LogiCORE™

Distributed Memory V1.0.3

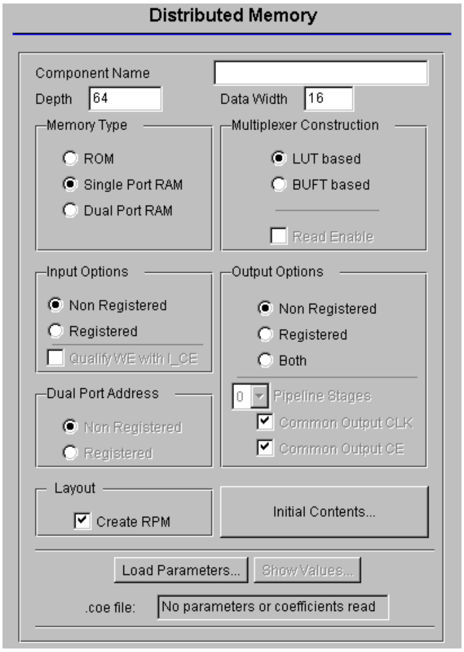
December 17, 1999 Product Specification

XILINX®

Xilinx Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
URL: www.xilinx.com/support/techsup/appinfo
www.xilinx.com/ipcenter

Features

- Drop-in module for Virtex, Virtex™-E and Spartan™–II FPGAs
- Generates ROMs, single/dual-port RAMs
- Supports data widths ranging from 1 to 64 bits wide
- Supports depths ranging from 16 to 256 words



Distributed Memory

Component Name [ ]

Depth 64

Data Width 16

Memory Type: ○ ROM ● Single Port RAM ○ Dual Port RAM

Multiplexer Construction: ● LUT based ○ BUFT based □ Read Enable

Input Options: ● Non Registered ○ Registered □ Qualify WE with I_CE

Output Options: ● Non Registered ○ Registered ○ Both

0 Pipeline Stages

☑ Common Output CLK

☑ Common Output CE

Dual Port Address: ● Non Registered ○ Registered

Layout: ☑ Create RPM

.coe file: No parameters or coefficients read

Figure 1: Main Distributed Memory Parameterization Screen

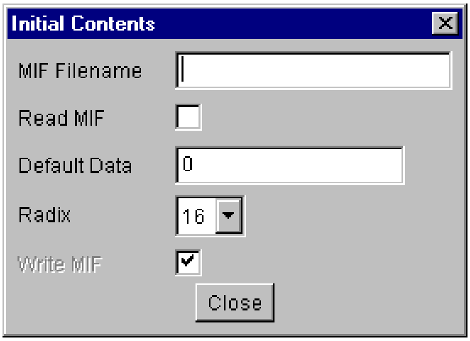
- Optional registered output and pipelining
- Relationally Placed Macro (RPM) mapping and placement technology
- Incorporates Xilinx Smart-IP technology for maximum performance
- To be used with version 2.1i and later of the Xilinx CORE Generator System

Functional Description

The distributed memory module is used to create memory structures using the Select-RAM. It can be used to create Read Only Memory (ROM), single port Random Access Memory (RAM) and pseudo dual port RAM. Data widths of up to 64 bits are supported with depths of up to 256 words. Options are available for simple registering of inputs and outputs in addition to pipelining capabilities. The module can optionally be generated as a Relationally Placed Macro (RPM) or as unplaced logic.

Pinout

Signal names are shown in Figures 3, 4, 5 and 6 and described in Table 1.



Initial Contents

MIF Filename [ ]

Read MIF □

Default Data 0

Radix 16

Write MIF ☑

Figure 2: Distributed Memory Initial Contents Screen

Table 1: Core Signal Pinout

Signal	Signal Direction	Description
D[P:0]	Output	Non-registered dual port output bus. One of two non-registered output buses on dual-port RAMs.
A[N:0]	Input	Address inputs. Only address input for ROMs and single port RAMs. On dual port memories it defines memory location written to and memory location read out on the SPO{N:0] outputs.
DPRA[N:0]	Input	Dual Port Read Address. Port is only present on Dual port RAMs and defines memory location read out on the SPO{N:0] outputs.
SPO[P:0]	Output	Non-registered single port output bus. Non-registered data output bus for ROMs and single-port RAMs. One of two non-registered output buses on dual-port RAMs.
QSPO[P:0]	Output	Registered single port output bus. Registered data output bus for ROMs, single-port RAMs. One of two non-registered output buses on dual-port RAMs.
DPO[P:0]	Output	Non-registered dual port output bus - one of the non-registered data output bus for dual-port RAMs. Data stored at the address location specified by DPRA[N:0] appears at this port.
QDPO[P:0]	Output	Registered dual port output bus. One of two registered output buses on dual-port RAMs.
CLK	Input	Write clock and register clock for ROMs and single-port RAMs. On dual-port RAMs signal is the write clock and register clock for single port input and output registers.
QDPO_CLK	Input	On dual-port RAMs signal is the write clock and register clock for dual port RAM input and output registers
WE	Input	Write Enable
I_CE	Input	Input Clock Enable. Signal is present for RAMs which have registered inputs. The clock enable controls input data register, address register and WE register.
QSPO_CE	Input	On ROMs clock enable controls all input and output registers. On dual-port memories controls output register and pipeline registers in QSPO path.
QDPO_CE	Input	Only present on dual-port RAMs. Controls output register and pipeline registers in QDPO path.

Note:

1. All control inputs are Active High. If an Active Low input is required for a particular control pin an inverter must be placed in the path to the pin. The inverter will be absorbed appropriately during mapping.

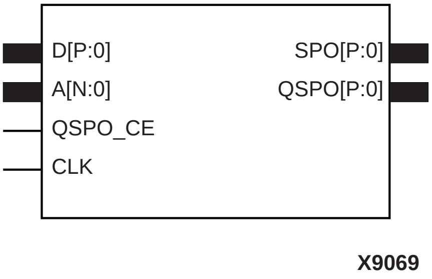


Figure 3: Core Schematic Symbol for ROMs

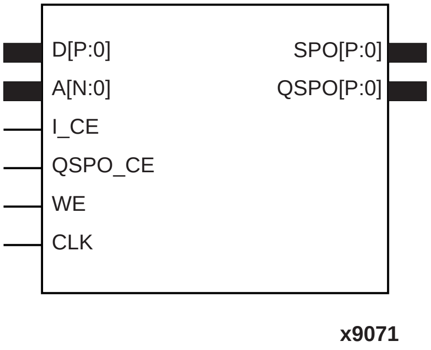


Figure 4: Core Schematic Symbol for single-port RAMs

CORE Generator Parameters

The main CORE Generator parameterization screen for this module is shown in Figure 1. The parameters are as follows:

- **Component Name**: The component name is used as the base name of the output files generated for this module. Names must begin with a letter and must be composed from the following characters: a to z, 0 to 9 and "_" .
- **Depth**: Enter the required memory depth. The valid range is 16 to 256 in steps of 16. If an illegal value less than 256 is entered it will be rounded up to the nearest legal value. Values above 256 will be rounded down to 256. The default value is 64.
- **Data Width**: Enter the width of the memory. The valid range is 1 to 64. If a value is entered that is illegal it will be rounded to the nearest legal value. The default value is 16.
- **Memory Type**: Select the appropriate radio button for the types of memory required. The default setting is **Single Port RAM**.
  - **ROM**: A schematic diagram showing the structure of the ROM modules is shown in Figure 7. The Address register is optional (controlled by the setting of the **Input Options** parameter). Output registering and pipelining are also optional (controlled by the setting of the **Output Options** parameter).
  - **Single Port RAM**: A schematic diagram showing the structure of the single port RAM modules is shown in Figure 8. The address and data registers are optional (controlled by the setting of the I**nput Options** parameter). Output registering and pipelining are also optional (controlled by the setting of the **Output Options** parameter).
  - **Dual Port RAM**: A schematic diagram showing the structure of the dual port RAM modules is shown in Figure 9. The address and data registers are optional (controlled by the setting of the **Input Options** parameter). The dual port read address register is optional (controlled by the setting of the **Dual Port Address** parameter). Output registering and pipelining for both output ports are also optional (controlled by the setting of the **Output Options** parameter). When registered outputs are selected the two output ports can be clocked by the same or different clock signals and can have the same or different clock enables (based on the settings chosen for the **Common Output Clock** and **Common Output CE** parameters).

    Note: The Dual Port RAM implementations are based on using the Select RAM feature of the LUTs. In this mode the LUTs behave as a pseudo dual port memory and for details of the fundamental operation refer to the Virtex Product Specification.

    Note: The QSPO_CE and QDPO_CE registers are automatically generated when input and outputs are both registered in order to ensure correct phasing of the CE signal relative to the addresses.
  - **Multiplexer Construction**: For modules with depths greater than 16 (dual-port RAMs) or 32 (ROMS and single-port RAMS) multiplexing is required in the output paths. This can be performed using multiplexers built in Look Up Tables (LUTs) or using the tristate buffer primitives (BUFTs). If the multiplexer is constructed using BUFTs, the output of the module cannot be registered, although 1 pipeline stage is optionally available. The default setting is **LUT Based**.
  - **Input Options**: Select the appropriate radio button for the types of inputs required. The action of setting

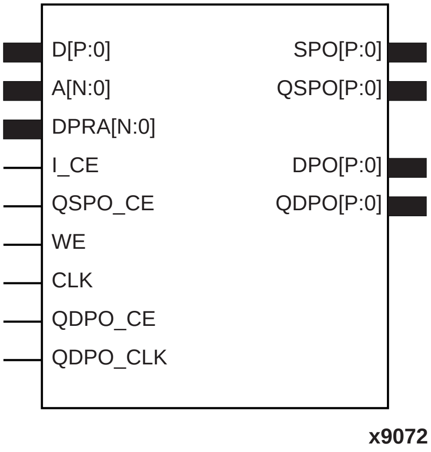


Figure 5: Core Schematic Symbol for dual-port RAMs

this parameter to **Registered** will have different effects depending on the **Memory Type** selection:

- **ROM**: An address register will be generated
- **Single Port RAM** and **Dual Port RAM**: A register on the A[N:0] Address input, a data input register and a WE register will be generated.

The default setting is **Non Registered**.

Note: The QSPO_CE and QDPO_CE registers are automatically generated when input and outputs are both registered in order to ensure correct phasing of the CE signal relative to the addresses.

- **Qualify WE with I_CE**: This parameter is only valid for single port RAM and dual port RAM with **Input Options** set to **Registered**. When the checkbox is not checked the WE register has no clock enable control. When checked the WE register has a clock enable that is driven by the I_CE input.
- **Dual Port Address**: This parameter is only valid for dual port RAMs. It controls the presence or absence of a register on the DPRA[N:0] inputs. The default setting is **Non Registered**.
- **Output Options**: Select the appropriate radio button for the types of outputs required. The default setting is **Non Registered**.
- **Pipeline Stages**: This parameter controls the amount of pipelining in the output side of the modules. Pipelining is only supported when the **Output Options** parameter is set to **Registered**. The amount of pipelining available depends on the **Memory Type** and the **Depth** of the memory. When 1 pipeline stage is selected the register is at the output of the module. A second pipeline stage adds registers at the outputs of the memory primitives. On modules that support a third pipeline stage the third register is added in the middle of the multiplexer. When a module is maximally pipelined its operating speed will be maximized. The amount of pipelining supported for LUT-based **Multiplexer Construction** depends on the **Memory Type** and is shown in Table 2. For BUFT-based Multiplexer Construction there can only be one stage of pipelining. In this case the registers are placed at the outputs of the memory primitives.
- **Common Output Clock**: This checkbox is only enabled for registered dual port RAMs. If not checked, the SPO registers will be enabled by QSPO_CEclocked by the CLK input and the DPO registers will be clocked from the QDPO_CLK input. The default is checked, where all output registers are be clocked from the CLK input.
- **Common Output CE**: This checkbox is only enabled for registered dual port RAMs and only if **Common Output Clock** is also checked. If **Common Output CE** is not checked, the SPO register clocks will be enabled by the QSPO_CE input and the DPO register clocks will be enabled from the QDPO_CE input. The default is checked, where all output register clocks are enabled by the QSPO_CE input.
- **Create RPM**: When this box is checked the module will be generated with relative location attributes attached. The resulting placement of the module will be in a column with two bits per slice. The default setting is to create an RPM.
- **Initial Contents ….**: This button displays the Initial Contents dialog shown in Figure 6. Defining the initial contents of the memory is optional. If this step is not carried out the **Default Data** value is used to initialize the memory. Initialization data is defined in a Memory Initialization File (MIF). This file is also used by the simulation model so that non-default initial values are modeled correctly. For a description of the MIF file refer to the section titled "Specifying Memory Contents using a MIF File".
  - **MIF Filename**: Specify the name of the MIF file to be used to define the initial values of the memory. The **MIF Filename** defaults to the **Component Name** value. If no extension is given a default extension of ".mif" will be used.
  - **Read MIF**: Check this box if the specified MIF file exists and the initial memory contents are to be read from that file.
  - **Default Data**: Enter the initial value to be stored in any memory location not specified by another means. When no value is entered this field defaults to 0. Values may be entered in Binary, Decimal or Hex format, as defined by the **Radix** entry.
  - **Radix**: Choose the radix of the **Default Data** value. Valid entries are 2,10 and 16.
  - **Write MIF**: This check box is not user modifiable. A MIF file will always be written when the module is generated.
  - **Close**: This button closes the dialog box and returns to the main parameterization dialog.
- **Load Parameters….**: In order to provide backwards compatibility with older memory cores the initial

Table 2: Amount of Pipelining Supported for Different Memory Types and Depths

Type of Memory	Number of Pipeline Stages supported		
	1 stage	2 stages	3 stages
ROM	≤ 64 words deep	> 64 words deep	
Single-port RAM	≤ 32 words deep	> 32 words deep	
Dual-port RAM	≤ 16 words deep	> 16 ≤ 128 words deep	> 128 words deep

memory contents can be loaded using the .COE format. The .COE file can contain other core parameters in addition to the memory initialization data.

> **Note**: This mechanismfor specifying intial values is not the preferred method. For new memory modules use a MIF file for memory initialization data and the .XCO file for all other core parameters.
>
> For a description of the .COE file refer to the section titled "Specifying Memory Contents using a .COE File".

- **Show Values**: Display the initial content values that have been read from the .coe file.
- **.COE file**: Displays the name of the .COE file that was read. Changing a parameter value that was previously loaded from a .COE file causes the name in this field to be highlighted in red as an indication that the settings have changed since the file was read.

Specifying Memory Contents using a MIF File

The initial contents of the memory can be defined using a text file known as a Memory Initialization File (MIF). MIF files may take any root file name and extension, although the extension will normally be ".mif".

The MIF file consists of one line of text per memory location, the first line in the file corresponding to address 0, the second line corresponding to address 1, and so on. The text on each line must be the initialization value (MSB first)

Table 3: XCO File Values and Default Values

Parameter	XCO File Values	Default GUI Setting
component_name	ASCII text starting with a letter and based upon the following character set: a .. z, 0..9 and _	blank
depth	Integer in the range 16 to 256 in steps of 16	64
data_width	Integer in the range 1 to 64	16
memory_type	One of the following keywords: rom, single_port_ram, dual_port_ram	single_port_ram
multiplexer_construction	One of the following keywords: lut_based or buft_based	lut_based
read_enable	One of the following keywords: true, false	false
input_options	One of the following keywords: true, false	non registered
qualify_we_with_i_ce	One of the following keywords: true, false	false
dual_port_address	One of the following keywords: non_registered, or registered	non registered
output_options	One of the following keywords: non_registered, registered, both	non registered
pipeline_stages	One of the following keywords: true, false	0
common_output_clk	One of the following keywords: true, false	true
common_output_ce	One of the following keywords: true, false	true
create_rpm	One of the following keywords: true, false	true
mif_filename	ASCII text starting with a letter and based upon the following character set: a .. z, 0..9 and _	blank
read_mif	One of the following keywords: true, false	false
default_data	Numeric value in the Radix specified by the radix keyword whose value does not exceed $2^{DATA \backslash WIDTH - 1}$	0
radix	One of the following keywords: 2, 10, 16	16
write_mif	The value must be set to true	true

for the corresponding memory address in binary format, with exactly one binary digit per bit of the memory's width.

The file must not contain more lines than there are memory locations, but may contain fewer. In the latter case all undefined memory locations will be initialized with the **Default Data** value.

Specifying Memory Contents using a .COE File

This mechanism is not preferred and is only provided for backwards compatibility with existing .COE files. The preferred mechanism for new memory modules is to use a MIF file for memory initialization data and the .XCO file for all other core parameters.

In addition to the initial memory contents, a .coe file can define all the parameters visible on the parameterization window. A .COE file may take any root file name but must end with the extension ".coe".

For a detailed description of the .COE file syntax, please refer to the *Xilinx CORE Generator 2.1i User Guide* on the Xilinx Alliance 2.1i CD.

Parameter Values in the XCO File

Names of XCO file parameters and their parameter values are identical to the names and values shown in the GUI, except that underscore characters (_) are used instead of spaces. The text in an XCO file is case insensitive.

Table 3 shows the XCO file parameters and values, as well as summarizing the GUI defaults. The following is an example of the CSET parameters in an XCO file:

```
CSET component_name = abc123
CSET depth = 64
CSET data_width = 16
CSET memory_type = Single_Port_RAM
CSET multiplexer_construction = LUT_based
CSET read_enable = FALSE
CSET input_options = Non_Registered
CSET qualify_we_with_i_ce = FALSE
CSET dual_port_address = Non_Registered
CSET output_options = Non_Registered
CSET pipeline_stages = 0
CSET common_output_clk = TRUE
CSET common_output_ce = TRUE
CSET create_rpm = TRUE
CSET mif_filename = abc123
CSET read_mif = FALSE
CSET default_data = 0
CSET radix = 16
CSET write_mif = TRUE
```

Core Resource Utilization

The resource utilization figures for this core are shown in Table 4. The resource counts shown are for non-registered usage and should be multiplied by the **Data Width** of the module for a total resource count. When output registering or any style of pipelining is requested the registers used are in slices already used for the memory primitives or LUTs that are used to construct the multiplexers, so no additional slices are required.

When input registering is requested extra registers are required, one flip flop per control bit (WE, QSPO_CE and QDPO_CE) and one flip flop per bit of data and address (D[P:0], A[N:0] and DPRA[N:0]).

If both input registering and output registering are requested, extra registers are required to implement the A Pipe Register and (for dual port RAMs only) the DPRA Pipe Register. This requires one additional flip-flop per bit of A[N:0] and (on dual port RAMs only) DPRA[N:0].

Ordering Information

This core is downloadable free of charge from the Xilinx IP Center (www.xilinx.com/ipcenter), for use with the Xilinx Core Generator System version 2.1i and later. The Core Generator System 2.1i tool is bundled with the Alliance 2.1i and Foundation 2.1i implementation tools.

To order Xilinx software contact your local Xilinx sales representative at www.xilinx.com/company/sales.htm.

Table 4: Resource Utilization (LUT/MUXF5/MUXF6)

Memory Type	Depth	Resources Used
ROM	≤ 16	1 / 0 / 0
	17 - 32	2 / 1 / 0
	33 - 48	3 / 2 / 1
	49 - 64	4 / 2 / 1
	65 - 80	6 / 2 / 1
	81 - 96	7 / 3 / 1
	97 - 112	8 / 4 / 2
	113 - 128	9 / 4 / 2
	129 - 144	11 / 5 / 2
	145 - 160	12 / 6 / 2
	161 - 176	13 / 7 / 3
	179 - 192	14 / 7 / 3
	195 - 208	15 / 7 / 3
	211 - 224	16 / 8 / 3
	227 - 240	17 / 9 / 3
	240 - 256	18 / 9 / 3

Memory Type	Depth	Resources Used
Single Port RAM	≤ 16	1 / 0 / 0
	17 - 32	2 / 0 / 0
	33 - 48	3 / 2 / 1
	49 - 64	4 / 2 / 1
	65 - 80	6 / 2 / 1
	81 - 96	7 / 3 / 1
	97 - 112	8 / 4 / 2
	113 - 128	9 / 4 / 2
	129 - 144	11 / 5 / 2
	145 - 160	12 / 6 / 2
	161 - 176	13 / 7 / 3
	179 - 192	14 / 7 / 3
	195 - 208	15 / 7 / 3
	211 - 224	16 / 8 / 3
	227 - 240	17 / 9 / 3
	240 - 256	18 / 9 / 3

Memory Type	Depth	Resources Used
Dual Port RAM	≤ 16	2 / 0 / 0
	17 - 32	6 / 0 / 0
	33 - 48	10 / 2 / 0
	49 - 64	12 / 2 / 0
	65 - 80	18 / 4 / 2
	81 - 96	20 / 4 / 2
	97 - 112	22 / 4 / 2
	113 - 128	24 / 4 / 2
	129 - 144	27 / 4 / 2
	145 - 160	31 / 4 / 2
	161 - 176	35 / 6 / 2
	179 - 192	37 / 6 / 2
	195 - 208	43 / 8 / 4
	211 - 224	45 / 8 / 4
	227 - 240	47 / 8 / 4
	240 - 256	49 / 8 / 4

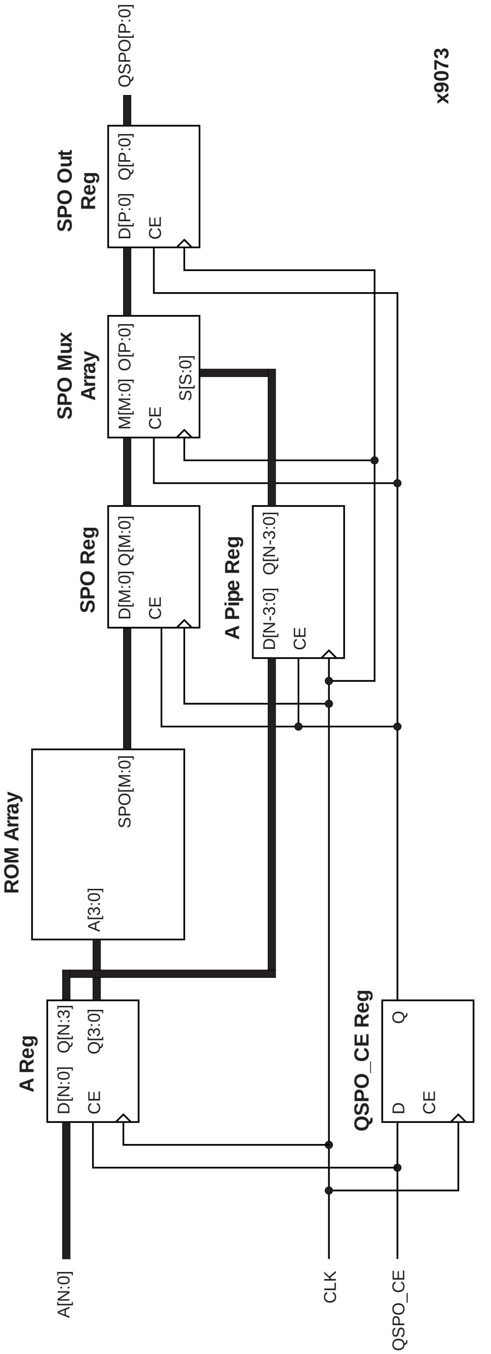


Figure 7: Schematic of a ROM Module

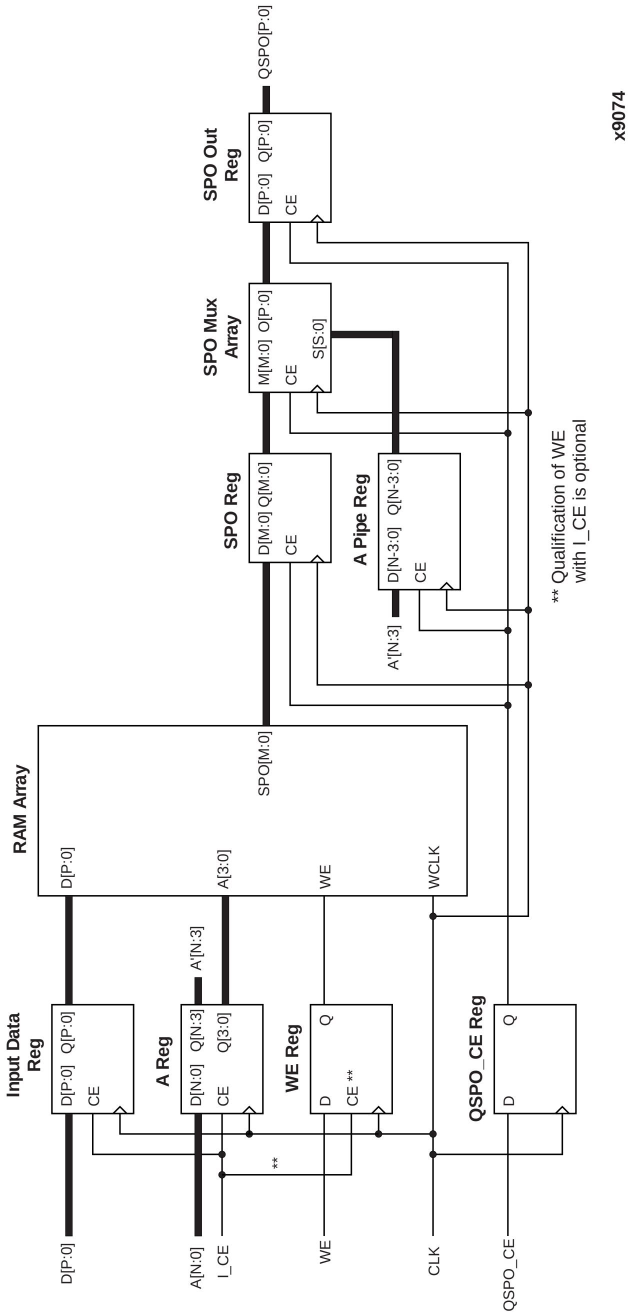


Figure 8: Schematic of a Single Port RAM Module

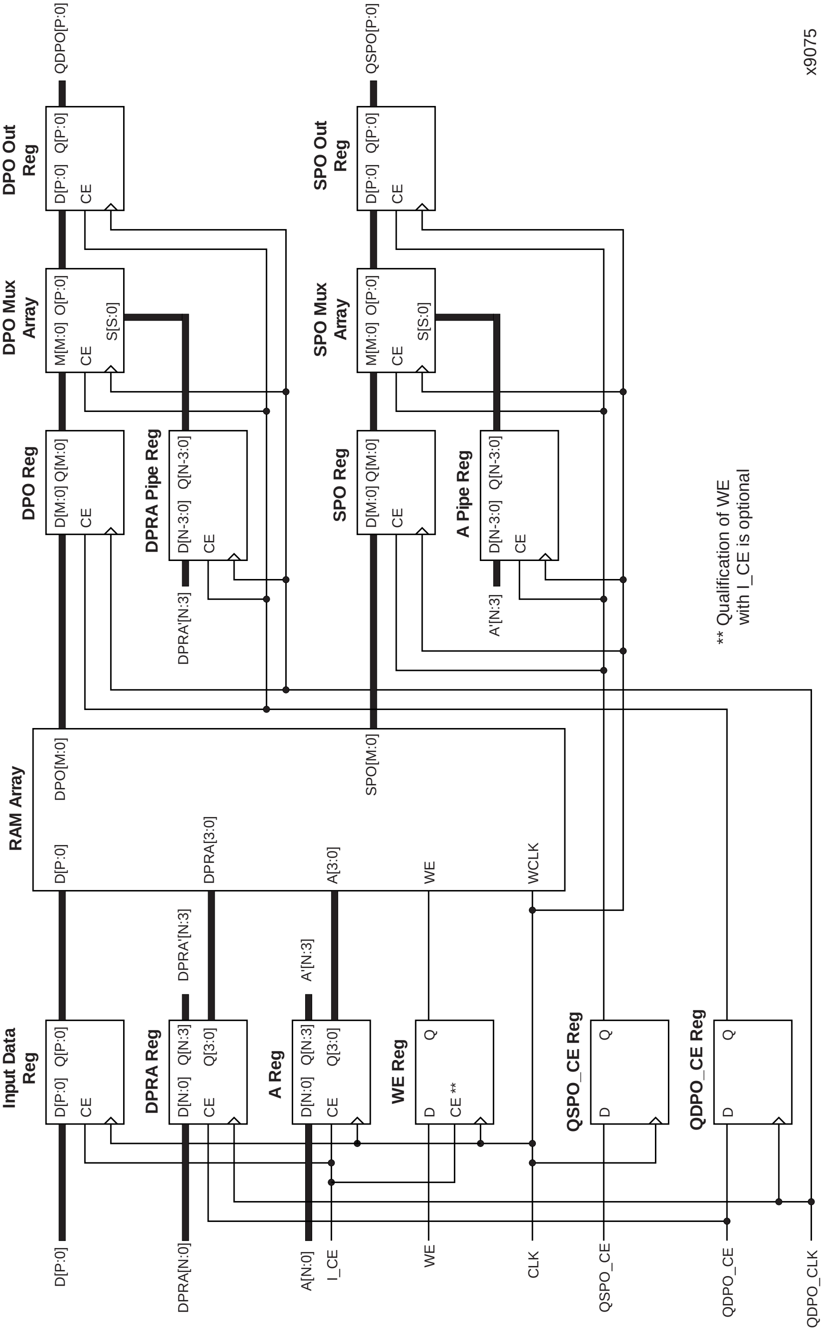


Figure 9: Schematic of a Dual Port RAM Module