMatrix memory.

**Figure 6–7. Arria GX Simple Dual-Port Memory Note (1)**



**Note to Figure 6–7:**

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

TriMatrix memory supports mixed-width configurations, allowing different read and write port widths. Tables 6–8 through 6–10 show mixed-width configurations for M512, M4K, and M-RAM blocks, respectively.

**Table 6–8. Arria GX M512 Block Mixed-Width Configurations (Simple Dual-Port Mode)**

Read Port	Write Port						
	512 × 1	256 × 2	128 × 4	64 × 8	32 × 16	64 × 9	32 × 18
512 × 1	✓	✓	✓	✓	✓		
256 × 2	✓	✓	✓	✓	✓		
128 × 4	✓	✓	✓	✓	✓		
64 × 8	✓	✓	✓	✓	✓		
32 × 16	✓	✓	✓	✓	✓		
64 × 9						✓	✓
32 × 18						✓	✓

**Table 6–9. Arria GX M4K Block Mixed-Width Configurations (Simple Dual-Port Mode) (Part 1 of 2)**

Read Port	Write Port								
	4K × 1	2K × 2	1K × 4	512 × 8	256 × 16	128 × 32	512 × 9	256 × 18	128 × 36
4K × 1	✓	✓	✓	✓	✓	✓			
2K × 2	✓	✓	✓	✓	✓	✓			

**Table 6–9. Arria GX M4K Block Mixed-Width Configurations (Simple Dual-Port Mode) (Part 2 of 2)**

Read Port	Write Port								
	4K × 1	2K × 2	1K × 4	512 × 8	256 × 16	128 × 32	512 × 9	256 × 18	128 × 36
1K × 4	✓	✓	✓	✓	✓	✓			
512 × 8	✓	✓	✓	✓	✓	✓			
256 × 16	✓	✓	✓	✓	✓	✓			
128 × 32	✓	✓	✓	✓	✓	✓			
512 × 9							✓	✓	✓
256 × 18							✓	✓	✓
128 × 36							✓	✓	✓

**Table 6–10. Arria GX M-RAM Block Mixed-Width Configurations (Simple Dual-Port Mode)**

Read Port	Write Port				
	64K × 9	32K × 18	18K × 36	8K × 72	4K × 144
64K × 9	✓	✓	✓	✓	
32K × 18	✓	✓	✓	✓	
18K × 36	✓	✓	✓	✓	
8K × 72	✓	✓	✓	✓	
4K × 144					✓

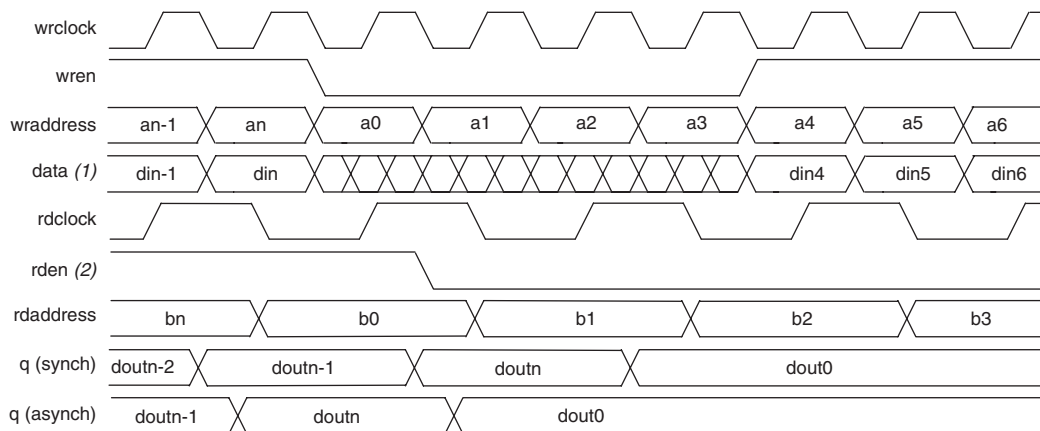
In simple dual-port mode, M512 and M4K blocks have one write enable and one read enable signal. However, M-RAM blocks contain only a write enable signal, which is held high to perform a write operation. M-RAM blocks are always enabled for read operations. The Quartus II software can emulate a read-enable signal for M-RAM blocks by using the clock-enable signal if it is not already used. If the read address and write address select the same address location during a write operation, M-RAM block output is unknown.

TriMatrix memory blocks do not support a clear port on the write enable and read enable registers. When the read enable is deactivated, the current data is retained at the output ports. If the read enable is activated during a write operation with the same address location selected, the

simple dual-port RAM output is either unknown or can be set to output the old data stored at the memory address. For more information, refer to “Read-During-Write Operation at the Same Address” on page 6–32.

Figure 6–8 shows timing waveforms for read and write operations in simple dual-port mode.

**Figure 6–8. Arria GX Simple Dual-Port Timing Waveforms** Notes (1), (2)

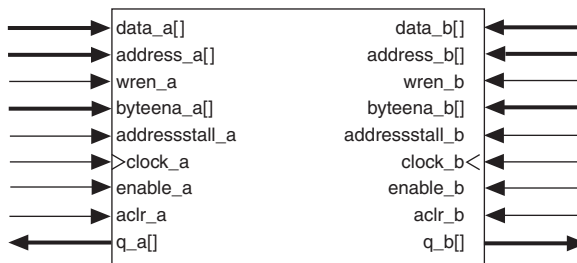


**Notes to Figure 6–8:**

- (1) The crosses in the data waveform during read mean "don't care."
- (2) The read enable rden signal is not available in M-RAM blocks. The M-RAM block in simple dual-port mode always reads out the data stored at the current read address location.

### True Dual-Port Mode

Arria GX M4K and M-RAM memory blocks support true dual-port mode. True dual-port mode supports any combination of two-port operations: two reads, two writes, or one read and one write at two different clock frequencies. Figure 6–9 shows Arria GX true dual-port memory configuration.

**Figure 6–9. Arria GX True Dual-Port Memory Note (1)**


**Note to Figure 6–9:**

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

The widest bit configuration of the M4K and M-RAM blocks in true dual-port mode is as follows:

- 256 × 16-bit (×18-bit with parity) (M4K)
- 8K × 64-bit (×72-bit with parity) (M-RAM)

The 128 × 32-bit (×36-bit with parity) configuration of the M4K block and the 4K × 128-bit (×144-bit with parity) configuration of the M-RAM block 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, the maximum width of the true dual-port RAM equals half of the total number of output drivers.

Table 6–11 lists the possible M4K block mixed-port width configurations.

**Table 6–11. Arria GX M4K Block Mixed-Port Width Configurations (True Dual-Port)**

Read Port	Write Port						
	4K × 1	2K × 2	1K × 4	512 × 8	256 × 16	512 × 9	256 × 18
4K × 1	✓	✓	✓	✓	✓		
2K × 2	✓	✓	✓	✓	✓		
1K × 4	✓	✓	✓	✓	✓		
512 × 8	✓	✓	✓	✓	✓		
256 × 16	✓	✓	✓	✓	✓		
512 × 9						✓	✓
256 × 18						✓	✓

Table 6–12 lists the possible M-RAM block mixed-port width configurations.

Read Port	Write Port			
	64K × 9	32K × 18	18K × 36	8K × 72
64K × 9	✓	✓	✓	✓
32K × 18	✓	✓	✓	✓
18K × 36	✓	✓	✓	✓
8K × 72	✓	✓	✓	✓

In true dual-port configuration, the RAM outputs can only be configured for read-during-write mode. This means that during write operation, data being written to the A or B port of the RAM flows through to the A or B outputs, respectively. When the output registers are bypassed, the new data is available on the rising edge of the same clock cycle on which it was written. For waveforms and information on mixed-port read-during-write mode, refer to “[Read-During-Write Operation at the Same Address](#)” on page 6–32.

Potential write contentions must be resolved external to the RAM because writing to the same address location at both ports results in unknown data storage at that location. For a valid write operation to the same address of the M-RAM block, the rising edge of the write clock for port A must occur following the maximum write cycle time interval after the rising edge of the write clock for port B. Data is written on the rising edge of the write clock for the M-RAM block.

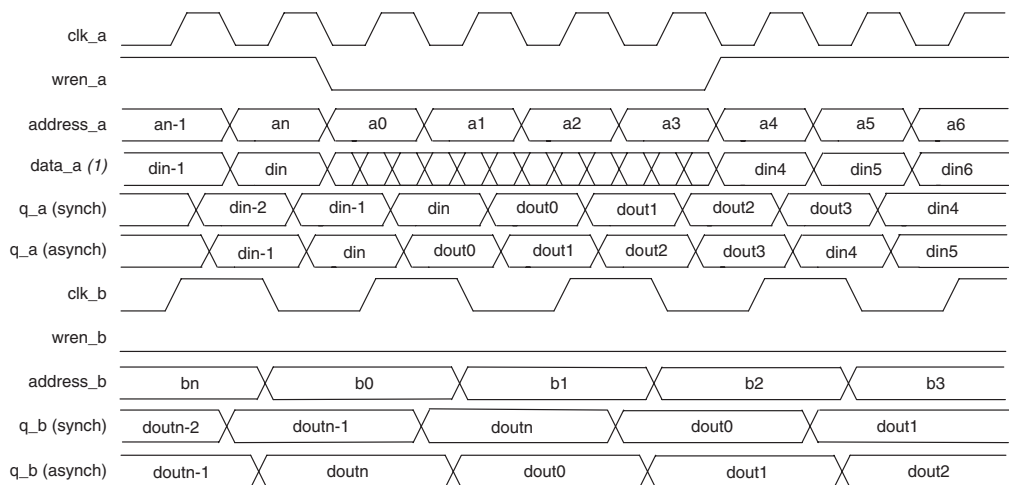
Because data is written into the M512 and M4K blocks at the falling edge of the write clock, the rising edge of the write clock for port A should occur following half of the maximum write cycle time interval after the falling edge of the write clock for port B. If this timing is not met, the data stored in that particular address will be invalid.



For the maximum synchronous write cycle time, refer to the [Arria GX Device Family Data Sheet](#) in volume 1 of the *Arria GX Device Handbook*.

Figure 6–10 shows true dual-port timing waveforms for the write operation at port A and the read operation at port B.

**Figure 6–10. Arria GX True Dual-Port Timing Waveforms Note (1)**



**Note to Figure 6–10:**

(1) The crosses in the data\_a waveform during write mean "don't care."

## Shift-Register Mode

All Arria GX memory blocks support shift-register mode.

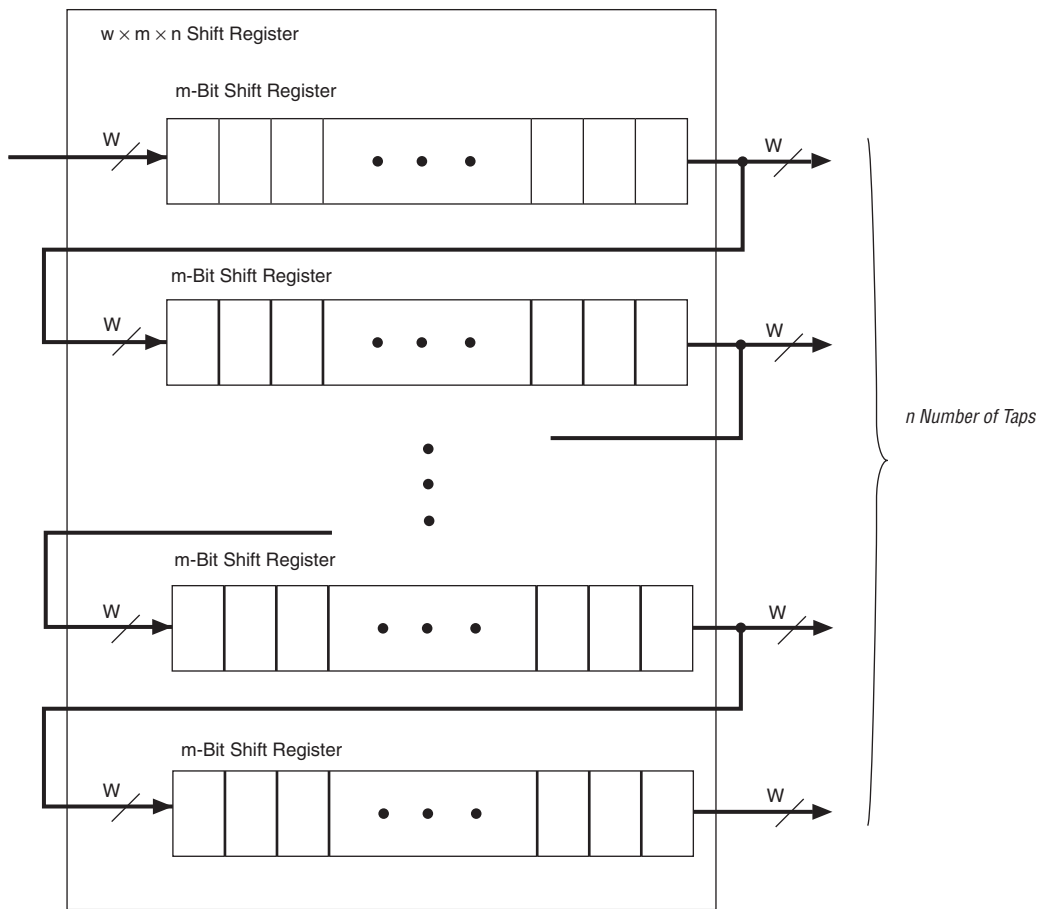
Embedded memory block configurations can implement shift registers for digital signal processing (DSP) applications, such as finite impulse response (FIR) filters, pseudo-random number generators, multi-channel filtering, and auto-correlation and cross-correlation functions. These and other DSP applications require local data storage, traditionally implemented with standard flip flops 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 ( $w \times m \times n$ ) shift register is determined by the input data width ( $w$ ), the length of the taps ( $m$ ), and the number of taps ( $n$ ), and must be less than or equal to the maximum number of memory bits in the respective block: 576 bits for the M512 block, 4,608 bits for the M4K block, and 589,824 bits for the MRAM block. In addition, the size of  $w \times n$  must be less than or equal to the maximum width of the respective block: 18 bits

for the M512 block, 36 bits for the M4K block, and 144 bits for the MRAM block. If a larger shift register is required, the memory blocks can be cascaded.

In M512 and M4K blocks, data is written into each address location at the falling edge of the clock and read from the address at the rising edge of the clock. Shift-register mode logic automatically controls the positive and negative edge clocking to shift the data in one clock cycle. The MRAM block performs reads and writes on the rising edge. Figure 6–11 shows the TriMatrix memory block in the shift-register mode.

Figure 6–11. Arria GX Shift-Register Memory Configuration



## ROM Mode

M512 and M4K memory blocks support ROM mode. A memory initialization file (.mif) initializes the ROM contents of these blocks. The address lines of the ROM are registered. The outputs can be registered or unregistered. The ROM read operation is identical to the read operation in the single-port RAM configuration.

## FIFO Buffers Mode

TriMatrix memory blocks support FIFO mode. M512 memory blocks are ideal for designs with many shallow FIFO buffers. All memory configurations have synchronous inputs; however, the FIFO buffer outputs are always combinational. Simultaneous read and write from an empty FIFO buffer is not supported.



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

## Clock Modes

Depending on which TriMatrix memory mode you select, the following clock modes are available:

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

Table 6–13 shows these clock modes supported by all TriMatrix blocks when configured as respective memory modes.

<b>Clocking Modes</b>	<b>True Dual-Port Mode</b>	<b>Simple Dual-Port Mode</b>	<b>Single-Port Mode</b>
Independent	✓		
Input/output	✓	✓	✓
Read/write		✓	
Single clock	✓	✓	✓

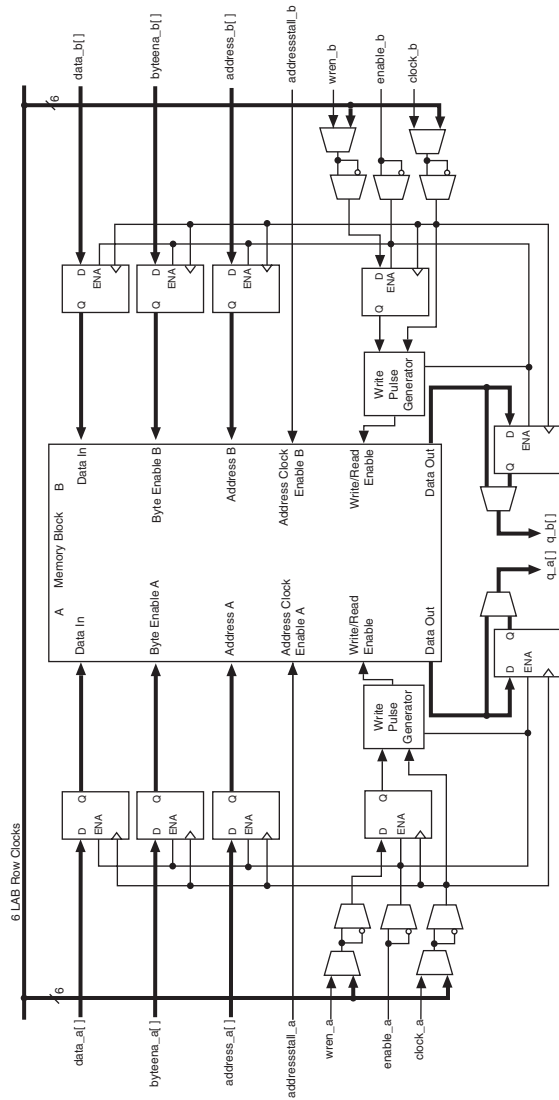
### Independent Clock Mode

TriMatrix memory blocks can implement independent clock mode for true dual-port memory. 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 B registers. Asynchronous clear signals for the registers; however, are not supported.

Figure 6–12 shows a TriMatrix memory block in independent clock mode.

**Figure 6–12. Arria GX TriMatrix Memory Block in Independent Clock Mode**

*Note (1)*



**Note to Figure 6–12:**

- (1) 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.

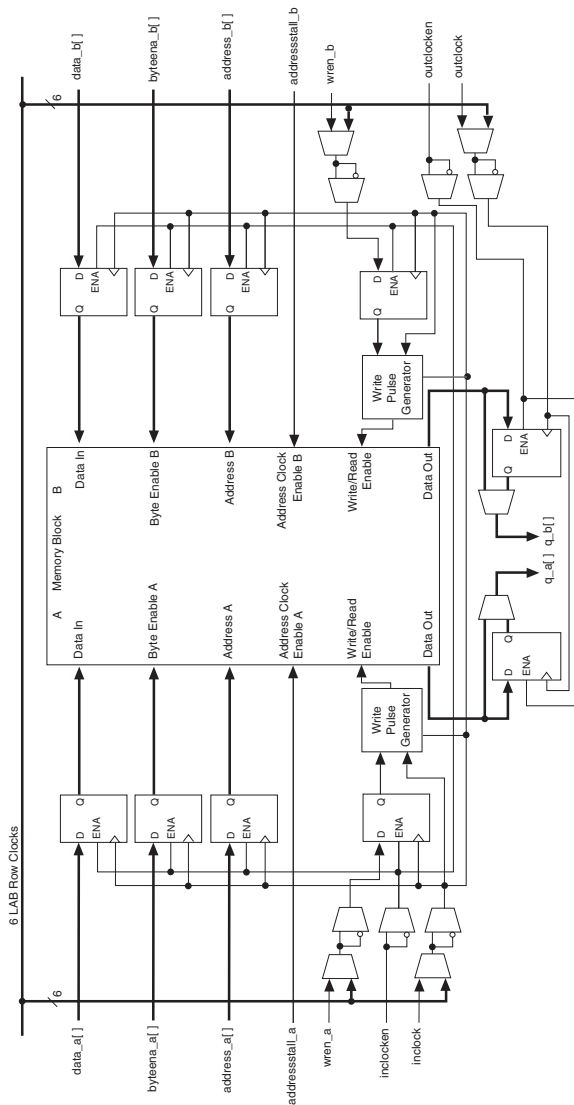
### Input and Output Clock Mode

Arria GX TriMatrix memory blocks can implement input and output clock mode for true and simple dual-port memory. On each of the two ports, A and B, one clock controls all registers for the following inputs into the memory block: data input, write enable, and address. The other clock controls the blocks' data output registers. Each memory block port also supports independent clock enables for input and output registers. Asynchronous clear signals for the registers; however, are not supported.

Figures 6–13 through 6–15 show the memory block in input and output clock mode for true dual-port, simple dual-port, and single-port modes, respectively.

**Figure 6–13. Arria GX Input/Output Clock Mode in True Dual-Port Mode**

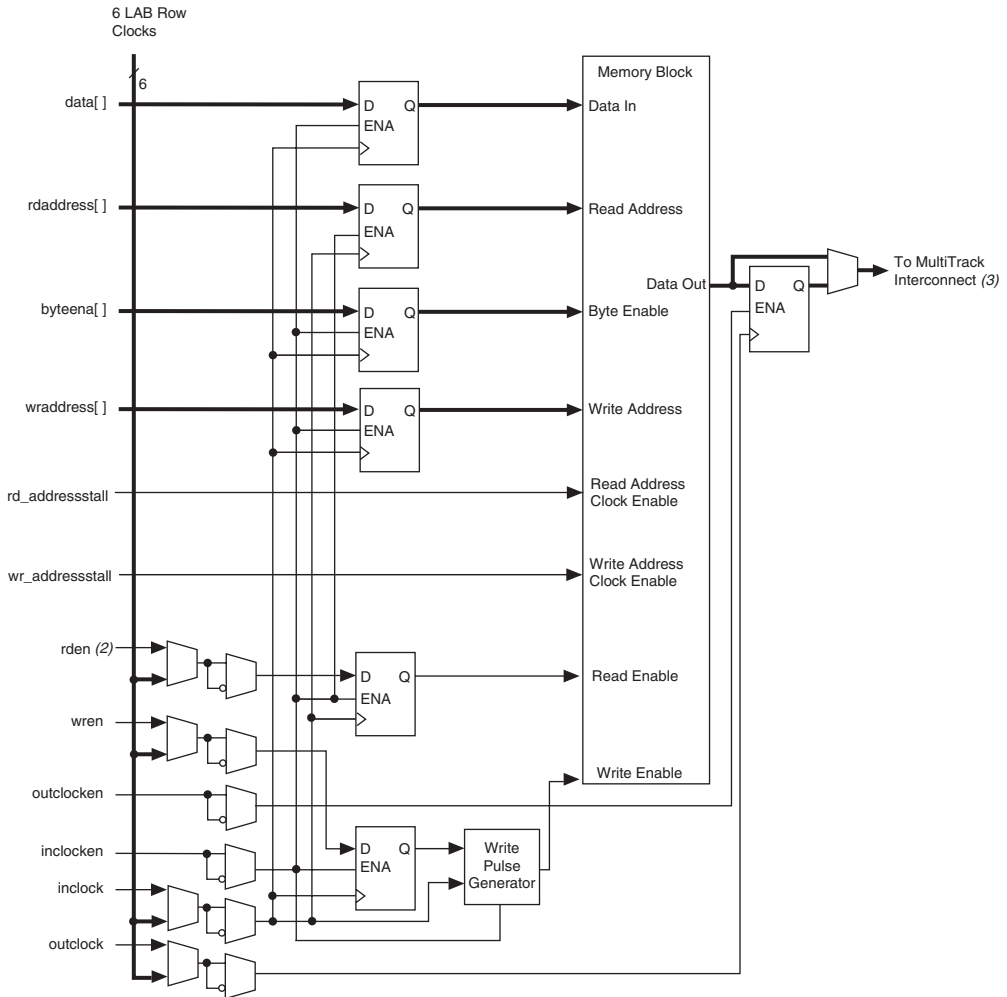
*Note (1)*



**Note to Figure 6–13:**

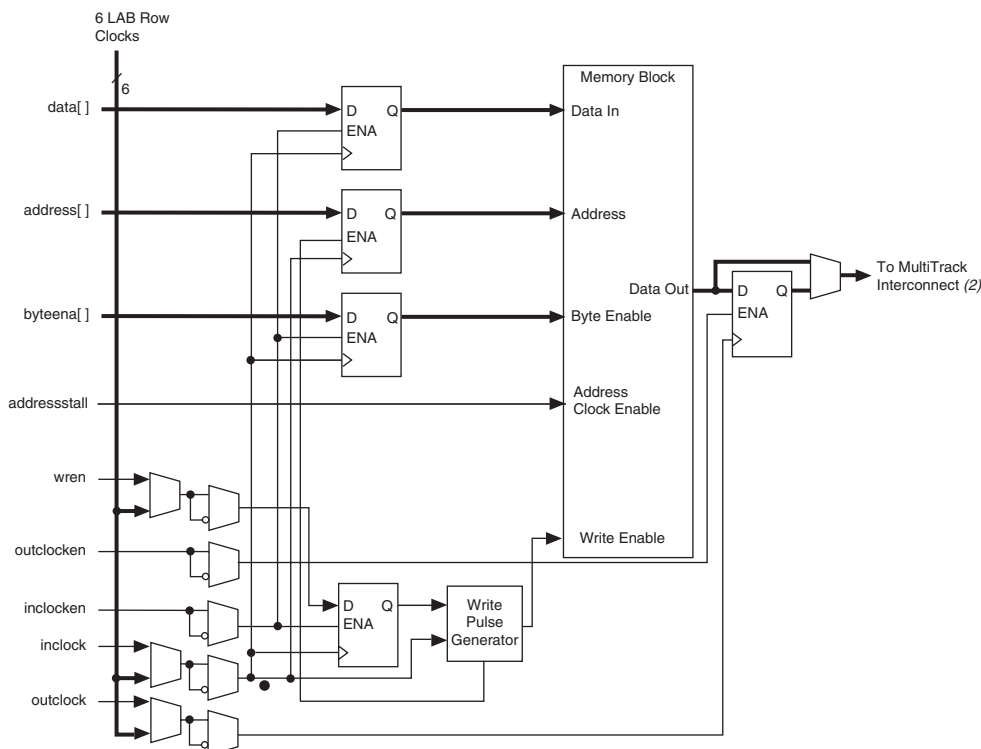
- (1) Violating the setup or hold time on the memory block address registers could corrupt the memory contents. This applies to both read and write operations.

**Figure 6–14. Arria GX Input/Output Clock Mode in Simple Dual-Port Mode** Notes (1), (2), (3)



**Notes to Figure 6–14:**

- (1) 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.
- (2) The read enable *rden* signal is not available in the M-RAM block. An M-RAM block in simple dual-port mode is always reading out the data stored at the current read address location.
- (3) For more information about the MultiTrack interconnect, refer to the *Arria GX Device Family Data Sheet* in volume 1 of the *Arria GX Device Handbook*.

**Figure 6–15. Arria GX Input/Output Clock Mode in Single-Port Mode** *Notes (1), (2)***Notes to Figure 6–15:**

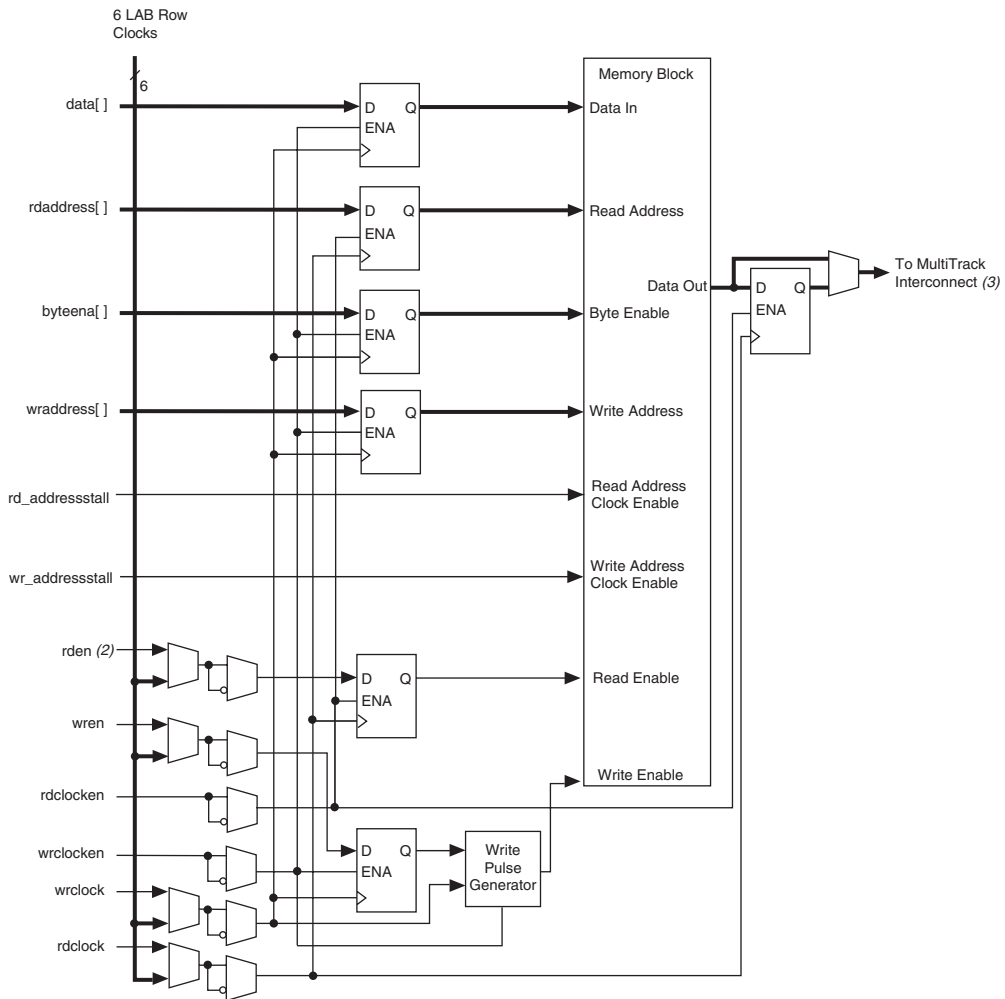
- (1) Violating the setup or hold time on the memory block address registers could corrupt the memory contents. This applies to both read and write operations.
- (2) For more information about the MultiTrack interconnect, refer to the *Arria GX Device Family Data Sheet* in volume 1 of the *Arria GX Device Handbook*.

## Read and Write Clock Mode

Arria GX TriMatrix memory blocks can implement read and write clock mode for simple dual-port memory. This mode uses up to two clocks. The write clock controls the blocks' data inputs, write address, and write enable signals. The read clock controls the data output, read address, and read-enable signals. The memory blocks support independent clock enables for each clock for the read- and write-side registers. However, asynchronous clear signals for the registers are not supported.

Figure 6–16 shows a memory block in read and write clock mode.

**Figure 6–16. ArriaGX Read and Write Clock Mode** *Notes (1), (2), (3)*



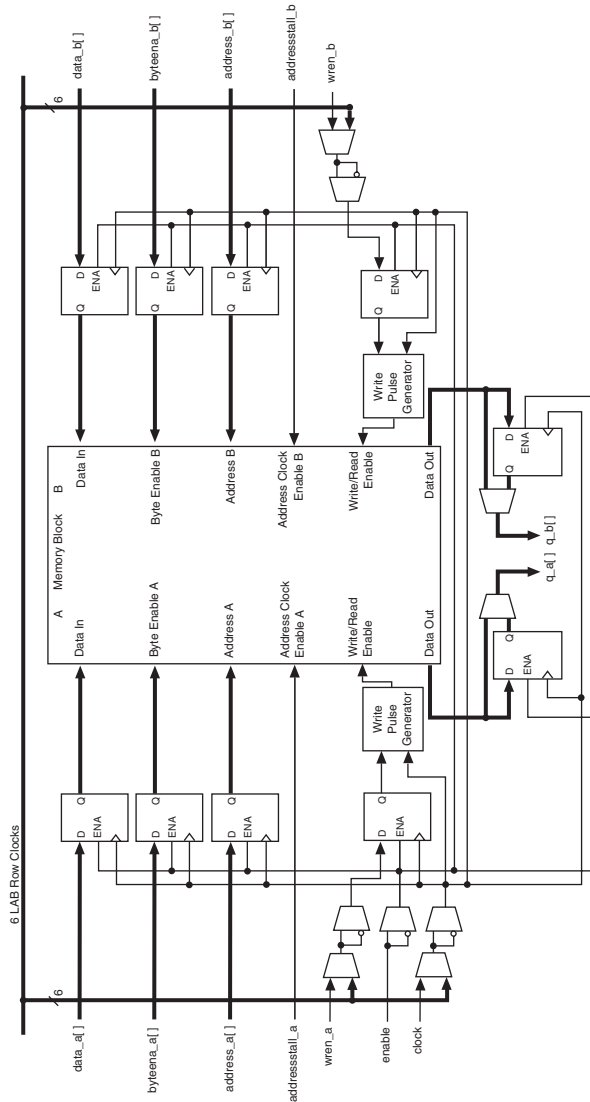
**Notes to Figure 6–16:**

- (1) 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.
- (2) The read enable *rden* signal is not available in the M-RAM block. An M-RAM block in simple dual-port mode is always reading the data stored at the current read address location.
- (3) For more information about the MultiTrack interconnect, refer to the *Arria GX Device Family Data Sheet* in volume 1 of the *Arria GX Device Handbook*.

## Single-Clock Mode

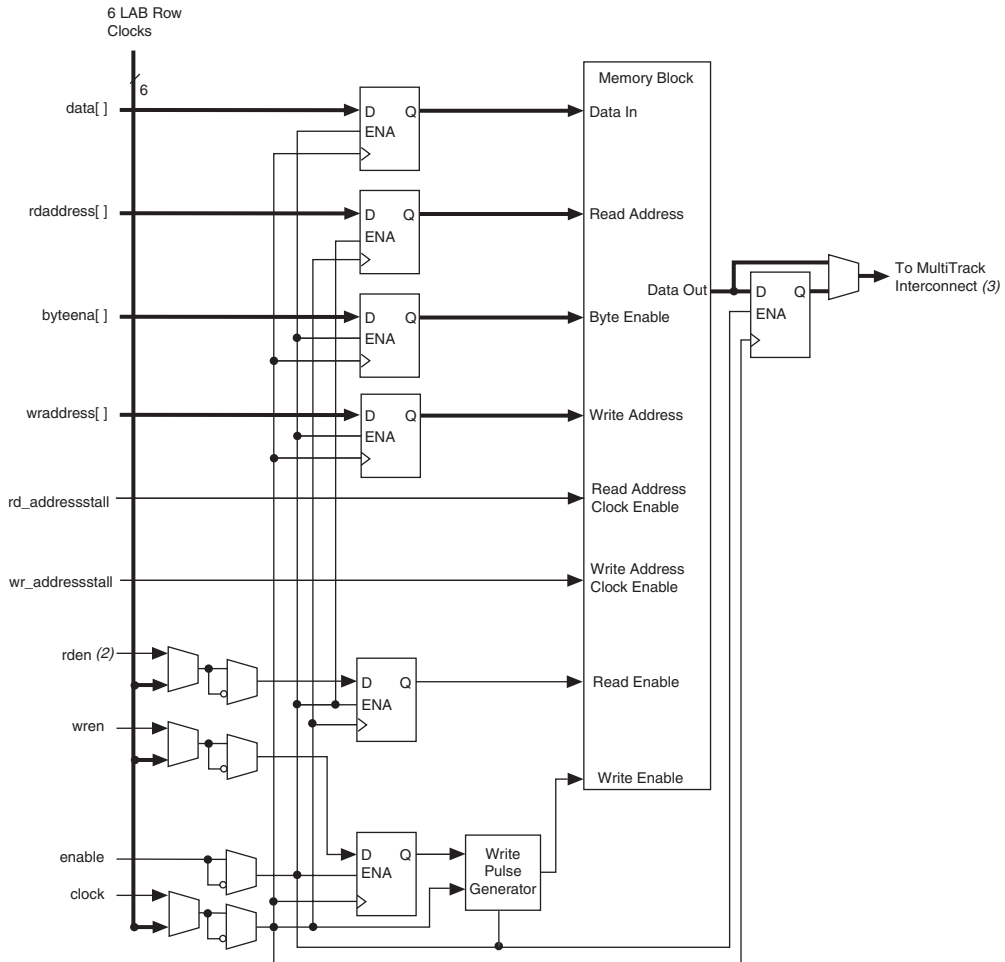
Arria GX TriMatrix memory blocks implement single-clock mode for true dual-port, simple dual-port, and single-port memory. In this mode, a single clock, together with clock enable, is used to control all registers of the memory block. However, asynchronous clear signals for the registers are not supported. [Figures 6–17](#) through [6–19](#) show the memory block in single-clock mode for true dual-port, simple dual-port, and single-port modes, respectively.

Figure 6-17. ArriaGX Single-Clock Mode in True Dual-Port Mode *Note (1)*



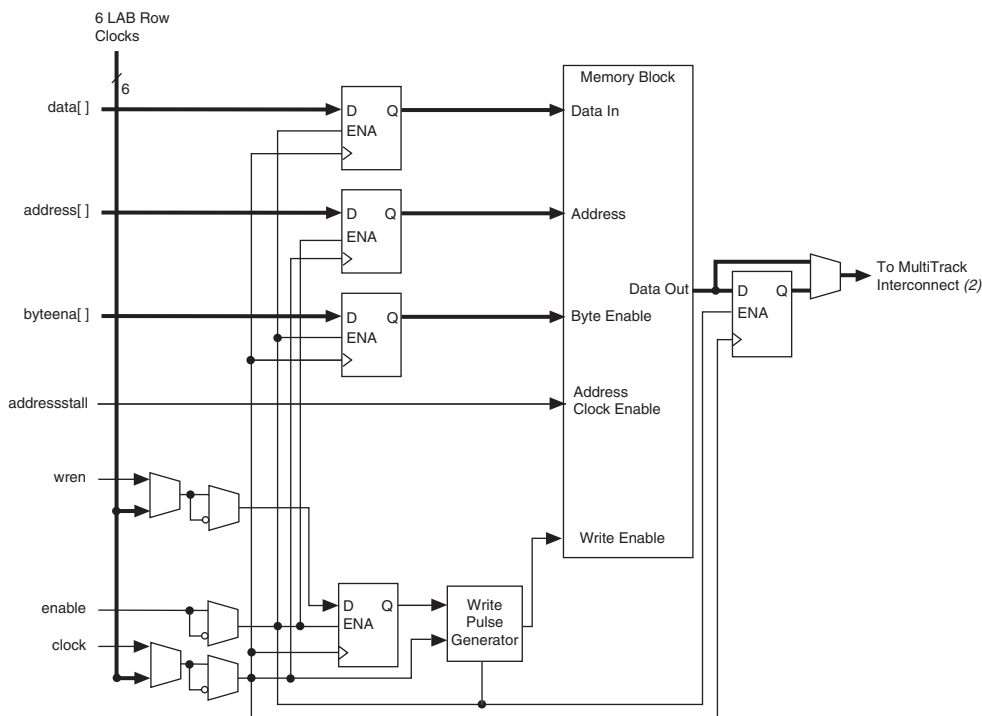
**Note to Figure 6-17:**

- (1) 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.

**Figure 6–18. ArriaGX Single-Clock Mode in Simple Dual-Port Mode** *Note (1), (2), (3)***Notes to Figure 6–18:**

- (1) 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.
- (2) The read enable `rden` signal is not available in the M-RAM block. A M-RAM block in simple dual-port mode is always reading the data stored at the current read address location.
- (3) For more information about the MultiTrack interconnect, refer to the *Arria GX Device Family Data Sheet* in volume 1 of the *Arria GX Device Handbook*.

Figure 6–19. Figure 7–19. Arria GX Single-Clock Mode in Single-Port Mode Note (1), (2)



**Notes to Figure 6–19:**

- (1) 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.
- (2) For more information about the MultiTrack interconnect, refer to the *Arria GX Device Family Data Sheet* in volume 1 of the *Arria GX Device Handbook*.

## Designing With TriMatrix Memory

When instantiating TriMatrix memory, it is important to understand the features that set it apart from other memory architectures. The following sections describe the unique attributes and functionality of TriMatrix memory.

### Selecting TriMatrix Memory Blocks

The Quartus II software automatically partitions user-defined memory into embedded memory blocks using the most efficient size combinations. The memory can also be manually assigned to a specific block size or a mixture of block sizes. Table 6–1 on page 6–2 is a guide for selecting a TriMatrix memory block size based on supported features.



For more information about selecting the appropriate memory block, refer to *AN 207: TriMatrix Memory Selection Using the Quartus II Software*.

## Synchronous and Pseudo-Asynchronous Modes

TriMatrix memory architecture implements synchronous RAM by registering the input and output signals to the RAM block. The inputs to all TriMatrix memory blocks are registered providing synchronous write cycles, while the output registers can be bypassed. In a synchronous operation, RAM generates its own self-timed strobe write-enable signal derived from the global or regional clock. In contrast, a circuit using an asynchronous RAM must generate the RAM write enable signal while ensuring that its data and address signals meet setup and hold time specifications relative to the write enable signal. During a synchronous operation, the RAM is used in pipelined mode (inputs and outputs registered) or flow-through mode (only inputs registered). However, in an asynchronous memory, neither the input nor the output is registered.

While Arria GX devices do not support asynchronous memory, they do support a pseudo-asynchronous read where the output data is available during the clock cycle when the read address is driven into it. Pseudo-asynchronous reading is possible in the simple and true dual-port modes of the M512 and M4K blocks by clocking the read-enable and read-address registers on the negative clock edge and bypassing the output registers.



For more information, refer to *AN 210: Converting Memory from Asynchronous to Synchronous for Stratix & Stratix GX Designs*.

## Power-Up Conditions & Memory Initialization

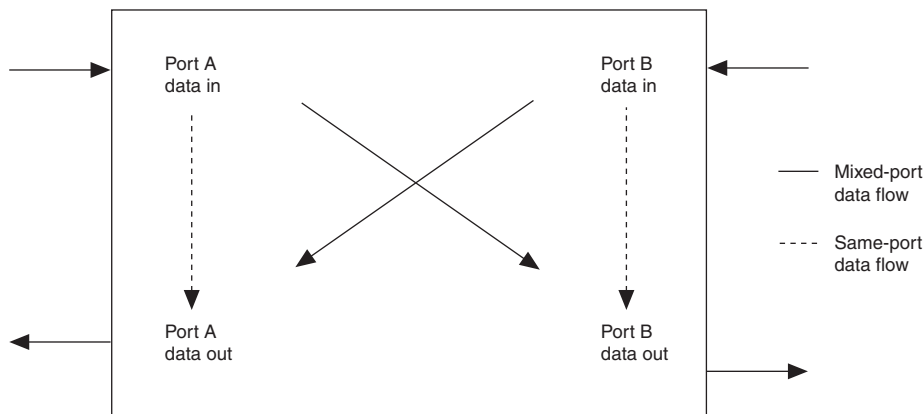
Upon power up, TriMatrix memory is in an idle state. The M512 and M4K block outputs always power up to zero, regardless of whether the output registers are used or bypassed. Even if a MIF (.mif) is used to pre-load the contents of the RAM block, the outputs will still power up as cleared. For example, if address 0 is pre-initialized to FF, the M512 and M4K blocks power up with the output at 00.

M-RAM blocks do not support .mif files; therefore, they cannot be pre-loaded with data upon power up. M-RAM blocks asynchronous outputs and memory controls always power up to an unknown state. If M-RAM block outputs are registered, the registers power up as cleared. When a read is performed immediately after power up, the output from the read operation is undefined since the M-RAM contents are not initialized. The read operation continues to be undefined for a given address until a write operation is performed for that address.

## Read-During-Write Operation at the Same Address

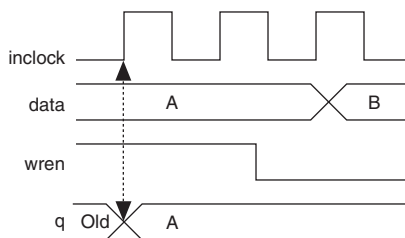
The “[Same-Port Read-During-Write Mode](#)” and “[Mixed-Port Read-During-Write Mode](#)” sections describe the functionality of the various RAM configurations when reading from an address during a write operation at that same address. There are two read-during-write data flows: same-port and mixed-port. [Figure 6–20](#) shows the difference between these flows.

**Figure 6–20. ArriaGX Read-During-Write Data Flow**



### Same-Port Read-During-Write Mode

For a read-during-write operation of a single-port RAM or the same port of a true dual-port RAM, the new data is available on the rising edge of the same clock cycle on which it was written. This behavior is valid on all memory block sizes. [Figure 6–21](#) shows a sample functional waveform. When using byte enables in true dual-port RAM mode, the outputs for the masked bytes on the same port are unknown (refer to [Figure 6–1 on page 6–6](#)). The non-masked bytes are read out, as shown in [Figure 6–21](#).

**Figure 6–21. Arria GX Same-Port Read-During-Write Functionality Note (1)**

**Note to Figure 6–21:**

(1) Outputs are not registered.

## Mixed-Port Read-During-Write Mode

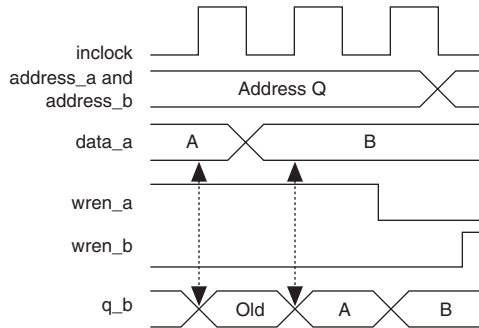
This mode is used when a RAM in simple or true dual-port mode has one port reading and the other port writing to the same address location with the same clock.

The `READ_DURING_WRITE_MODE_MIXED_PORTS` parameter for M512 and M4K memory blocks determines whether to output the old data at the address or a "don't care" value. Setting this parameter to `OLD_DATA` outputs the old data at that address. Setting this parameter to `DONT_CARE` outputs a "don't care" or unknown value. Figures 6–22 and 6–23 show sample functional waveforms where both ports have the same address. These figures assume that the outputs are not registered.

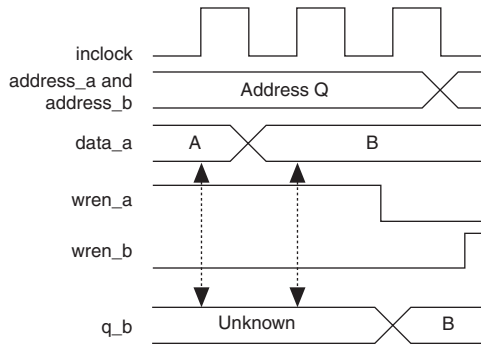
The `DONT_CARE` setting allows memory implementation in any TriMatrix memory block, whereas the `OLD_DATA` setting restricts memory implementation to only M512 or M4K memory blocks. Selecting `DONT_CARE` gives the compiler more flexibility when placing memory functions into TriMatrix memory.

The RAM outputs are unknown for a mixed-port read-during-write operation of the same address location of an M-RAM block, as shown in Figure 6–23.

**Figure 6–22. Arria GX Mixed-Port Read-During-Write: OLD\_DATA**



**Figure 6–23. Arria GX Mixed-Port Read-During-Write: DONT\_CARE**



Mixed-port read-during-write is not supported when two different clocks are used in a dual-port RAM. The output value is unknown during a mixed-port read-during-write operation.

## Conclusion

The TriMatrix memory structure of Arria GX devices provides an enhanced RAM architecture with high memory bandwidth. It addresses the needs of different memory applications in FPGA designs with features such as different memory block sizes and modes, byte enables, parity bit storage, address clock enables, mixed clock mode, shift register mode, mixed-port width support, and true dual-port mode.

## Referenced Documents

This chapter references the following documents:

- *AN 207: TriMatrix Memory Selection Using the Quartus II Software*
- *AN 210: Converting Memory from Asynchronous to Synchronous for Stratix & Stratix GX Designs*
- *Arria GX Device Family Data Sheet* in volume 1 of the *Arria GX Device Handbook*
- *FIFO Partitioner Megafunction User Guide*
- *Single- and Dual-Clock FIFO Megafunctions User Guide*

## Document Revision History

Table 6–14 shows the revision history for this chapter.

<b>Date and Document Version</b>	<b>Changes Made</b>	<b>Summary of Changes</b>
May 2008, v1.2	Updated the “Introduction” section.	—
	Minor text edits.	—
August 2007, v1.1	Added the “Referenced Documents” section.	—
May 2007, v1.0	Initial release.	—

