BRAM Wrapper Core User Guide

UG-CORE-BRAM-WRAPPER-v1.4
September 2023
www.elitestek.com

Contents

Introduction **3**

Features **3**

Device Support **3**

Resource Utilization and Performance **4**

Release Notes **7**

Functional Description **8**

- Ports 10
- Mixed Width Port Configuration 12
- Memory Modes 13
- Write Modes 14
  - Write First Mode 15
  - Read First Mode 16
  - No Change Mode 17
  - Read Unknown Mode 17
  - Conflict Resolution 18
- Clock Mode 19
- Address Update 20
- Byte Enable 21
- Read Enable 21
- Optional Output Register 21
- Optional Reset Pin 21
- Memory Initialization Files 22

IP Manager **24**

Customizing the BRAM Wrapper **25**

BRAM Wrapper Example Design **28**

BRAM Wrapper Testbench **29**

Revision History **29**

Introduction

The BRAM Wrapper core utilizes the embedded block memory within the FPGA to optimize the utilization of memory primitives. It can be configured as Single-port RAM, Simple Dual-port RAM, True Dual-port RAM, Single-port ROM, or True Dual-port ROM. The BRAM Wrapper consists of Port A and Port B, which serve as the write and read interfaces within a shared memory space, offering a wide range of configurations.

Use the IP Manager to select IP, customize it, and generate files. The BRAM Wrapper core has an interactive wizard to help you set parameters. The wizard also has options to create a testbench and/or example design targeting an 易灵思® development board.

Features

- Supports:
  - Single-port RAM (SP RAM)
  - Simple Dual-port RAM (SDP RAM)
  - True Dual-port RAM (TDP RAM)
  - Single-port ROM (SP ROM)
  - Dual-port ROM (DP ROM)
- Supports wide range of memory address and data width (limited only by the FPGA's block RAM)
- Supports:
  - Symmetric port ratio for all RAM and ROM
  - Asymmetric aspect ratio for dual-port configuration (write-to-read port ratios ranging from 1:16 to 16:1).
- Selectable RAM Mode: WRITE_FIRST, READ_FIRST and READ_UNKNOWN or NO_CHANGE
- Configurable memory initialization
- Single clock and dual clock mode
- Additional option to pipeline data out for better performance
- Includes example design targeting 钛金系列 Ti60F225 and Trion T120BGA576 Development Boards
- Testbench demonstrating memory operation simulation

Device Support

Table 1: BRAM Wrapper Core Device Support

FPGA Family	Supported Device
Trion	All
钛金系列	All

Resource Utilization and Performance

Note: The resources and performance values provided are based on some of the supported FPGAs. These values are just guidance and change depending on the device resource utilization, design congestion, and user design.

Each memory type are tested with the following settings:

- *Single-port RAM* and *Single-port ROM*—512-bit data width and 512 depth
- *Simple Dual-port RAM*—256-bit data width and 256 depth
- *True Dual-port RAM* and *Dual-port ROM*—128-bit data width and 128 depth

Note: Slight fluctuation in the performance frequency may occur after you changed the pipeline configuration and re-synthesize the design. This is due to the Efinity random Placement and Routing behavior when you re-synthesize the design.

钛金系列 Resource Utilization andPerformance

Table 2: Synchronous Clock BRAM Wrapper

FPGA	Memory Type	Width Ratio	Output Pipeline Register	Logic Elements [1]	Memory Blocks	DSP Blocks	f_{MAX} (MHz) *clk*	Efinity Version
Ti60 F225 C4	SP RAM	1:1	Disable	0/62016 (0%)	26/256 (10%)	0/160 (0%)	638	2023.1
			Enable	0/62016 (0%)	26/256 (10%)	0/160 (0%)	1010	
	SDP RAM	1:1	Disable	0/62016 (0%)	13/256 (5%)	0/160 (0%)	635	
			Enable	0/62016 (0%)	13/256 (5%)	0/160 (0%)	1028	
		1:2	Disable	0/62016 (0%)	32/256 (12%)	0/160 (0%)	642	
			Enable	0/62016 (0%)	32/256 (12%)	0/160 (0%)	1024	
	TDP RAM	1:1	Disable	0/62016 (0%)	13/256 (5%)	0/160 (0%)	659	
			Enable	0/62016 (0%)	13/256 (5%)	0/160 (0%)	1111	
		1:2	Disable	0/62016 (0%)	32/256 (12%)	0/160 (0%)	642	
			Enable	0/62016 (0%)	32/256 (12%)	0/160 (0%)	950	
	SP ROM	1:1	Disable	0/62016 (0%)	26/256 (10%)	0/160 (0%)	660	
			Enable	0/62016 (0%)	26/256 (10%)	0/160 (0%)	1054	
	DP ROM	1:1	Disable	0/62016 (0%)	13/256 (5%)	0/160 (0%)	1113	
			Enable	0/62016 (0%)	13/256 (5%)	0/160 (0%)	1116	
		1:2	Disable	0/62016 (0%)	32/256 (12%)	0/160 (0%)	1015	
			Enable	0/62016 (0%)	32/256 (12%)	0/160 (0%)	1033	

[1] Logic, Adders, Flipflops, etc.

Table 3: Asynchronous Clock BRAM Wrapper (True Dual-port RAM and Dual-port ROM)

clk_a/b in True Dual-port RAM is for simultaneous write and read operation for each respective port.
clk_a/b in Dual-port ROM is for read operation only for each respective port.

FPGA	Memory Type	Width Ratio	Output Pipeline Register	Logic Elements[1]	Memory Blocks	DSP Blocks	f_{MAX} (MHz)		Efinity Version
							clk_a	*clk_b*	
Ti60 F225 C4	TDP RAM	1:1	Disable	0/62016 (0%)	13/256 (5%)	0/160 (0%)	682	656	2023.1
			Enable	0/62016 (0%)	13/256 (5%)	0/160 (0%)	1082	1082	
		1:2	Disable	0/62016 (0%)	32/256 (12%)	0/160 (0%)	670	657	
			Enable	0/62016 (0%)	32/256 (12%)	0/160 (0%)	961	998	
	DP ROM	1:1	Disable	0/62016 (0%)	13/256 (5%)	0/160 (0%)	1011	1177	
			Enable	0/62016 (0%)	13/256 (5%)	0/160 (0%)	1097	1129	
		1:2	Disable	0/62016 (0%)	32/256 (12%)	0/160 (0%)	982	1048	
			Enable	0/62016 (0%)	32/256 (12%)	0/160 (0%)	1028	1053	

Table 4: Asynchronous Clock BRAM Wrapper (Simple Dual-port RAM)

FPGA	Memory Type	Width Ratio	Output Pipeline Register	Logic Elements[1]	Memory Blocks	DSP Blocks	f_{MAX} (MHz)		Efinity Version
							wclk	*rclk*	
Ti60 F225 C4	SDP RAM	1:1	Disable	0/62016 (0%)	13/256 (5%)	0/160 (0%)	1000	662	2023.1
			Enable	0/62016 (0%)	13/256 (5%)	0/160 (0%)	1000	1094	
		1:2	Disable	0/62016 (0%)	32/256 (12%)	0/160 (0%)	1000	641	
			Enable	0/62016 (0%)	32/256 (12%)	0/160 (0%)	1000	1002	

Trion Resource Utilization and Performance

Table 5: Synchronous Clock BRAM Wrapper

FPGA	Memory Type	Width Ratio	Output Pipeline Register	Logic Elements (1)	Memory Blocks	DSP Blocks	f_{MAX} (MHz) *clk*	Efinity Version
T120 F576 C4	SP RAM	1:1	Disable	0/112128 (0%)	52/1056 (5%)	0/320 (0%)	219	2023.1
			Enable	0/112128 (0%)	52/1056 (5%)	0/320 (0%)	306	
	SDP RAM	1:1	Disable	0/112128 (0%)	13/1056 (1%)	0/320 (0%)	229	
			Enable	0/112128 (0%)	13/1056 (1%)	0/320 (0%)	327	
		1:2	Disable	0/112128 (0%)	32/1056 (3%)	0/320 (0%)	222	
			Enable	0/112128 (0%)	32/1056 (3%)	0/320 (0%)	345	
	TDP RAM	1:1	Disable	0/112128 (0%)	4/1056 (1%)	0/320 (0%)	264	
			Enable	0/112128 (0%)	4/1056 (1%)	0/320 (0%)	383	
		1:2	Disable	0/112128 (0%)	32/1056 (3%)	0/320 (0%)	229	
			Enable	0/112128 (0%)	32/1056 (3%)	0/320 (0%)	356	
	SP ROM	1:1	Disable	0/112128 (0%)	26/1056 (2%)	0/320 (0%)	229	
			Enable	0/112128 (0%)	26/1056 (2%)	0/320 (0%)	342	
	DP ROM	1:1	Disable	0/112128 (0%)	13/1056 (1%)	0/320 (0%)	235	
			Enable	0/112128 (0%)	13/1056 (1%)	0/320 (0%)	383	
		1:2	Disable	0/112128 (0%)	32/1056 (3%)	0/320 (0%)	216	
			Enable	0/112128 (0%)	32/1056 (3%)	0/320 (0%)	342	

Table 6: Asynchronous Clock BRAM Wrapper (True Dual-port RAM and Dual-port ROM)

clk_a/b in True Dual-port RAM is for simultaneous write and read operation for each respective port.
clk_a/b in Dual-port ROM is for read operation only for each respective port.

FPGA	Memory Type	Width Ratio	Output Pipeline Register	Logic Elements(1)	Memory Blocks	DSP Blocks	f_{MAX} (MHz)		Efinity Version
							clk_a	*clk_b*	
T120 F576 C4	TDP RAM	1:1	Disable	0/112128 (0%)	13/1056 (1%)	0/320 (0%)	237	223	2023.1
			Enable	0/112128 (0%)	13/1056 (1%)	0/320 (0%)	404	383	
		1:2	Disable	0/112128 (0%)	32/1056 (3%)	0/320 (0%)	254	221	
			Enable	0/112128 (0%)	32/1056 (3%)	0/320 (0%)	377	330	
	DP ROM	1:1	Disable	0/112128 (0%)	13/1056 (1%)	0/320 (0%)	250	239	
			Enable	0/112128 (0%)	13/1056 (1%)	0/320 (0%)	390	355	
		1:2	Disable	0/112128 (0%)	32/1056 (3%)	0/320 (0%)	250	229	
			Enable	0/112128 (0%)	32/1056 (3%)	0/320 (0%)	379	334	

Table 7: Asynchronous Clock BRAM Wrapper (Simple Dual-port)

FPGA	Memory Type	Width Ratio	Output Pipeline Register	Logic Elements(1)	Memory Blocks	DSP Blocks	f_{MAX} (MHz)		Efinity Version
							wclk	*rclk*	
T120 F576 C4	SDP RAM	1:1	Disable	0/112128 (0%)	13/1056 (1%)	0/320 (0%)	478	219	2023.1
			Enable	0/112128 (0%)	13/1056 (1%)	0/320 (0%)	493	330	
		1:2	Disable	0/112128 (0%)	32/1056 (3%)	0/320 (0%)	389	225	
			Enable	0/112128 (0%)	32/1056 (3%)	0/320 (0%)	401	331	

Release Notes

You can refer to the IP Core Release Notes for more information about the IP core changes. The IP Core Release Notes is available in the Efinity Downloads page under each Efinity software release version.

Note: You must be logged in to the Support Portal to view the IP Core Release Notes.

Functional Description

The BRAM Wrapper core includes the following ports for read and write access to the memory:

- Single-port RAM—Port A for both write and read access
- Simple Dual-port RAM—Port A for write access and port B for read access
- True Dual-port RAM—Port A and port B can be either write or read access
- Single-port ROM—Port B for read access only
- Dual-port ROM—Port A and Port B for read access only

Figure 1: Single-Port RAM Block Diagram

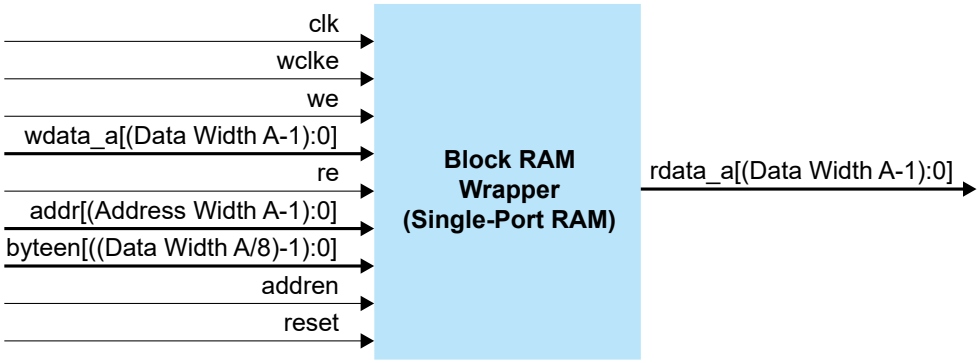


Figure 2: Simple Dual-Port RAM Block Diagram

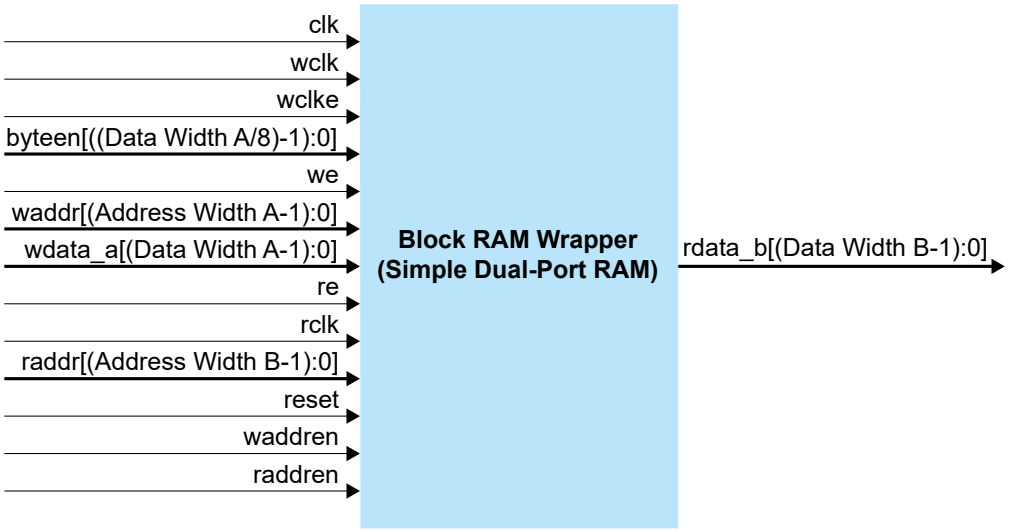


Figure 3: True Dual-Port RAM Block Diagram

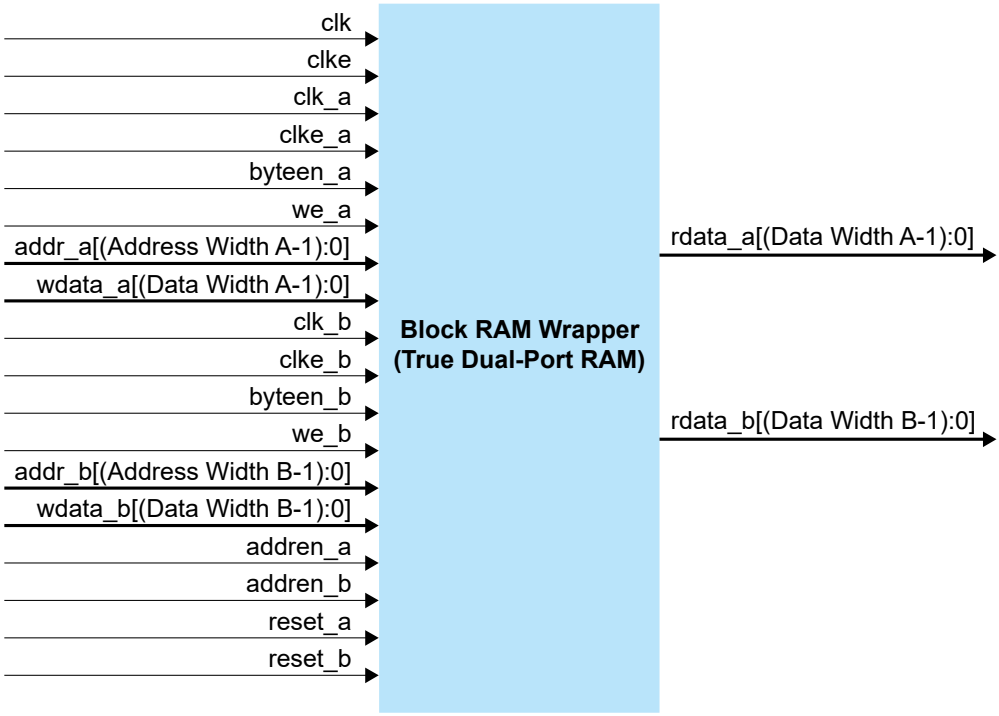


Figure 4: Single-Port ROM Block Diagram

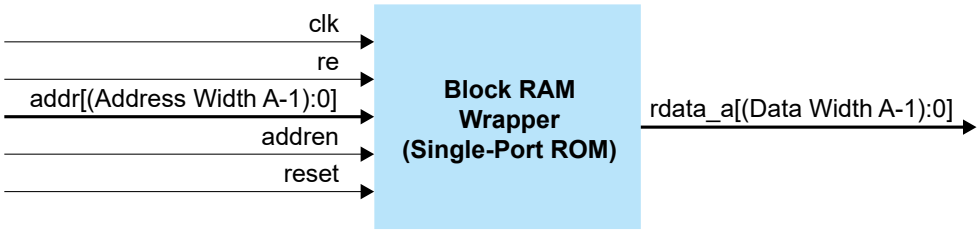
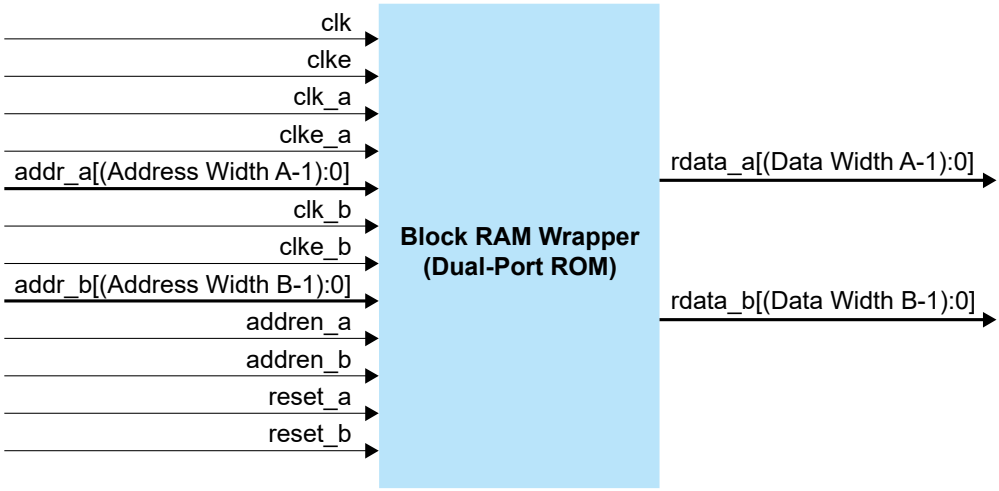


Figure 5: Dual-Port ROM Block Diagram



Ports

Table 8: Single-Port RAM Ports

Port	Direction	Clock Domain	Description
clk	Input	–	Clock input.
wclke	Input	clk	Write clock-enable input.
we	Input	clk	Write-enable input.
re	Input	clk	Read-enable input.
wdata_a[Data Width A-1:0]	Input	clk	Write data input.
rdata_a[Data Width A-1:0]	Output	clk	Read data output.
addr[Address Width A-1:0]	Input	clk	Address input.
byteen[(Data Width A/8)-1:0]	Input	clk	Byteen input. This pin is responsible to mask off a byte of data for each bit. The minimum byteen bus width is 1.
reset	Input	clk	Reset pin. Applicable to 钛金系列 FPGAs only.
addren	Input	clk	Enable or disable **Address Update** on page 20 operation. Applicable to 钛金系列 FPGAs only.

Table 9: Simple Dual-Port RAM Ports

Port	Direction	Clock Domain	Description
clk	Input	–	Clock input for Single-clock mode.
wclk	Input	–	Write clock input for Dual-Clock Mode.
wclke	Input	clk/wclk	Write clock-enable input.
byteen [(Data Width A/8)-1:0]	Input	clk/wclk	Byteen input. The minimum byteen bus width is 1.
we	Input	clk/wclk	Write-enable input.
waddr[(Address Width A-1):0]	Input	clk/wclk	Write address input.
wdata_a[(Data Width A-1:0]	Input	clk/wclk	Write data input.
rclk	Input	–	Read clock input for Dual-Clock Mode.
re	Input	clk/rclk	Read-enable input.
raddr[Address Width B-1:0]	Input	clk/rclk	Read address input.
rdata_b[Data Width B-1:0]	Output	clk/rclk	Read data output.
reset	Input	clk/rclk	Reset pin. Applicable to 钛金系列 FPGAs only.
waddren	Input	clk/wclk	Enable or disable write **Address Update** on page 20 operation. Applicable to 钛金系列 FPGAs only.
raddren	Input	clk/rclk	Enable or disable read **Address Update** on page 20 operation. Applicable to 钛金系列 FPGAs only.

Table 10: True Dual-Port RAM Ports

Port	Direction	Clock Domain	Description
clk	Input	–	Clock input in single-clock mode.
clke	Input	clk	Clock-enable input.
clk_a	Input	–	Clock input for Port A in dual-clock mode.
clke_a	Input	clk_a	Clock-enable input for Port A.
byteen_a	Input	clk/clk_a	ByteEnable input for Port A.
we_a	Input	clk/clk_a	Write-enable for Port A.
addr_a[(Address Width A-1):0]	Input	clk/clk_a	Write and read address input for Port A.
wdata_a[(Data Width A-1:0]	Input	clk/clk_a	Write data input for Port A.
rdata_a[(Data Width A-1:0]	Output	clk/clk_a	Read data output for Port A.
clk_b	Input	–	Clock input for Port B in dual-clock mode.
clke_b	Input	clk_b	Clock-enable input for Port B.
byteen_b	Input	clk/clk_b	ByteEnable input for Port B.
we_b	Input	clk/clk_b	Write-enable for Port B.
addr_b[(Address Width B-1):0]	Input	clk/clk_b	Write and read address input for Port B.
wdata_b[(Data Width B-1:0]	Input	clk/clk_b	Write data input for Port B.
rdata_b[(Data Width B-1:0]	Output	clk/clk_b	Read data output for Port B.
reset_a	Input	clk/clk_a	Reset for Port A. Applicable to 钛金系列 FPGAs only.
addren_a	Input	clk/clk_a	Enable or disable write **Address Update** on page 20 operation via Port A. Applicable to 钛金系列 FPGAs only.
reset_b	Input	clk/clk_b	Reset for Port B. Applicable to 钛金系列 FPGAs only.
addren_b	Input	clk/clk_b	Enable or disable write **Address Update** on page 20 operation via Port B. Applicable to 钛金系列 FPGAs only.

Table 11: Single-Port ROM Ports

Port	Direction	Clock Domain	Description
clk	Input	–	Clock input for single-clock mode.
re	Input	clk	Read-enable input.
rdata_a[Data Width A-1:0]	Output	clk	Read data output.
addr[Address Width A-1:0]	Input	clk	Address input.
reset	Input	clk	Reset pin. Applicable to 钛金系列 FPGAs only.
addren	Input	clk	Enable or disable **Address Update** on page 20 operation. Applicable to 钛金系列 FPGAs only.

Table 12: Dual-Port ROM Ports

Port	Direction	Clock Domain	Description
clk	Input	–	Clock input in one clock mode.
clke	Input	clk	Clock-enable input.
clk_a	Input	–	Clock input for Port A in dual clock mode.
clke_a	Input	clk_a	Clock-enable input for Port A.
addr_a[(Address Width A-1):0]	Input	clk/clk_a	Write and read address input for Port A.
rdata_a[(Data Width A-1:0]	Output	clk/clk_a	Read data output for Port A.
clk_b	Input	–	Clock input for Port B in dual clock mode.
clke_b	Input	clk_b	Clock-enable input for Port B.
addr_b[(Address Width B-1):0]	Input	clk/clk_b	Write and read address input for Port B.
rdata_b[(Data Width B-1:0]	Output	clk/clk_b	Read data output for Port B.
reset_a	Input	clk/clk_a	Reset for Port A. Applicable to 钛金系列 FPGAs only.
addren_a	Input	clk/clk_a	Enable or disable write **Address Update** on page 20 operation via Port A. Applicable to 钛金系列 FPGAs only.
reset_b	Input	clk/clk_b	Reset for Port B. Applicable to 钛金系列 FPGAs only.
addren_b	Input	clk/clk_b	Enable or disable write **Address Update** on page 20 operation via Port B. Applicable to 钛金系列 FPGAs only.

Mixed Width Port Configuration

Simple Dual-port RAM, True Dual-port RAM, and Dual-port ROM support mixed-width port configuration where you can set port A and port B to have different data widths. The support for mixed-width port depends on the width ratio between port A and port B. You only set the **Width Ratio**, **Port A: Data Bus Width**, and **Port A: BRAM Depth** in the IP Manager. The core automatically set the **Data Bus Width** and **BRAM Depth** for Port B based on the ratio you select.

The ratio of the depths is always the inverse of the ratio of the widths. For instance, if the **Write Data Width** interface for Port A is 32 and **Read Data Width** for Port B is 8 (4:1 ratio), then the memory depth ratio for Port A:Port B is 1:4.

Memory Modes

The BRAM Wrapper core supports area-optimized and speed-optimized modes for all RAM and ROM.

Table 13: BRAM Wrapper Memory Modes

Memory Mode	Description
Area-Optimized	• Utilizes minimum number of block RAM • Performance slightly lower than speed optimized mode • Supported in symmetrical port width configuration only
Speed-Optimized	• Utilizes minimal the number of multiplexer, address encoder logic and fanout for better performance • Costs more embedded block memory resources • Supported in symmetrical and asymmetrical port width configuration

The following table shows the logic utilization and f_{MAX} comparison of a design with the memory mode set to area-optimized and speed-optimized. The design uses the default BRAM Wrapper core parameters except for the **BRAM Depth** which is set to 8192.

Table 14: Area vs. Speed Optimization

FPGA	Memory Type	Memory Mode	Logic Elements	Memory Blocks	f_{MAX} (MHz)	Efinity Version
Ti60 F225 C4	SDP RAM	Area	36/62016 (1%)	13/256 (5%)	536	2023.1
		Speed	0/62016 (0%)	16/256 (6%)	648	

Note: You may not notice significant performance differences in designs that use smaller **BRAM Depth**. Generally, the higher the **BRAM Depth**, the greater the performance differences tend to be.

Write Modes

The BRAM Wrapper core supports four different write mode during write operation to meet the desired output value during read operations.

Table 15: Supported RAM and ROM Write Mode

Write Mode	SP RAM	SDP RAM	TDP RAM	SP ROM	DP ROM
Write First	✓	✓	✓	-	-
Read First	✓	✓	✓	✓	✓
No Change	-	-	✓	-	-
Read Unknown	✓	✓	-	-	-

The following truth table describes the memory function logic for the supported write modes.

Table 16: Write Mode Memory Function Truth Table

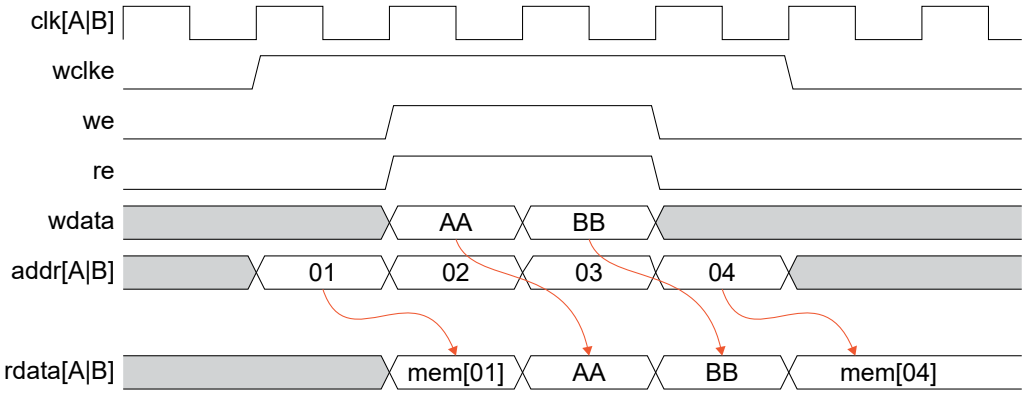
Write Mode	clk[A/B]	clke[A/B]	we[A/B]	re[A/B](2)	wdata[A/B]	rdata[A/B]
Write First	Logic high	Logic high	Logic high	Logic high	Data in	Data in
Read First	Logic high	Logic high	Logic high	Logic high	Data in	Read out previously written data
No change	Logic high	Logic high	Logic high	-	Data in	Retain the same data of the last read data
Read Unknown	Logic high	Logic high	Logic high	Logic high	Data in	Unknown value

(2) Applicable to Single-port and Simple Dual-port RAM only.

Write First Mode

In Write First mode, the input data is simultaneously written into the memory and driven on the read data output.

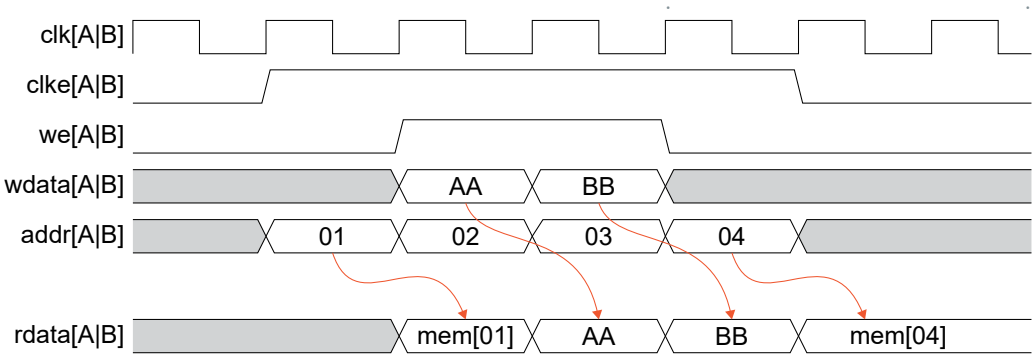
Figure 6: Write First Mode Example (Single-Port and Simple Dual-Port RAM)



Note: For Simple Dual-port RAM in Write First mode, both read and write port data width and address must be identical.

Figure 7: Write First Example (True Dual-port RAM)

The [A|B] signal labels are referring to either port at one time Not both ports at the same time



Read First Mode

In Read First mode, the read data output returns the data that was previously stored at each respective address. At the same time, writing new input data into the memory.

Figure 8: Read First Mode Example (Single-Port and Simple Dual-Port RAM)

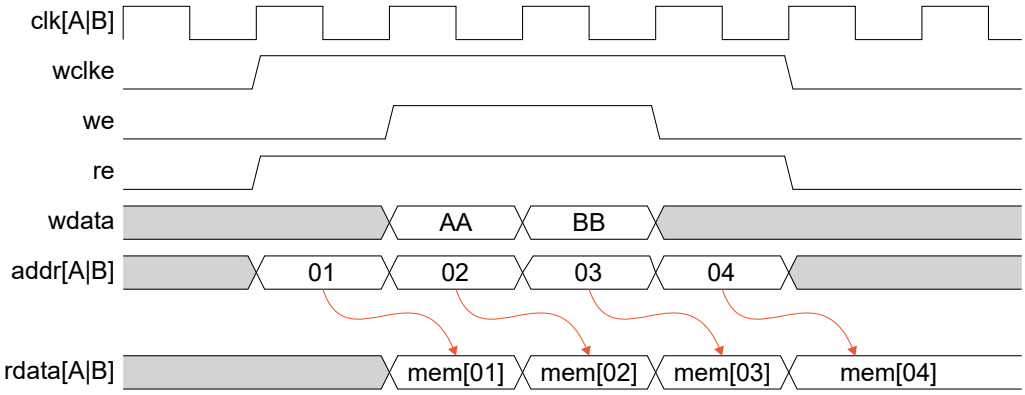
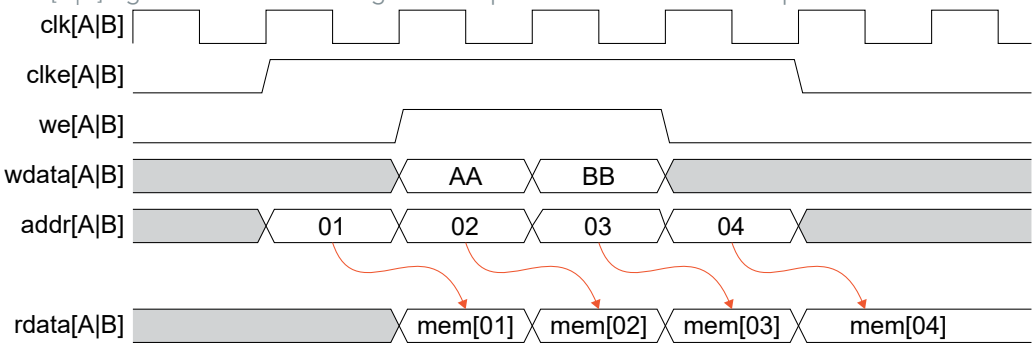


Figure 9: Read First Mode Example (True Dual-Port RAM)

The [A|B] signal labels are referring to either port at one time. Not both ports at the same time.



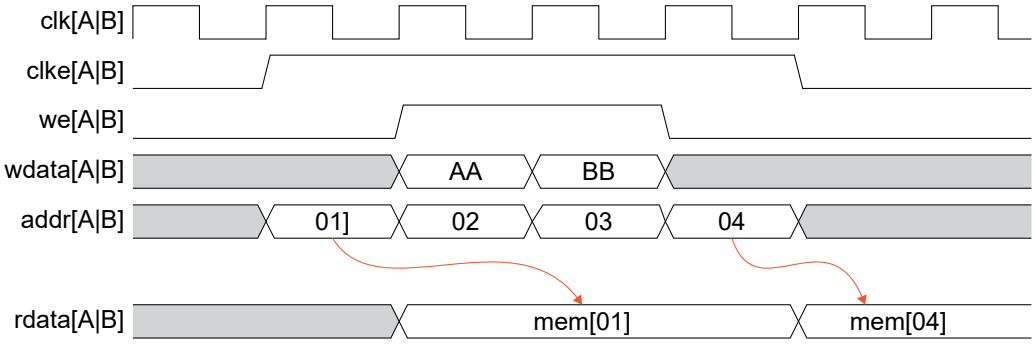
Learn more: Refer to Conflict Resolution on page 18 if the data input on port A and Port B are different in True Dual-Port RAM.

No Change Mode

In No Change mode, the `rdata[A|B]` (read data) remains unchanged if both `we[A|B]` and `clke[A|B]` signals are high. However, if `we[A|B]` signal goes low, the RAM behaves as the Read First mode where it returns the previous stored data in the corresponding address as occurred in address 1 and address 4 in the following figure.

Figure 10: No Change Mode Example (True Dual-Port RAM)

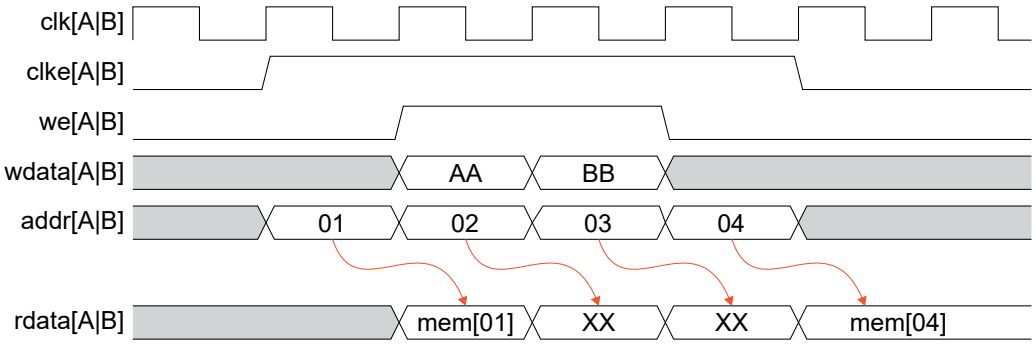
The [A|B] in signal labels refers to either port at one time. Not both ports at the same time.



Read Unknown Mode

In Read Unknown mode, the write and read are unsynchronized, therefore the results of the address can conflict which returns undeterministic output, `XX`.

Figure 11: Read Unknown Mode Example (Single-Port and Simple Dual-Port RAM)



Conflict Resolution

The True Dual-port RAM can perform two write operations to the same memory location for both data from port A and port B. Accessing the same memory location from both ports in the same clock cycle will cause address collisions. There is no internal circuitry to prevent this conflict. Therefore, user must resolve it externally during configuration to prevent unknown data being written to the address.

The following table illustrates the Read Data Port A, Read Data Port B outcome and the resulting RAM content based on Write Mode [A|B] and Write Enable [A|B] configuration.

Table 17: Resulting RAM Content based on Write Mode[A|B] and Write Enable[A|B]

Write Enable[A|B] = 1 : Simultaneous Write and Read

Write Enable[A|B] = 0 : Read only

Write Mode Port A	Write Mode Port B	Write Enable Port A	Write Enable Port B	Read Data Port A	Read Data Port B	Resulting RAM content
Read First	Read First	0	0	Valid	Valid	Valid
		0	1	Valid	Valid	Valid
		1	0	Valid	Valid	Valid
		1	1	Valid	Valid	Unknown
Read First	Write First	0	0	Valid	Valid	Valid
		0	1	Valid	Valid	Valid
		1	0	Valid	Valid	Valid
		1	1	Valid	Valid	Unknown
Read First	No Change	0	0	Valid	Valid	Valid
		0	1	Valid	Valid	Valid
		1	0	Valid	Valid	Valid
		1	1	Valid	Valid	Unknown
Write First	Read First	0	0	Valid	Valid	Valid
		0	1	Valid	Valid	Valid
		1	0	Valid	Valid	Valid
		1	1	Valid	Valid	Unknown
Write First	Write First	0	0	Valid	Valid	Valid
		0	1	Valid	Valid	Valid
		1	0	Valid	Valid	Valid
		1	1	Valid	Valid	Unknown
Write First	No Change	0	0	Valid	Valid	Valid
		0	1	Valid	Valid	Valid
		1	0	Valid	Valid	Valid
		1	1	Valid	Valid	Unknown
No Change	Read First	0	0	Valid	Valid	Valid
		0	1	Valid	Valid	Valid
		1	0	Valid	Valid	Valid

Write Mode Port A	Write Mode Port B	Write Enable Port A	Write Enable Port B	Read Data Port A	Read Data Port B	Resulting RAM content
		1	1	Valid	Valid	Unknown
No Change	Write First	0	0	Valid	Valid	Valid
		0	1	Valid	Valid	Valid
		1	0	Valid	Valid	Valid
		1	1	Valid	Valid	Unknown
No Change	No Change	0	0	Valid	Valid	Valid
		0	1	Unknown	Valid	Valid
		1	0	Valid	Unknown	Valid
		1	1	Valid	Valid	Unknown

Clock Mode

The BRAM Wrapper core supports single clock and dual clock clocking modes depending on the memory type you choose.

Table 18: Clock Mode by Memory Type

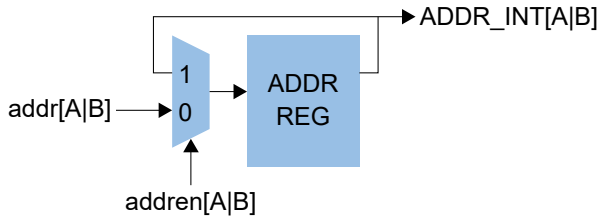
Clock Mode	Description	SP RAM	SDP RAM	TDP RAM	SP ROM	DP ROM
Single Clock	One clock along with clock enable to control all registers of the memory block.	✓	✓	✓	✓	✓
Dual Clock	**TDP RAM and DP ROM:** Clock A controls all registers on the port A while Clock B control all registers on the port B. **SDP RAM:** Write clock controls all register for the write operation while read clock controls all registers for the read operation.	–	✓[3]	✓	–	✓

[3] Only single clock mode for Write First and Read First memory behaviour settings.

Address Update

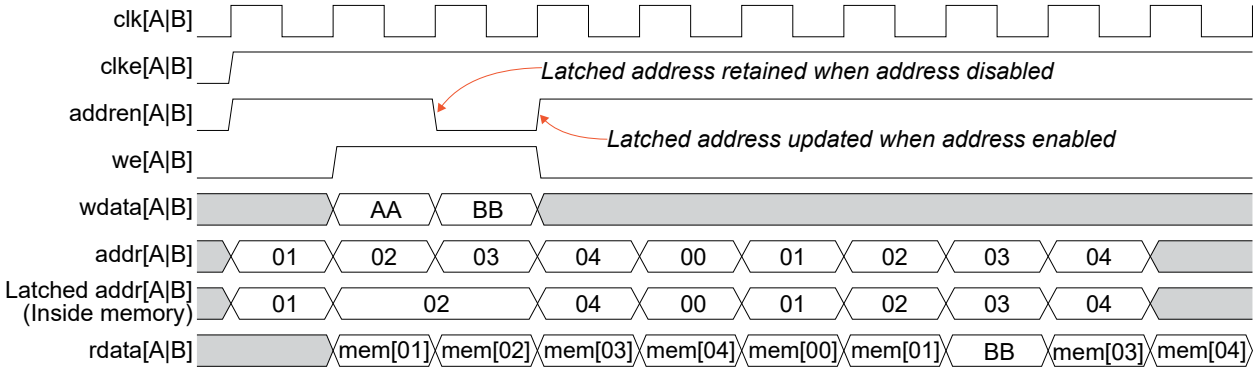
The addren signal is an additional clock enable signal for the address register which allows you to enable or disable (stall) the address update. When disabled while performing read/write operation, it holds the previous address value. The default value for the address enable is high (enabled).

Figure 12: Address Update Block



In the following example, the data BB on rdata overwrites the previously written AA in address addr[02] since the addren signal is low when the we signal is high. The data BB will not be written to address addr[03] so rdata outputs the previously stored data.

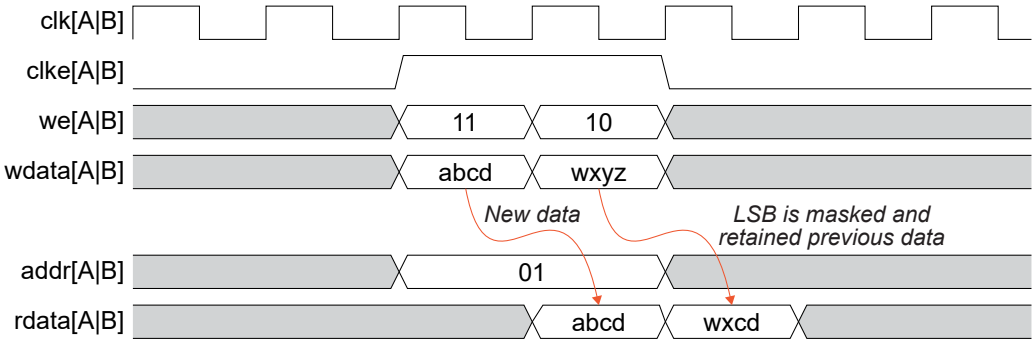
Figure 13: Address Update Example in Write Example



Byte Enable

The embedded memory block supports byte enable feature which allows you to mask off unwanted byte during a write operation. The unwritten (masked) bytes retain the previously written data. Each bit of byte-enable (`byteen`) port corresponds to 1 byte of data when the Byte Enable feature is enabled. For example, if you use a RAM block in the x16 mode with byte-enable port is 10, data [15:8] is enabled while data [7:0] is disabled.

Figure 14: Byte Enable Control Example



Read Enable

Single-port RAM and Simple Dual-port RAM support the control of the read-enable feature during write operations. When disabled, the data output will retain the last value that was held during the most recent active read enable. If enabled, the data output value will depend on the **Write Mode** selected.

Optional Output Register

The BRAM Wrapper core includes an optional output register (pipeline) feature that can improve the core performance. In True Dual-port RAM, you can enable the optional output register for independent Port A and Port B separately. Note that each optional register stage used adds an additional clock cycle latency to the Read operation. Enabled output register at the output of the primitive block reduces the impact of the clock-to-out delay of the primitives.

Optional Reset Pin

The block RAM in 钛金系列 FPGAs support optional reset pin. The reset pin resets last register in the output stage of primitive or primitive memory output latches if the output register is disabled. It does not reset the RAM or ROM content.

Memory Initialization Files

By default, The BRAM Wrapper core initialized memory value are all zeros. You can initialize memory with your own values by enabling the **Initialize BRAM Memory File** setting and providing the memory initialization file (in **.mem** format).

The memory initialization file has the following requirements:

- Data created in the memory file must be equal or less than the configured data width. Extra bits are ignored.
- Data separated by **Return/Enter** key are treated as value for the subsequent address.
- Values after a double slash (//) within the same line are ignored so that you can label each data according to respective address to avoid confusion.

Table 19: Memory File Data Example

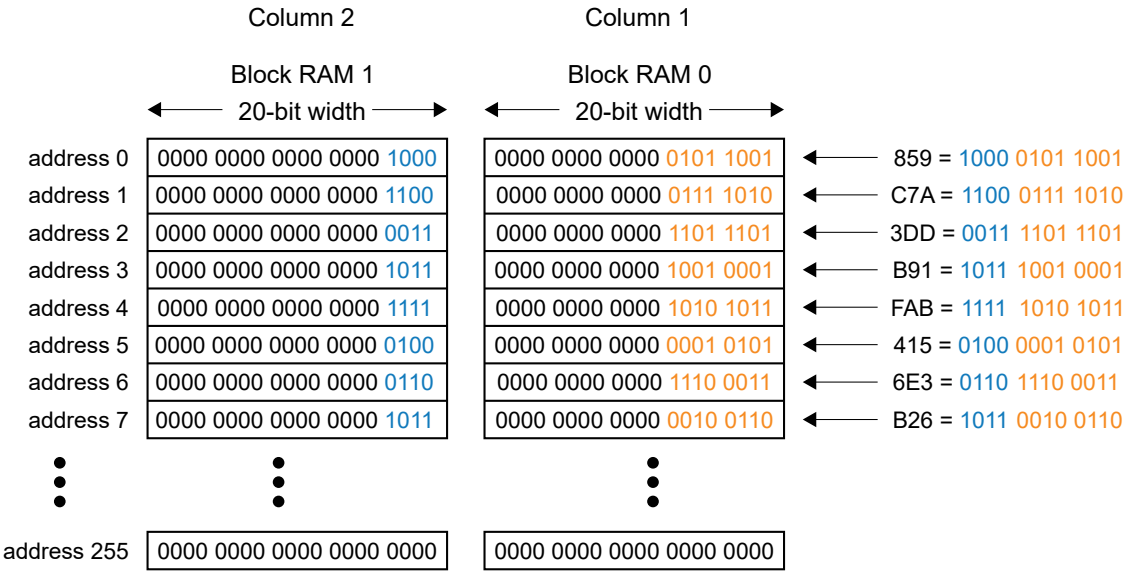
Data	Address
859	0
C7A	1
3DD	2
B91	3
FAB	4
415	5
6E3	6
B26	7

Figure 15: .mem File Example

```
//data separated by <Return/Enter> key
//you can comment or label your data using double slash
//the next line is always next address unless the entire line is commented out
859 //address 0
C7A //address 1
3DD
B91
415
6E3
B26
```

The following figure illustrates how the block RAM are updated.

Figure 16: Block RAM Initialization Example



Block RAM 0 = 0000 0000 0000 0010 0110 0000 0000 0000 1110 0011 0000 0000 0000 0001
0101 0000 0000 0000 1010 1011 0000 0000 0000 1001 0001 0000 0000 0000
1101 1101 0000 0000 0000 0111 1010 0000 0000 0000 0101 1001

= 26000E300015000AB00091000DD0007A00059

Block RAM 1 = 0000 0000 0000 0000 1011 0000 0000 0000 0000 0110 0000 0000 0000 0000
0100 0000 0000 0000 0000 1111 0000 0000 0000 0000 1011 0000 0000 0000
0000 0011 0000 0000 0000 0000 1100 0000 0000 0000 0000 1000

= B00006000040000F0000B000030000C00008

The BRAM 0 and BRAM 1 in the **bram_ini.vh** file are updated as below:

```
function [255:0] bram_ini_table;
input integer index;//Mode type
input integer val_; //Port A index, Port B Index, Number of Items in Loop, Port A Start, Port B
 Start, reserved
case (index)
   0: bram_ini_table=
(val_== 0)?256'h00000000000000000000000000026000e300015000ab00091000dd0007a00059:
(val_== 1)?256'h00000000000000000000000000000000000000000000000000000000000000000:
(val_== 2)?256'h00000000000000000000000000000000000000000000000000000000000000000:
...
(val_== 39)?256'h00000000000000000000000000000000000000000000000000000000000000000:
0;
   1: bram_ini_table=
(val_== 0)?256'h0000000000000000000000000b0000600004000f0000b000030000c00008:
(val_== 1)?256'h00000000000000000000000000000000000000000000000000000000000000000:
(val_== 2)?256'h00000000000000000000000000000000000000000000000000000000000000000:
...
(val_== 39)?256'h00000000000000000000000000000000000000000000000000000000000000000:
0;
   endcase
endfunction
```

IP Manager

The Efinity® IP Manager is an interactive wizard that helps you customize and generate 易灵思® IP cores. The IP Manager performs validation checks on the parameters you set to ensure that your selections are valid. When you generate the IP core, you can optionally generate an example design targeting an 易灵思 development board and/or a testbench. This wizard is helpful in situations in which you use several IP cores, multiple instances of an IP core with different parameters, or the same IP core for different projects.

Note: Not all 易灵思 IP cores include an example design or a testbench.

Generating the BRAM Wrapper Core with the IP Manager

The following steps explain how to customize an IP core with the IP Configuration wizard.

1. Open the IP Catalog.
2. Choose **Memory > BRAM Wrapper** core and click **Next**. The **IP Configuration** wizard opens.
3. Enter the module name in the **Module Name** box.

   Note: You cannot generate the core without a module name.
4. Customize the IP core using the options shown in the wizard. For detailed information on the options, refer to the *Customizing the BRAM Wrapper* section.
5. (Optional) In the **Deliverables** tab, specify whether to generate an IP core example design targeting an Efinix® development board and/or testbench. These options are turned on by default.易灵思
6. (Optional) In the **Summary** tab, review your selections.
7. Click **Generate** to generate the IP core and other selected deliverables.
8. In the **Review configuration generation** dialog box, click **Generate**. The Console in the **Summary** tab shows the generation status.

   Note: You can disable the **Review configuration generation** dialog box by turning off the **Show Confirmation Box** option in the wizard.
9. When generation finishes, the wizard displays the **Generation Success** dialog box. Click **OK** to close the wizard.

The wizard adds the IP to your project and displays it under **IP** in the Project pane.

Generated Files

The IP Manager generates these files and directories:

- **<module name>_define.vh**—Contains the customized parameters.
- **<module name>_tmpl.v**—Verilog HDL instantiation template.
- **<module name>_tmpl.vhd**—VHDL instantiation template.
- **<module name>.v**—IP source code.
- **settings.json**—Configuration file.
- **<kit name>_devkit**—Has generated RTL, example design, and Efinity® project targeting a specific development board.
- **Testbench**—Contains generated RTL and testbench files.

Customizing the BRAM Wrapper

The core has parameters so you can customize its function. You set the parameters in the General tab of the core's IP Configuration window.

Table 20: BRAM Wrapper Core Master Parameters

Parameter	Options	Memory Type	Description
Device Family	TITANIUM, TRION	All	Select the target device family. Default: TITANIUM
Memory Type	Single-port RAM, Simple Dual-port RAM, True Dual-port RAM, Single-port ROM, Dual-port ROM	All	Select the target memory type. Default: Simple Dual-port RAM
Memory Mode	Speed, Area	All	Select to optimize speed or area. Default: Speed
Write Mode	READ_FIRST, WRITE_FIRST, READ_UKNOWN,	Single-port RAM, Simple Dual-port RAM	Define memory configuration: READ_FIRST: Old memory content is read. WRITE_FIRST: Write data is passed to the read port. READ_UNKNOWN: Read and write are unsynchronized, therefore the results of the address can conflict. Default: READ_FIRST
Write Mode A	READ_FIRST, WRITE_FIRST, NO_CHANGE	True Dual-port RAM	Define memory behaviour: READ_FIRST: Old memory content is read. WRITE_FIRST: Write data is passed to the read port. NO_CHANGE: The output keeps the memory content of last read address if the write operation is applied. Default: READ_FIRST
Write Mode B	READ_FIRST, WRITE_FIRST, NO_CHANGE	True Dual-port RAM	Define memory behaviour: READ_FIRST: Old memory content is read. WRITE_FIRST: Write data is passed to the read port. NO_CHANGE: The output keeps the memory content of last read address if the write operation is applied. Default: READ_FIRST
Width Ratio	16:1, 8:1, 4:1, 2:1, 1:1, 1:2, 1:4, 1:8, 1:16	Simple Dual-port RAM, True Dual-port RAM, Dual-port ROM	Select symmetrical or asymmetrical width ratios. Default: 1:1
Port A: Data Bus Width	1-1024	All	Define BRAM write data bus width. The width is multiple of 2 from 2^0 - 2^{10}. Default: 16

Parameter	Options	Memory Type	Description
Port B: Data Bus Width	-	Single-port RAM, Simple Dual-port RAM, True Dual-port RAM, Dual-port ROM	Define BRAM read data bus width. The width is auto calculated by the choice of width ratio. Default: 16
Port A: BRAM Depth	1-16384	All	Define the BRAM depth. The depth is multiples of 2 from 2^1 - 2^{13}. Default: 8
Port B: BRAM Depth	-	Single-port RAM, Simple Dual-port RAM, True Dual-port RAM, Dual-port ROM	Define the BRAM depth. The depth is auto calculated via Width Ratio selections. Default: 8
Byte Enable	Enable, Disable	Single-port RAM, Simple Dual-port RAM, True Dual-port RAM	Define a group of data to be masked. Default: Disable
Group Data Width	-	Single-port RAM, Simple Dual-port RAM, True Dual-port RAM	Support 8 bit data per group only.
Byte Enable Width	-	Single-port RAM, Simple Dual-port RAM, True Dual-port RAM	Define number of groups. Auto calculated.
Write Enable	Enable, Disable	Single-port RAM, Simple Dual-port RAM	Define Enable Write operations. Default: Enable
Port A: Write Enable	Enable, Disable	True Dual-port RAM	Define Enable Write operations via Port A. Default: Enable
Port B: Write Enable	Enable, Disable	True Dual-port RAM	Define Enable Write operations via Port B. Default: Enable
Read Enable	Enable, Disable	Single-port RAM, Simple Dual-port RAM, Single-port ROM	Define Enable read operations. Default: Enable
Write Address Enable	Enable, Disable	Single-port RAM, Simple Dual-port RAM	Define address stall during write operations. Default: Enable
Read Address Enable	Enable, Disable	Single-port RAM, Simple Dual-port RAM	Define address stall during read operation. Default: Enable
Port A: Address Enable	Enable, Disable	True Dual-port RAM	Define address stall during write/read operation via Port A. Default: Enable
Port B: Address Enable	Enable, Disable	True Dual-port RAM	Define address stall during write/read operation via Port B. Default: Enable
Reset Enable	Enable, Disable	Single-port RAM, Simple Dual-port RAM, Single-port ROM	Define BRAM reset pin. Default: Enable
Port A: Reset	Enable, Disable	True Dual-port RAM, Dual-port ROM	Define BRAM reset via Port A. Default: Enable
Port B: Reset	Enable, Disable	True Dual-port RAM, Dual-port ROM	Define BRAM reset via Port B. Default: Enable

Parameter	Options	Memory Type	Description
Write Clock Enable	Enable, Disable	Single-port RAM, Simple Dual-port RAM	Define clock gate for write operation. Default: Enable
Output Register Pipeline	Enable, Disable	Single-port RAM, Simple Dual-port RAM, Single-port ROM	Add pipeline stage to read data to improve performance. Default: Disable
Port A: Output Register Pipeline	Enable, Disable	True Dual-port RAM, Dual-port ROM	Add pipeline stage to read data to improve performance. Default: Disable
Port B: Output Register Pipeline	Enable, Disable	True Dual-port RAM, Dual-port ROM	Add pipeline stage to read data to improve performance. Default: Disable
Clock Mode	Single Clock, Dual Clock	Simple Dual-port RAM, True Dual-port RAM, Dual-port ROM	Define BRAM clock dependency for read and write operations. Default: Single Clock
Initialize BRAM Memory File	Enable, Disable	All	Define user input BRAM Memory file. See **Memory Initialization Files** on page 22. Default: Disable
Memory File (.mem)	-	All	Enter the path to your target user application. The file must be in **.mem** format.

BRAM Wrapper Example Design

You can choose to generate the example design when generating the core in the IP Manager Configuration window. Compile the example design project and download the **.hex** or **.bit** file to your board.

Important: 易灵思 tested the example design generated with the default parameter options only.

The example design targets the 钛金系列 Ti60 F225 and Trion T120 F576 Development Boards. The design demonstrates write and read operations with the default settings. The system reset is tied to logic low so that the example design runs from start of write state to the end of read state and enters the idle state. The test pattern block produces 16-bit data width which is directly written into the Simple Dual-port RAM (configured as 1:1 ratio and synchronous clock setting).

Then, the read data from the test pattern block are compared with the read data from Simple Dual-port RAM on every clock cycle. You can observe the LED light indicator to observe the pass or fail status.

Figure 17: BRAM Wrapper Example Design Block Diagram

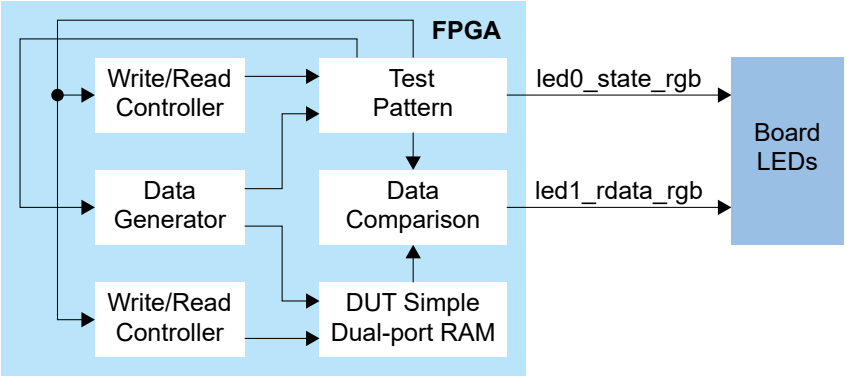


Table 21: LED Outputs for 钛金系列 Ti60 F225 Development Board Design Example

LED	Status	Description
D16	All on	Power-good
	All off	Configuration done
D17	Blue on	Comparison passed
	Red on	Comparison failed

Table 22: LED Outputs for Trion T120 F576 Development Board Design Example

LED	Status	Description
D1, D2, D3	000	Configuration done
D3, D4, D5	100	Comparison passed
	001	Comparison failed

Table 23: Design Example Implementation

FPGA	Memory Type	Logic Elements	Memory Blocks	DSP Blocks	f_{MAX} (MHz)	Efinity Version
Ti60 F225 C4	SDP RAM	82/62016 (1%)	1/256 (1%)	0/160 (0%)	465	2023.1
T120 F576 C4		62/112128 (1%)	1/1056 (1%)	0/320 (0%)	160	

BRAM Wrapper Testbench

You can choose to generate the testbench when generating the core in the IP Manager Configuration window.

Note: You must include all **.v** files generated in the **/testbench** directory in your simulation.

易灵思 provides a simulation script for you to run the testbench quickly using the Modelsim software. To run the Modelsim testbench script, run `vsim -do modelsim.do` in a terminal application. You must have Modelsim installed on your computer to use this script.

Revision History

Table 24: Revision History

Date	Version	Description
September 2023	1.4	Added Conflict Avoidance topic for True Dual-port RAM Configuration. (DOC-1466)
June 2023	1.3	Added Device Support section. (DOC-1234) Updated for IP Manager release in Efinity version 2023.1. (DOC-1308)
March 2023	1.2	Corrected maximum supported address width. (DOC-1182)
October 2022	1.1	Added support for BRAM memory initialization, Simple Dual-port RAM, and True Dual-port RAM.
August 2022	1.0	Initial release.