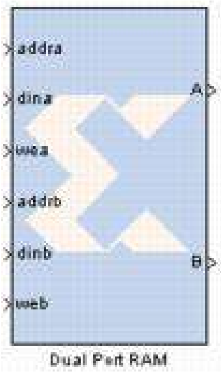
Dual Port RAM

This block is listed in the following Xilinx Blockset libraries: Control Logic, Memory, Floating-Point and Index.



The Xilinx Dual Port RAM block implements a random access memory (RAM). Dual ports enable simultaneous access to the memory space at different sample rates using multiple data widths.

Block Interface

The block has two independent sets of ports for simultaneous reading and writing. Independent address, data, and write enable ports allow shared access to a single memory space. By default, each port set has one output port and three input ports for address, input data, and write enable. Optionally, you can also add a port enable and synchronous reset signal to each input port set.

A dual-port RAM can be implemented using either distributed memory, block RAM, or UltraRAM resources in the FPGA.

Form Factors

The Dual Port RAM block also supports various Form Factors (FF). Form factor is defined as:

$FF = W_B / W_A$ where W_B is data width of Port B and W_A is Data Width of Port A.

The Depth of port B (D_B) is inferred from the specified form factor as follows:

$D_B = D_A / FF$.

The data input ports on Port A and B can have different arithmetic type and binary point position for a form factor of 1. For form factors greater than 1, the data input ports on Port A and Port B should have an unsigned arithmetic type with binary point at 0. The output ports, labeled A and B, have the same types as the corresponding input data ports.

The location in the memory block can be accessed for reading or writing by providing the valid address on each individual address port. A valid address is an unsigned integer from 0 to d-1, where d denotes the RAM depth (number of words in the RAM) for the particular port. An attempt to read past the end of the memory is caught as an error in simulation. When the dual-port RAM is implemented in distributed memory or block RAM, the initial RAM contents can be specified through a block parameter. Each write enable port must be a boolean value. When the WE port is 1, the value on the data input is written to the location indicated by the address line.

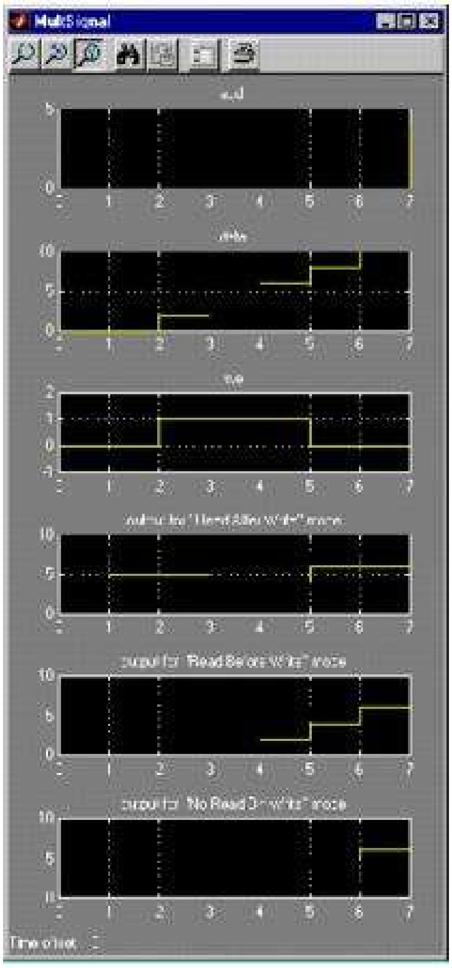
Write Mode

When the Dual Port RAM block is implemented in block RAM, you can set the write mode for the block in the block parameters dialog box.

The output during a write operation depends on the write mode. When the WE is 0, the output port has the value at the location specified by the address line. During a write operation (WE asserted), the data presented on the input data port is stored in memory at the location selected by the port's address input. During a write cycle, you can configure the behavior of each data out port A and B to one of the following choices:

- **Read after write**
- **Read before write**
- **No read on write**

The write modes can be described with the help of the figure below. In the figure, the memory has been set to an initial value of 5 and the address bit is specified as 4. When using **No read on write** mode, the output is unaffected by the address line and the output is the same as the last output when the WE was 0. For the other two modes, the output is obtained from the location specified by the address line, and hence is the value of the location being written to. This means that the output can be the old value which corresponds to **Read after write**.


MultiSignal
addr
write
we
output for "Read After Write" mode
output for "Read Before Write" mode
output for "No Read On Write" mode
0
1
2
3
4
5
6
7
10
Time offset:

Collision Behavior

The result of simultaneous access to both ports is described below:

Read-Read Collisions

If both ports read simultaneously from the same memory cell, the read operation is successful.

Write-Write Collisions

If both ports try to write simultaneously to the same memory cell, both outputs are marked as invalid (nan).

Write-Read Collisions

This collision occurs when one port writes and the other reads from the same memory cell. While the memory contents are not corrupted, the validity of the output data on the read port depends on the Write Mode of the write port.

- If the write port is in **Read before write** mode, the other port can reliably read the old memory contents.
- If the write port is in **Read after write** or **No read on write**, data on the output of the read port is invalid (nan).

You can set the Write Mode of each port using the Advanced tab of the block parameters dialog box.

Block Parameters

The block parameters dialog box can be invoked by double-clicking the icon in your Simulink model.

Basic tab

Parameters specific to the Basic tab are:

- **Depth**: specifies the number of words in the memory for Port A, which must be a positive integer. The Port B depth is inferred from the form factor specified by the input data widths.
- **Initial value vector**: for distributed memory or block RAM, specifies the initial memory contents. The size and precision of the elements of the initial value vector are based on the data format specified for Port A. When the vector is longer than the RAM, the vector's trailing elements are discarded. When the RAM is longer than the vector, the RAM's trailing words are set to zero. The initial value vector is saturated and rounded according to the precision specified on the data port A of RAM.

  Note: UltraRAM memory is initialized to all 0's during power up or device reset. If implemented in UltraRAM, the Single Port RAM block cannot be initialized to user defined values.

- **Memory Type**: option to select whether the dual port RAM will be implemented in **Distributed memory**, **Block RAM**, or **UltraRAM**. The distributed dual port RAM is always set to use port A in Read Before Write mode and port B in read-only mode.

  Depending on your selection for **Memory Type**, the dual-port RAM will be inferred or implemented in this way when the design is compiled:

  - If the block will be implemented in **Distributed memory**, the Distributed Memory Generator v8.0 LogiCORE IP will be inferred or implemented when the design is compiled. This LogiCORE IP is described in the *Distributed Memory Generator v8.0 Product Guide* (PG063).
  - If the block will be implemented in **Block RAM**, the Block Memory Generator v8.3 LogiCORE IP will be inferred or implemented when the design is compiled. This LogiCORE IP is described in the *Block Memory Generator v8.3 Product Guide* (PG058).
  - If the block will be implemented in **UltraRAM**, the XPM_MEMORY_TDPRAM

(True Dual Port RAM) macro will be inferred or implemented when the design is compiled. For information on the XPM_MEMORY_TDPRAM Xilinx Parameterized Macro (XPM), see this link in the *UltraScale Architecture Libraries Guide* (UG974).

- **Initial value for port A output Register**: specifies the initial value for port A output register. The initial value is saturated and rounded according to the precision specified on the data port A of RAM.
- **Initial value for port B output register**: specifies the initial value for port B output register. The initial value is saturated and rounded according to the precision specified on the data port B of RAM.
- **Provide synchronous reset port for port A output register**: when selected, allows access to the reset port available on the port A output register of the Block RAM or UltraRAM. The reset port is available only when the latency of the Block RAM or UltraRAM is greater than or equal to 1.
- **Provide synchronous reset port for port B output register**: when selected, allows access to the reset port available on the port B output register of the Block RAM or UltraRAM. The reset port is available only when the latency of the Block RAM or UltraRAM is greater than or equal to 1.
- **Provide enable port for port A**: when selected, allows access to the enable port for port A. The enable port is available only when the latency of the block is greater than or equal to 1.
- **Provide enable port for port B**: when selected, allows access to the enable port for port B. The enable port is available only when the latency of the block is greater than or equal to 1.

Advanced tab

Parameters specific to the Advanced tab are:

Write Modes

- **Port A** or **Port B**: when the Dual Port RAM block is implemented in block RAM, specifies memory behavior for port A or port B when WE is asserted. Supported modes are: **Read after write**, **Read before write**, and **No read On write**. **Read after write** indicates the output value reflects the state of the memory after the write operation. **Read before write** indicates the output value reflects the state of the memory before the write operation. **No read on write** indicates that the output value remains unchanged irrespective of change of address or state of the memory. There are device specific restrictions on the applicability of these modes. Also refer to the Write Mode topic above for more information.

Other parameters used by this block are explained in the topic Common Options in Block Parameter Dialog Boxes.

LogiCORE™ and XPM Documentation

LogiCORE IP Block Memory Generator v8.3 (Block RAM)

LogiCORE IP Distributed Memory Generator v8.0 (Distributed Memory)

UltraScale Architecture Libraries Guide - XPM_MEMORY_TDPRAM Macro (UltraRAM)