RTG4 Dual-Port Large SRAM Configuration User Guide

Introduction (Ask a Question)

A Dual-Port Large SRAM enables read and write access on both ports: Port A and Port B.

The core configurator automatically cascades Large SRAM blocks to create wider and deeper memories by choosing the most efficient aspect ratio. It also handles the grounding of unused bits. The core configurator supports the generation of memories that have different aspect ratios on each port.

A Dual-Port Large SRAM functions synchronously for memory write and read operations, setting up the addresses as well as writing and reading the data. Memory write and read operations are triggered at the rising edge of the clock. The address, data, block-port select, write-enable, and read-enable inputs are registered.

Optional pipeline registers are available at both the read data ports to improve the clock-to-out delay.

When ECC is enabled, output flags are generated to indicate single-bit-correct and double-bit-detect for each port.

This user guide describes how to configure a Dual-Port Large SRAM instance and how to define the way in which the signals are connected. For more details about the Dual-Port Large SRAM, see the RTG4 User's Guide.

Figure 1. Dual-Port Large SRAM Configurator

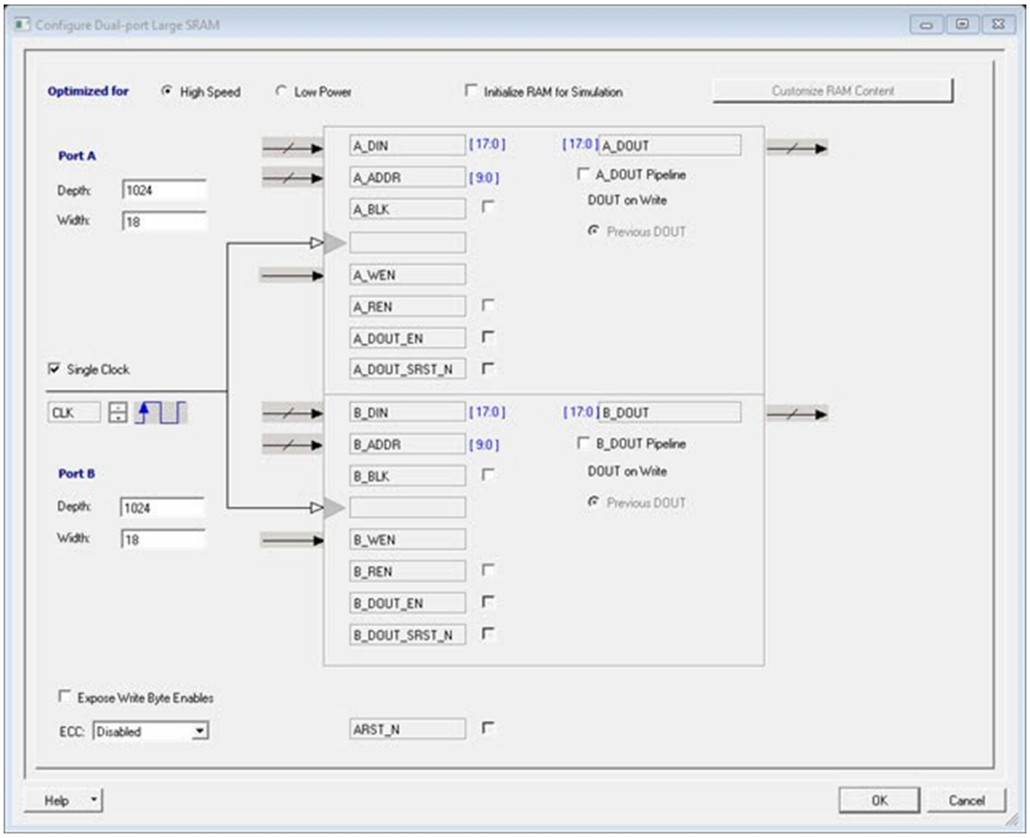


Table of Contents

1. Functionality (Ask a Question)

1.1 Optimized for High Speed and Low Power (Ask a Question)

Selecting High Speed results in a macro optimized for speed and area (width cascading).

Selecting Low Power results in a macro optimized for low power, but uses additional logic at the input and output (depth cascading). Performance for a low-power optimized macro may be inferior to a macro optimized for speed.

1.2 Port A Depth/Width and Port B Depth/Width (Ask a Question)

The depth range for each port is 1-32768. The width range for each port is 1-3762.

The two ports can be independently configured for any depth and width. (Port A Depth * Port A Width) must equal (Port B Depth * Port B Width).

1.3 Single Clock (CLK) or Independent Port A and B Clocks (A_CLK, B_CLK) (Ask a Question)

The default configuration for Dual-Port Large SRAM is a Single clock (CLK) to drive both A and B ports with the same clock. Uncheck the **Single clock** check box to drive independent clocks — one for each port (A_CLK and B_CLK).

Click the waveform next to any of the clock signals to toggle its active edge.

1.4 Block Select (A_BLK, B_BLK) and Read/Write Control (A_WEN, B_WEN) (Ask a Question)

De-asserting A_BLK forces A_DOUT to zero. De-asserting B_BLK forces B_DOUT to zero. Asserting A_BLK when A_WEN is low reads the RAM at the address A_ADDR onto the input of the A_DOUT register, on the next rising edge of A_CLK.

Asserting A_BLK when A_WEN is high writes the data A_DIN into the RAM at the address A_ADDR, on the next rising edge of A_CLK.

Asserting B_BLK when B_WEN is low, reads the RAM at the address B_ADDR onto the input of the B_DOUT register, on the next rising edge of B_CLK.

Asserting B_BLK when B_WEN is high, writes the data B_DIN into the RAM at the address B_ADDR, on the next edge of B_CLK.

The default configuration for A_BLK and B_BLK is unchecked, which ties the signal to the active state and removes it from the generated macro. Click the respective check box to insert that signal on the generated macro. Click the signal arrow (when available) to toggle its polarity.

1.5 Pipeline for Read Data Output of Port A and B (Ask a Question)

Click the **Pipeline** check box to enable pipelining of Read data (A_DOUT or B_DOUT).

Turning off pipelining of Read data of a port also disables the configuration options of the respective DOUT_EN and DOUT_SRST_N signals.

1.6 Register Enable (A_DOUT_EN and B_DOUT_EN) (Ask a Question)

The pipeline registers for ports A and B have active high, enable inputs. The default configuration is to tie these signals to the active state and remove them from the generated macro. Click each signal's check box to insert that signal on the generated macro.

Click the signal arrow (when available) to toggle its polarity.

1.7 Synchronous Reset (A_DOUT_SRST_N and B_DOUT_SRST_N) (Ask a Question)

The pipeline registers for ports A and B have active low, synchronous reset inputs. The default configuration is to tie these signals to the inactive state and remove them from the generated macro. Click each signal's check box to insert that signal on the generated macro.

Click the signal arrow (when available) to toggle its polarity.

1.8 Read Enable (A/B_REN) (Ask a Question)

De-asserting A_REN holds the previous Read data on port A and de-asserting B_REN holds the previous Read data on port B.

Asserting the A_REN reads the RAM at the A_ADDR onto port A's Read Data register on the next rising edge of the clock. Similarly, asserting the B_REN reads the RAM at the B_ADDR onto port B's Read Data register on the next rising edge of the clock.

The default configuration for the A_REN or B_REN option is unchecked, which ties the signal to the active state and removes it from the generated macro. Click the check box (when available) to insert that signal on the generated macro. Click the signal arrow (when available) to toggle its polarity.

The A_REN option for Port A and the B_REN option for Port B are disabled (greyed-out) in the Configurator if the Depth x Width of the port and the Optimization mode of the macro requires depth cascading and the address space is fractured.

Consider, for example, a 2048x18 Dual-Port Large SRAM macro without ECC. If it is optimized for High Speed, the Configurator generates two RAM1K18_RT blocks, each configured for 2048x9 (width-wise cascading). Because there is no depth-wise cascading, the Configurator enables the A_REN and B_REN options. When each of the check boxes is selected, the respective signal is exposed as an input port.

When the same 2048x18 Dual-Port Large SRAM macro is optimized for Low Power, the Configurator generates two RAM1K18_RT blocks, each configured for 1024x18 (depth-wise cascading). The Configurator disables the A_REN and B_REN options and these signals cannot be generated.

For a 4096xn Dual-Port SRAM macro, regardless of Low Power or High Speed optimization, depth-wise cascading is necessary, and it causes the address space to be fractured. The Configurator disables the A_REN and B_REN options, and these signals cannot be generated.

1.9 Expose Write Byte Enables (A/B_WBYTE_EN) (Ask a Question)

When enabled, write byte enables (A_WBYTE_EN and B_WBYTE_EN) are available as top-level buses. Each bit of A_WBYTE_EN gated by the A_WEN signal, enables writing to an individual byte of A_DIN. Each bit of B_WBYTE_EN gated by the B_WEN signal, enables writing to an individual byte of B_DIN.

1.10 Asynchronous Reset (ARST_N) (Ask a Question)

The pipeline registers for ports A and B have an active low, asynchronous reset input. The default configuration is to tie this signal to the inactive state and remove it from the generated macro. Click the check box to insert the asynchronous active low ARST_N signal on the generated macro.

Click the signal arrow (when available) to toggle its polarity.

1.11 Error Correction Code (ECC) (Ask a Question)

Three options are available for ECC:

- Disabled
- Pipelined
- Non-Pipelined

When ECC is disabled, each port could be configured to either 18 bits or 9 bits width.

When ECC is enabled (Pipelined or Non-Pipelined), both ports have word widths equal to 18 bits.

1.12 DOUT Registers Truth Table (Ask a Question)

The following table lists the functionality of the control signals on the A_DOUT and B_DOUT registers.

Table 1-1. A_DOUT and B_DOUT Registers Truth Table

ARST_N	_CLK	DOUT_EN	DOUT_SRST_N	D	Qn+1
0	X	X	X	X	0
1	Not rising	X	X	X	Qn
1	↑	0	X	X	Qn
1	↑	1	0	X	0
1	↑	1	1	D	D

2. Internal Connections of the Configurator (Ask a Question)

The following example shows a Dual-Port LSRAM configuration that generates a component with the following address widths.

- A_ADDR[*A_MSB:0*]
- B_ADDR[*B_MSB:0*]

Let *M* be the width of the address on the A-port of each LSRAM block and *N* be the width of the address on the B-port of each LSRAM block.

Let the decoder logic function be *decode*(addr*[j:k], i*), where 0 <= *i* < 2(j-k+1). Let *D* be the depth of an LSRAM block in the array of blocks, starting at 0.

LSRAM Block Port Depth x Width	M, N
2Kx9	11
1Kx18	10

2.1 A_BLK, A_REN Connections (Ask a Question)

The **A_BLK** signal on the generated component is connected to the A port block select input (**A_BLK**) for each LSRAM block according to the block depth within the component and synchronized with **A_CLK.** When ECC Pipeline is enabled, the component level **A_BLK** is OR'd with pipelined version of component **A_BLK**, while the component level **A_REN** is AND'd with component level **A_BLK**.

Table 2-1. A_BLK, A_REN Connections

Depth	A_BLK[2]	A_BLK[1]	A_BLK[0]	A_REN
A_MSB < *M*	A_BLK	1	1	A_REN
A_MSB = *M*	A_BLK	1	*decode*(A_ADDR[*M*:*M*], *D*%2)	1
A_MSB = *M*+1	A_BLK	*decode*(A_ADDR[*M*+1:*M*+1], (*D*/2)%2)	*decode*(A_ADDR[*M*:*M*],*D*%2)	1
A_MSB > *M*+1	A_BLK & *decode*(A_ADDR[*A_MSB*:*M*+2], *D*/ 4)	*decode*(A_ADDR[*M*+1:*M*+1], (*D*/2)%2)	*decode*(A_ADDR[*M*:*M*],*D*%2)	1

Note: A_REN is available on the top-level generated component only when there is no depth cascading (*A_MSB* < *M*). De-asserting **A_REN** holds the previous read-data. De-asserting **A_BLK** generates zeros on the read-data.

2.2 B_BLK, B_REN Connections (Ask a Question)

The **B_BLK** signal on the generated component is connected to the B port block select input (**B_BLK**) for each LSRAM block according to the block depth within the component and synchronized with **B_CLK**. When ECC Pipeline is enabled, the component level **B_BLK** is OR'd with pipelined version of component **B_BLK**, while the component level **B_REN** is AND'd with component level **B_BLK**.

Table 2-2. B_BLK, B_REN Connections

Depth	B_BLK[2]	B_BLK[1]	B_BLK[0]	B_REN
B_MSB < *N*	B_BLK	1	1	B_REN
B_MSB = *N*	B_BLK	1	*decode*(B_ADDR[*N*:*N*] , *D*%2)1	1
B_MSB = *N*+1	B_BLK	*decode*(B_ADDR[*N*+1:*N*+1], (*D*/2)%2)	*decode*(B_ADDR[*N*:*N*] , *D*%2)	1
B_MSB > *N*+1	B_BLK & *decode*(B_ADDR[*A_MSB*:*N*+2], *D*/ 4)	*decode*(B_ADDR[*N*+1:*N*+1], (*D*/2)%2)	*decode*(B_ADDR[*N*:*N*] , *D*%2)	1

Note: **B_REN** is available on the top-level generated component only when there is no depth cascading (*B_MSB* < *N*). De-asserting **B_REN** holds the previous read-data. De-asserting **B_BLK** generates zeros on the read-data.

2.3 A_DIN, B_DIN Connections (Ask a Question)

The **A_DIN** and **B_DIN** bits on the generated component is partitioned into slices based on the width of the data on the respective port of each LSRAM block. Each bit of **A_DIN** is connected to all blocks in a slice at every depth and synchronized with **A_CLK**. Each bit of **B_DIN** is connected to all blocks in a slice at every depth and synchronized with **B_CLK**.

2.4 A_DOUT Logic (Ask a Question)

The **A_DOUT** bits on the generated component is partitioned into slices based on the width of the data on the A port of each LSRAM block. Each bit of read-data on the A-port from all blocks in a slice at every depth is OR'd together to generate a bit of **A_DOUT**. The **A_DOUT** bits are synchronized with **A_CLK** according to the following latency.

Table 2-3. A_DOUT Logic

ECC Pipeline	ECC	A_DOUT Pipeline	A_DOUT Latency
No	No	No	0
No	No	Yes	1
No	Yes	No	0
No	Yes	Yes	1
Yes	Yes	No	1
Yes	Yes	Yes	2

2.5 B_DOUT Logic (Ask a Question)

The **B_DOUT** bits on the generated component is partitioned into slices based on the width of the data on the B port of each LSRAM block. Each bit of read-data on the B-port from all blocks in a slice at every depth is OR'd together to generate a bit of **B_DOUT**. The **B_DOUT** bits are synchronized with **B_CLK** according to the following latency.

ECC Pipeline	ECC	B_DOUT Pipeline	B_DOUT Latency
No	No	No	0
No	No	Yes	1
No	Yes	No	0
No	Yes	Yes	1
Yes	Yes	No	1
Yes	Yes	Yes	2

2.6 A_SB_CORRECT, A_DB_DETECT Logic (Ask a Question)

The **A_SB_CORRECT** and **A_DB_DETECT** outputs are synchronized with **A_CLK** according to the above **A_DOUT** latency. The **A_SB_CORRECT** flags of each LSRAM block are gated by its (**A_BLK** and **A_REN**) signals pipelined one more than the **A_DOUT** latency value and then OR'd together to generate the **A_SB_CORRECT** output of the component. Similarly, the **A_DB_DETECT** flags of each LSRAM block are gated by its (**A_BLK** and **A_REN**) signals pipelined one more than the **A_DOUT** latency value and then OR'd together to generate the **A_DB_DETECT** output of the component. As a result, both the outputs are zero whenever **A_BLK** or **A_REN** is de-asserted.

2.7 B_SB_CORRECT, B_DB_DETECT Logic (Ask a Question)

The **B_SB_CORRECT** and **B_DB_DETECT** outputs are synchronized with **B_CLK** according to the above **B_DOUT** latency. The **B_SB_CORRECT** flags of each LSRAM block are gated by its (**B_BLK** and **B_REN)** signals pipelined one more than the **B_DOUT** latency value and then OR'd together to generate the **B_SB_CORRECT** output of the component. Similarly, the **B_DB_DETECT** flags of each LSRAM block are

gated by its (**B_BLK** and **B_REN**) signals pipelined one more than the **B_DOUT** latency value and then OR'd together to generate the **B_DB_DETECT** output of the component. As a result, both the outputs are zero whenever **B_BLK** or **B_REN** is de-asserted.

3. Implementation Rules (Ask a Question)

3.1 Caveats for Dual-Port Large SRAM Generation (Ask a Question)

- The core configurator only supports depth cascading up to 32 blocks.
- The software returns a configuration error for unsupported configurations.
- All unused inputs must be grounded.
- ARST_N does not reset the memory contents. It resets only the pipeline registers for read data.
- Writing different data to the same address using both ports in Dual-Port Large SRAM is undefined and must be avoided.
- Writing to and reading from the same address is undefined and must be avoided. There is no collision prevention or detection. However, correct data is expected to be written into the memory.
- Reading from both ports at the same location is allowed.

4. RAM Content Manager (Ask a Question)

The RAM Content Manager allows you to specify the contents of your memory so that you can avoid the simulation cycles required for initializing the memory, which reduces simulation runtime.

The RAM core generator removes much of the complexity required in the generation of large memory that uses one or more RAM blocks on the device. The configurator uses one or more memory blocks to generate a RAM matching your configuration. It also creates the surrounding cascading logic.

The configurator cascades RAM blocks in three ways.

- Cascaded deep (for example, 2 blocks of 1024x18 to create a 2048x18)
- Cascaded wide (for example, 2 blocks of 1024x18 to create a 1024x36)
- Cascaded wide and deep (for example, 4 blocks of 1024x18 to create a 2048x36, in a 2 blocks width-wise by 2 blocks depth-wise configuration)

Specify memory content in terms of total memory size. The configurator partitions your memory file so that the right content goes to the right block RAM when multiple blocks are cascaded.

4.1 Supported Formats (Ask a Question)

The Microchip implementation of these formats interprets data sets in bytes. This means that if the memory width is 7 bits, every 8th bit in the data set is ignored. Or, if the data width is 9, two bytes are assigned to each memory address and the upper 7 bits of each 2-byte pair are ignored.

The following examples illustrate how the data is interpreted for various word sizes:

For the given data: FF 11 EE 22 DD 33 CC 44 BB 55 (where 55 is the MSB and FF is the LSB) For 32-bit word size:

```
0x22EE11FF (address 0)

0x44CC33DD (address 1)

0x000055BB (address 2)
```

For 16-bit word size:

```
0x11FF (address 0)

0x22EE (address 1)

0x33DD (address 2)

0x44CC (address 3)

0x55BB (address 4)
```

For 8-bit word size:

```
0xFF (address 0)

0x11 (address 1)

0xEE (address 2)

0x22 (address 3)

0xDD (address 4)

0x33 (address 5)

0xCC (address 6)

0x44 (address 7)
```

```
0xBB (address 8)

0x55 (address 9)
```

For 9-bit word size:

```
0x11FF -> 0x01FF (address 0) 0x22EE -> 0x00EE (address 1) 0x33DD -> 0x01DD
(address 2) 0x44CC -> 0x00CC (address 3) 0x55BB -> 0x01BB (address 4)
```

Note: For 9-bit, the upper 7-bits of the 2-bytes are ignored.

4.1.1 Intel-HEX (Ask a Question)

Intel-HEX is a standard format created by Intel. Files end with a HEX or IHX extension (for example, `file2.hex` or `file3.ihx`).

Memory contents are stored in ASCII files using hexadecimal characters. Each file contains a series of records (lines of text) delimited by new line, '\n', characters and each record starts with a ':' character. For more information about this format, see the Intel-Hex Record Format Specification document on the web (search Intel Hexadecimal Object File for several examples).

The Intel Hex record is composed of five fields arranged as follows:

```
:llaaaatt[dd...]cc
```

where:

- : is the start code of every Intel Hex record.
- ll is the byte count of the data field.
- aaaa is the 16-bit address of the beginning of the memory position for the data. Address is big Endian.
- tt is the record type that defines the data field:
  - 00 data record
  - 01 end of file record
  - 02 extended segment address record
  - 03 start segment address record (ignored by Microchip SoC tools)
  - 04 extended linear address record
  - 05 start linear address record (ignored by Microchip SoC tools)
- [dd...] is a sequence of n bytes of the data (n is equivalent to what was specified in the ll field).
- cc is a checksum of count, address, and data. The following is an example of an Intel Hex record: `:0300300002337A1E`

4.1.2 Motorola S-Record (Ask a Question)

Motorola S-Records use the file extension S (for example, `file4.s`).

Like Intel-HEX, Motorola S-records use ASCII files, hex characters, and records to specify memory content. For more information about this format, search the web for several examples of the Motorola S-record description document. The RAM Content Manager uses only the S1 through S3 record types and ignores other record types.

The key difference between Intel-HEX and Motorola S-record types is the record format, along with the extra error-checking features incorporated into Motorola S.

In both formats, memory content is specified by providing a starting address and a data set. The upper bits of the data set are loaded into the starting address and leftovers overflow into the adjacent addresses until the entire data set has been used.

The Motorola S-record is composed of six fields and arranged as follows:

```
Stllaaaa[dd...]cc
```

where:

- S is the start code of every Motorola S-record.
- t is the record type that defines the data field.
- ll is the byte count of the data field.
- aaaa is a 16-bit address of the beginning of the memory position for the data. Address is big Endian.
- [dd...] is a sequence of n bytes of the data; n is equivalent to what was specified in the ll field.
- cc is the checksum of count, address, and data. The following is an example of a Motorola S-record:
  `S10a0000112233445566778899FFFA`

4.2 Opening RAM Content Manager (Ask a Question)

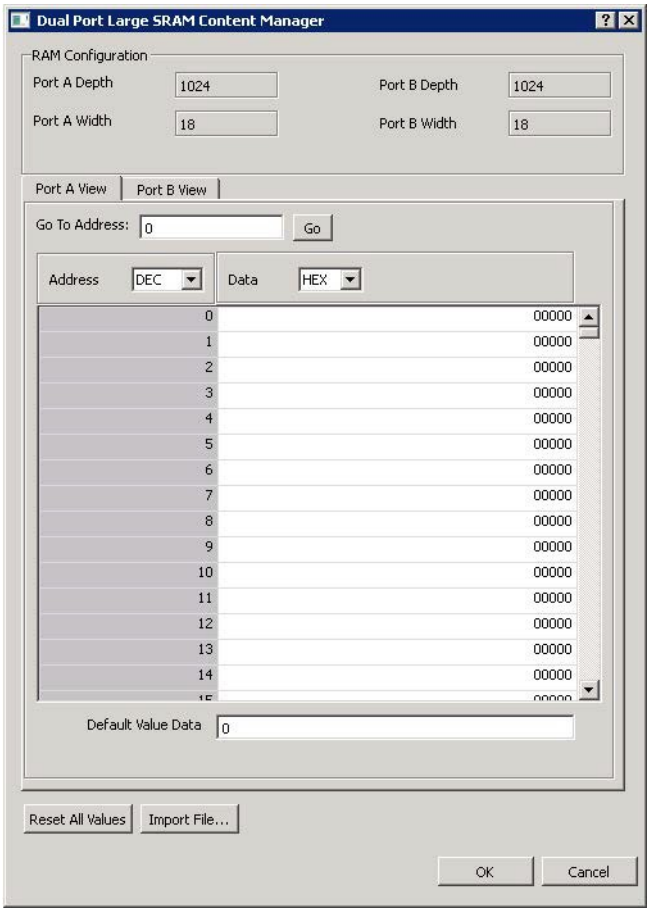
To open the RAM Content Manager after specifying your RAM (Read and Write Depth and Width) configurations, select the **Initialize RAM for Simulation** check box, and click **Customize RAM Content**.

Important:

1. Clearing and selecting the **Initialize RAM for Simulation** check box in the same dialog box does not discard the memory file data in the generated files. To discard the memory file data, clear the **Initialize RAM for Simulation** check box and click **OK**.
2. After selecting the **Initialize RAM for Simulation** check box, use the **Customize RAM Content** option to initialize memory content to zeros.

The RAM Content Manager appears as follows.

Figure 4-1. Customize RAM Content for Simulation



4.2.1 RAM Configuration (Ask a Question)

Write Depth and Write Width — Read-only value specified in the RAM core generator dialog box.

Read Depth and Read Width — Read-only value specified in the RAM core generator dialog box.

4.2.2 Write Port View/Read Port View (Ask a Question)

Go To Address — Allows you to go to a specific address in RAM Content Manager. Each memory block has many addresses, making it difficult to scroll through and find a specific one. This task is simplified by allowing you to type a specific address. The number display format (Hex, Bin, Dec) is controlled by the value you set in the drop-down menu above the Address column.

Address — Lists the address of a memory location. The drop-down menu specifies the number format for your address list (hexadecimal, binary, or decimal).

Data — Allows you to control the data format and data value in RAM Content Manager. Click the value to change it.
Note: Dialog boxes show all data with the MSB down to LSB. For example, if the row showed 0xAABB for a 16-bit word size, the AA would be the MSB and the BB would be LSB.

Default Data Value — Value given to memory addresses that have not been explicitly initialized (by importing content or editing manually). When changed, all default values in RAM Content Manager are updated to match the new value. The number display format (Hex, Bin, Dec) is controlled by the value you set in the drop-down menu above the Data column.

Reset All Values — Resets the data values.

Import File — Opens the Import Memory Content dialog box for you to select a memory content file (Intel-HEX) to load. File extensions are set to *.hex for Intel-Hex files during import.

OK — Closes RAM Content Manager and saves all the changes made to the memory and its contents.

Cancel — Closes RAM Content Manager, cancels all your changes in this instance of the manager, and returns the memory to the state it held before RAM Content Manager was opened.

4.3 MEMFILE (RAM Content Manager Output File) (Ask a Question)

To transfer RAM data from the RAM Content Manager to test equipment, use MEM files. RAM contents are first organized into the logical layer, and then reorganized to fit the hardware layer. Then the contents are stored in MEM files that are read by other systems and used for testing.

The MEM files are named according to the logical structure of RAM elements created by the configurator. Using this method, the highest order RAM blocks are named `CORE_R0C0.mem`, where “R” stands for row and “C” stands for column. For multiple RAM blocks, the naming continues with CORE_R0C1, CORE_R0C2, CORE_R1C0, and so on.

Data intended for RAM is stored as ASCII 1s and 0s within the file. Each memory address occupies one line. Words from logical layer blocks are concatenated or split to make them fit efficiently within the hardware blocks. If the logical layer width is less than the hardware layer, two or more logical layer words are concatenated to form one hardware layer word. In this case, the lowest bits of the hardware word comprise the lower address data bits from the logical layer. If the logical layer width exceeds the hardware layer, the words are split, with the lower bits placed in lower addresses.

If the logical layer words do not fit cleanly into the hardware layer words, the most significant bit of the hardware layer words is not used and defaults to zero. This is also done when the logical layer width is 1 to avoid left over memory at the end of the hardware block.

5. Port Description (Ask a Question)

The following table lists the Dual-Port Large SRAM signals in the generated macro.

Table 5-1. Dual-Port Large SRAM Signals

Port	Direction	Polarity	Description
CLK	In	Rising edge	Single clock signal that drives all three ports with the same clock.
A_DIN[]	In		Port A Write data.
A_ADDR[]	In		Port A Read address.
A_BLK	In	Active high	Port A Enable.
A_CLK	In	Rising edge	Port A clock.
A_WEN	In		Port A signal to switch between Read and Write modes: • Low = Read • High = Write
A_REN	In		Port A Read Data Enable (exposed only if there is no depth cascading).
A_WBYTE_EN[]	In	Active High	Port A Write Byte Enables (per byte).
A_DOUT[]	Out		Port A Read data.
A_DOUT_EN	In	Active High	Port A Read data register Enable.
A_DOUT_SRST_N	In	Active Low	Port A Read data register Synchronous reset.
A_SB_CORRECT	Out	Active High	Port A single-bit correct flag.
A_DB_DETECT	Out	Active High	Port A double-bit detect flag.
B_DIN[]	In	—	Port B Write data.
B_ADDR[]	In	—	Port B address.
B_BLK	In	Active High	Port B Enable.
B_CLK	In	Rising edge	Port B clock.
B_WEN	In	—	Port signal to switch between Read and Write modes: • Low = Read • High = Write
B_REN	In	—	Port B Read Data Enable (exposed only if there is no depth cascading)
B_WBYTE_EN[]	In	Active High	Port B Write Byte Enables (per byte)
B_DOUT[]	Out	—	Port B Read data
B_DOUT_EN	In	Active High	Port B Read data register Enable.
B_DOUT_SRST_N	In	Active Low	Port B Read data register Synchronous reset.
B_SB_CORRECT	Out	Active High	Port B single-bit correct flag.
B_DB_DETECT	Out	Active High	Port B double-bit detect flag.
ARST_N	In	Active Low	Port A and B Read data register Asynchronous reset.

6. Parameters (Ask a Question)

The following table lists the Dual-Port Large SRAM parameters in the generated macro.

Table 6-1. Dual-Port Large SRAM Parameters

Parameter	Valid Range	Default (Condition)	Description
LPMTYPE	LPM_RAM	LPM_RAM	Macro category
PTYPE	1,2	2	1: Two-port. 2: Dual-port.
CASCADE	0, 1	0	0: Cascading for WIDTH or Speed. 1: Cascading for DEPTH or Power.
CLKS	1, 2	1	1: Single Read/Write Clock. 2: Independent Read and Write Clocks.
CLK_EDGE	RISE, FALL	RISE (CLKS=1)	RISE: Rising edge single clock. FALL: Falling edge single clock.
CLOCK_PN	CLK CLK_N	CLK (CLKS=1)	Single clock port name.
A_CLK_EDGE	RISE, FALL	RISE (CLKS=2)	RISE: Rising edge Port A clock. FALL: Falling edge Port A clock.
CLKA_PN	A_CLK A_CLK_N	A_CLK CLKS=2	Port A clock port name.
B_CLK_EDGE	RISE, FALL	RISE (Must have CLKS=2)	RISE: Rising edge Port B clock. FALL: Falling edge Port B clock.
CLKB_PN	B_CLK B_CLK_N	B_CLK (CLKS=2)	Port B clock port name.
A_BLK_POLARITY	0, 1, 2	2	0: Active low A_BLK_N port will be exposed to exercise port A to be active or not. 1: Active high A_BLK port will be exposed to exercise Port A to be active or not. 2: Port A tied off to be always active.
BLKA_PN	A_BLK, A_BLK_N	A_BLK	Port A enable port name.
B_BLK_POLARITY	0,1, 2	2	0: Active low B_BLK_N port will be exposed to exercise port B to be active or not 1: Active high B_BLK port will be exposed to exercise Port B to be active or not . 2: Port B tied off to be always active.
BLKB_PN	B_BLK, B_BLK_N	B_BLK	Port B enable port name.
BYTEENABLES	0, 1	0	0: Do not generate A_WBYTE_EN or B_WBYTE_EN. 1: Generate A_WBYTE_EN and B_WBYTE_EN.
A_BYTE_ENABLE_WIDTH	—	0	Byte Enable width for port A. Values are based on the configuration.
B_BYTE_ENABLE_WIDTH	—	0	Byte Enable width for port B Values are based on the configuration.
ADDRESSA_PN	—	A_ADDR	Port A address port name.

.....continued

Parameter	Valid Range	Default (Condition)	Description
ADDRESSB_PN	—	B_ADDR	Port B address port name.
A_DEPTH	1-65536	1024	Port A address depth.
B_DEPTH	1- 65536	1024	Port B address depth.
A_WIDTH	1-3762	18	Port A data width.
B_WIDTH	1-3762	18	Port B data width.
RWA_PN	—	A_WEN	Port A Write enable port name.
RWB_PN	—	B_WEN	Port B Write enable port name.
A_PMODE	0, 1	0	0: Bypass Port A read data register. 1: Pipeline Port A read data.
A_DOUT_EN_POLARITY	0, 1, 2	2 (A_PMODE=1)	0: Active-low A_DOUT_EN_N port will be exposed to exercise Port A read data register enable. 1: Active-high A_DOUT_EN port will be exposed to exercise Port A read data register enable. 2: Port A read data register enable tied off to be always active.
A_DOUT_EN_PN	A_DOUT_EN, A_DOUT_EN_N	A_DOUT_EN (A_PMODE=1)	Port A read data register enable port name.
A_DOUT_SRST_POLARITY	0, 1, 2	2 (A_PMODE=1)	0: Active-low A_DOUT_SRST_N port will be exposed to exercise Port A read data register Sync-reset. 1: Active-high A_DOUT_SRST port will be exposed to exercise Port A read data register Sync-reset. 2: Port A read data register Sync- reset tied off to be always inactive.
A_DOUT_SRST_PN	A_DOUT_SRST_N, A_DOUT_SRST	A_DOUT_SRST_N (A_PMODE=1)	Port A read data register Sync-reset port name.
A_REN_POLARITY	0, 1, 2	2 (A_WMODE = 0)	0: Active-low A_REN_N port will be exposed to exercise Port A read data enable. 1: Active-high A_REN port will be exposed to exercise Port A read data enable. 2: Port A read data enable tied off to be always active.
A_REN_PN	A_REN, A_REN_N	A_REN	Port A Read data enable port name.
A_WMODE	0	0	0: Hold Port A read data.
B_PMODE	0, 1	0	0: Bypass Port B read data register. 1: Pipeline Port B read data.
B_DOUT_EN_POLARITY	0, 1, 2	2 (B_PMODE=1)	0: Active-low B_DOUT_EN_N port will be exposed to exercise Port B read data register enable. 1: Active-high B_DOUT_EN port will be exposed to exercise Port B read data register enable. 2: Port B read data register enable tied off to be always active.
B_DOUT_EN_PN	B_DOUT_EN, B_DOUT_EN_N	B_DOUT_EN (B_PMODE = 1)	Port B read data register enable port name.

.....continued

Parameter	Valid Range	Default (Condition)	Description
B_DOUT_SRST_POLARITY	0, 1, 2	2 (B_PMODE = 1)	0: Active-low B_DOUT_SRST_N port will be exposed to exercise Port B read data register Sync-reset. 1: Active-high B_DOUT_SRST port will be exposed to exercise Port B read data register Sync-reset. 2: Port B read data register Sync- reset tied off to be always inactive.
B_DOUT_SRST_PN	B_DOUT_SRST, B_DOUT_SRST_N	B_DOUT_SRST_N (B_PMODE = 1)	Port B read data register Sync-reset port name.
B_REN_POLARITY	0, 1, 2	2 (B_WMODE = 0)	0: Active-low B_REN_N port will be exposed to exercise Port B read data enable. 1: Active-high B_REN port will be exposed to exercise Port B read data enable. 2: Port B read data enable tied off to be always active.
B_REN_PN	B_REN, B_REN_N	B_REN (B_REN_POLARITY< 2)	Port B Read data enable port name.
B_WMODE	0	0	0: Hold Port B read data.
DATAA_IN_PN	—	A_DIN	Port A write data port name.
DATAB_IN_PN	—	B_DIN	Port B write data port name.
DATAA_OUT_PN	—	A_DOUT	Port A read data port name.
DATAB_OUT_PN	—	B_DOUT	Port B read data port name.
ARST_N_POLARITY	0, 1, 2	2 (Must have B_PMODE:1)	Asynchronous Reset Polarity. 0: Active-low ARST_N port will be exposed to exercise Read data register Async reset. 1: Active-high ARST port will be exposed to exercise Read data register Async reset. 2: Read data register Async-reset tied off to be always inactive.
RESET_PN	ARST_N ARST	ARST_N (A_PMODE = 1) or (B_PMODE = 1)	Read data registers Async-reset port name.
ECC	0, 1, 2	0	0: ECC disabled. 1: ECC Pipelined. 2: ECC Non-pipelined.
IMPORT_FILE	—	IMPORT_FILE Dummy parameter	Memory file to be imported to initialize RAM. No Tcl support yet.
INIT_RAM	F, T	F (INIT_RAM=T)	F: No RAM initialization for simulation. T: Initialize RAM for simulation.

7. Revision History (Ask a Question)

The revision history describes the changes that were implemented in the document. The changes are listed by revision, starting with the most current publication.

Revision	Date	Description
B	09/2023	Added a note about the **Initialize RAM for Simulation** option. See 4.2. Opening RAM Content Manager.
A	11/2020	Document is converted to the Microchip template. Initial Revision

Microchip FPGA Support

Microchip FPGA products group backs its products with various support services, including Customer Service, Customer Technical Support Center, a website, and worldwide sales offices. Customers are suggested to visit Microchip online resources prior to contacting support as it is very likely that their queries have been already answered.

Contact Technical Support Center through the website at www.microchip.com/support. Mention the FPGA Device Part number, select appropriate case category, and upload design files while creating a technical support case.

Contact Customer Service for non-technical product support, such as product pricing, product upgrades, update information, order status, and authorization.

- From North America, call **800.262.1060**
- From the rest of the world, call **650.318.4460**
- Fax, from anywhere in the world, **650.318.8044**

Microchip Information

The Microchip Website

Microchip provides online support via our website at www.microchip.com/. This website is used to make files and information easily available to customers. Some of the content available includes:

- **Product Support** – Data sheets and errata, application notes and sample programs, design resources, user's guides and hardware support documents, latest software releases and archived software
- **General Technical Support** – Frequently Asked Questions (FAQs), technical support requests, online discussion groups, Microchip design partner program member listing
- **Business of Microchip** – Product selector and ordering guides, latest Microchip press releases, listing of seminars and events, listings of Microchip sales offices, distributors and factory representatives

Product Change Notification Service

Microchip's product change notification service helps keep customers current on Microchip products. Subscribers will receive email notification whenever there are changes, updates, revisions or errata related to a specified product family or development tool of interest.

To register, go to www.microchip.com/pcn and follow the registration instructions.

Customer Support

Users of Microchip products can receive assistance through several channels:

- Distributor or Representative
- Local Sales Office
- Embedded Solutions Engineer (ESE)
- Technical Support

Customers should contact their distributor, representative or ESE for support. Local sales offices are also available to help customers. A listing of sales offices and locations is included in this document.

Technical support is available through the website at: www.microchip.com/support

Microchip Devices Code Protection Feature

Note the following details of the code protection feature on Microchip products:

- Microchip products meet the specifications contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is secure when used in the intended manner, within operating specifications, and under normal conditions.
- Microchip values and aggressively protects its intellectual property rights. Attempts to breach the code protection features of Microchip product is strictly prohibited and may violate the Digital Millennium Copyright Act.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of its code. Code protection does not mean that we are guaranteeing the product is “unbreakable”. Code protection is constantly evolving. Microchip is committed to continuously improving the code protection features of our products.

Legal Notice

This publication and the information herein may be used only with Microchip products, including to design, test, and integrate Microchip products with your application. Use of this information in any other manner violates these terms. Information regarding device applications is provided only for your convenience and may be superseded by updates. It is your responsibility to ensure that your application meets with your specifications. Contact your local Microchip sales office for additional support or, obtain additional support at www.microchip.com/en-us/support/design-help/client-support-services.

THIS INFORMATION IS PROVIDED BY MICROCHIP "AS IS". MICROCHIP MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WHETHER EXPRESS OR IMPLIED, WRITTEN OR ORAL, STATUTORY OR OTHERWISE, RELATED TO THE INFORMATION INCLUDING BUT NOT LIMITED TO ANY IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY, AND FITNESS FOR A PARTICULAR PURPOSE, OR WARRANTIES RELATED TO ITS CONDITION, QUALITY, OR PERFORMANCE.

IN NO EVENT WILL MICROCHIP BE LIABLE FOR ANY INDIRECT, SPECIAL, PUNITIVE, INCIDENTAL, OR CONSEQUENTIAL LOSS, DAMAGE, COST, OR EXPENSE OF ANY KIND WHATSOEVER RELATED TO THE INFORMATION OR ITS USE, HOWEVER CAUSED, EVEN IF MICROCHIP HAS BEEN ADVISED OF THE POSSIBILITY OR THE DAMAGES ARE FORESEEABLE. TO THE FULLEST EXTENT ALLOWED BY LAW, MICROCHIP'S TOTAL LIABILITY ON ALL CLAIMS IN ANY WAY RELATED TO THE INFORMATION OR ITS USE WILL NOT EXCEED THE AMOUNT OF FEES, IF ANY, THAT YOU HAVE PAID DIRECTLY TO MICROCHIP FOR THE INFORMATION.

Use of Microchip devices in life support and/or safety applications is entirely at the buyer's risk, and the buyer agrees to defend, indemnify and hold harmless Microchip from any and all damages, claims, suits, or expenses resulting from such use. No licenses are conveyed, implicitly or otherwise, under any Microchip intellectual property rights unless otherwise stated.

Trademarks

The Microchip name and logo, the Microchip logo, Adaptec, AVR, AVR logo, AVR Freaks, BesTime, BitCloud, CryptoMemory, CryptoRF, dsPIC, flexPWR, HELDO, IGLOO, JukeBlox, KeeLoq, Kleer, LANCheck, LinkMD, maXStylus, maXTouch, MediaLB, megaAVR, Microsemi, Microsemi logo, MOST, MOST logo, MPLAB, OptoLyzer, PIC, picoPower, PICSTART, PIC32 logo, PolarFire, Prochip Designer, QTouch, SAM-BA, SenGenuity, SpyNIC, SST, SST Logo, SuperFlash, Symmetricom, SyncServer, Tachyon, TimeSource, tinyAVR, UNI/O, Vectron, and XMEGA are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

AgileSwitch, ClockWorks, The Embedded Control Solutions Company, EtherSynch, Flashtec, Hyper Speed Control, HyperLight Load, Libero, motorBench, mTouch, Powermite 3, Precision Edge, ProASIC, ProASIC Plus, ProASIC Plus logo, Quiet-Wire, SmartFusion, SyncWorld, TimeCesium, TimeHub, TimePictra, TimeProvider, and ZL are registered trademarks of Microchip Technology Incorporated in the U.S.A.

Adjacent Key Suppression, AKS, Analog-for-the-Digital Age, Any Capacitor, AnyIn, AnyOut, Augmented Switching, BlueSky, BodyCom, Clockstudio, CodeGuard, CryptoAuthentication, CryptoAutomotive, CryptoCompanion, CryptoController, dsPICDEM, dsPICDEM.net, Dynamic

ISBN: 978-1-6683-3110-1

Quality Management System

For information regarding Microchip's Quality Management Systems, please visit www.microchip.com/quality.

Worldwide Sales and Service

AMERICAS	ASIA/PACIFIC	ASIA/PACIFIC	EUROPE
Corporate Office	**Australia - Sydney**	**India - Bangalore**	**Austria - Wels**
2355 West Chandler Blvd.	Tel: 61-2-9868-6733	Tel: 91-80-3090-4444	Tel: 43-7242-2244-39
Chandler, AZ 85224-6199	**China - Beijing**	**India - New Delhi**	Fax: 43-7242-2244-393
Tel: 480-792-7200	Tel: 86-10-8569-7000	Tel: 91-11-4160-8631	**Denmark - Copenhagen**
Fax: 480-792-7277	**China - Chengdu**	**India - Pune**	Tel: 45-4485-5910
Technical Support:	Tel: 86-28-8665-5511	Tel: 91-20-4121-0141	Fax: 45-4485-2829
www.microchip.com/support	**China - Chongqing**	**Japan - Osaka**	**Finland - Espoo**
Web Address:	Tel: 86-23-8980-9588	Tel: 81-6-6152-7160	Tel: 358-9-4520-820
www.microchip.com	**China - Dongguan**	**Japan - Tokyo**	**France - Paris**
Atlanta	Tel: 86-769-8702-9880	Tel: 81-3-6880- 3770	Tel: 33-1-69-53-63-20
Duluth, GA	**China - Guangzhou**	**Korea - Daegu**	Fax: 33-1-69-30-90-79
Tel: 678-957-9614	Tel: 86-20-8755-8029	Tel: 82-53-744-4301	**Germany - Garching**
Fax: 678-957-1455	**China - Hangzhou**	**Korea - Seoul**	Tel: 49-8931-9700
Austin, TX	Tel: 86-571-8792-8115	Tel: 82-2-554-7200	**Germany - Haan**
Tel: 512-257-3370	**China - Hong Kong SAR**	**Malaysia - Kuala Lumpur**	Tel: 49-2129-3766400
Boston	Tel: 852-2943-5100	Tel: 60-3-7651-7906	**Germany - Heilbronn**
Westborough, MA	**China - Nanjing**	**Malaysia - Penang**	Tel: 49-7131-72400
Tel: 774-760-0087	Tel: 86-25-8473-2460	Tel: 60-4-227-8870	**Germany - Karlsruhe**
Fax: 774-760-0088	**China - Qingdao**	**Philippines - Manila**	Tel: 49-721-625370
Chicago	Tel: 86-532-8502-7355	Tel: 63-2-634-9065	**Germany - Munich**
Itasca, IL	**China - Shanghai**	**Singapore**	Tel: 49-89-627-144-0
Tel: 630-285-0071	Tel: 86-21-3326-8000	Tel: 65-6334-8870	Fax: 49-89-627-144-44
Fax: 630-285-0075	**China - Shenyang**	**Taiwan - Hsin Chu**	**Germany - Rosenheim**
Dallas	Tel: 86-24-2334-2829	Tel: 886-3-577-8366	Tel: 49-8031-354-560
Addison, TX	**China - Shenzhen**	**Taiwan - Kaohsiung**	**Israel - Ra'anana**
Tel: 972-818-7423	Tel: 86-755-8864-2200	Tel: 886-7-213-7830	Tel: 972-9-744-7705
Fax: 972-818-2924	**China - Suzhou**	**Taiwan - Taipei**	**Italy - Milan**
Detroit	Tel: 86-186-6233-1526	Tel: 886-2-2508-8600	Tel: 39-0331-742611
Novi, MI	**China - Wuhan**	**Thailand - Bangkok**	Fax: 39-0331-466781
Tel: 248-848-4000	Tel: 86-27-5980-5300	Tel: 66-2-694-1351	**Italy - Padova**
Houston, TX	**China - Xian**	**Vietnam - Ho Chi Minh**	Tel: 39-049-7625286
Tel: 281-894-5983	Tel: 86-29-8833-7252	Tel: 84-28-5448-2100	**Netherlands - Drunen**
Indianapolis	**China - Xiamen**		Tel: 31-416-690399
Noblesville, IN	Tel: 86-592-2388138		Fax: 31-416-690340
Tel: 317-773-8323	**China - Zhuhai**		**Norway - Trondheim**
Fax: 317-773-5453	Tel: 86-756-3210040		Tel: 47-72884388
Tel: 317-536-2380			**Poland - Warsaw**
Los Angeles			Tel: 48-22-3325737
Mission Viejo, CA			**Romania - Bucharest**
Tel: 949-462-9523			Tel: 40-21-407-87-50
Fax: 949-462-9608			**Spain - Madrid**
Tel: 951-273-7800			Tel: 34-91-708-08-90
Raleigh, NC			Fax: 34-91-708-08-91
Tel: 919-844-7510			**Sweden - Gothenberg**
New York, NY			Tel: 46-31-704-60-40
Tel: 631-435-6000			**Sweden - Stockholm**
San Jose, CA			Tel: 46-8-5090-4654
Tel: 408-735-9110			**UK - Wokingham**
Tel: 408-436-4270			Tel: 44-118-921-5800
Canada - Toronto			Fax: 44-118-921-5820
Tel: 905-695-1980			
Fax: 905-695-2078			