Laboratory 5 (100 Points)

Assignment 1 (10 Points)

Design a logic diagram to implement 2-4 decoder with Enable pin (using buses for grouping lines in the same group)

1. Derive the decoder's truth table.
2. Design the logic diagram to implement the 2-4 decoder.
3. Simulate the circuit and check the result with the truth table.

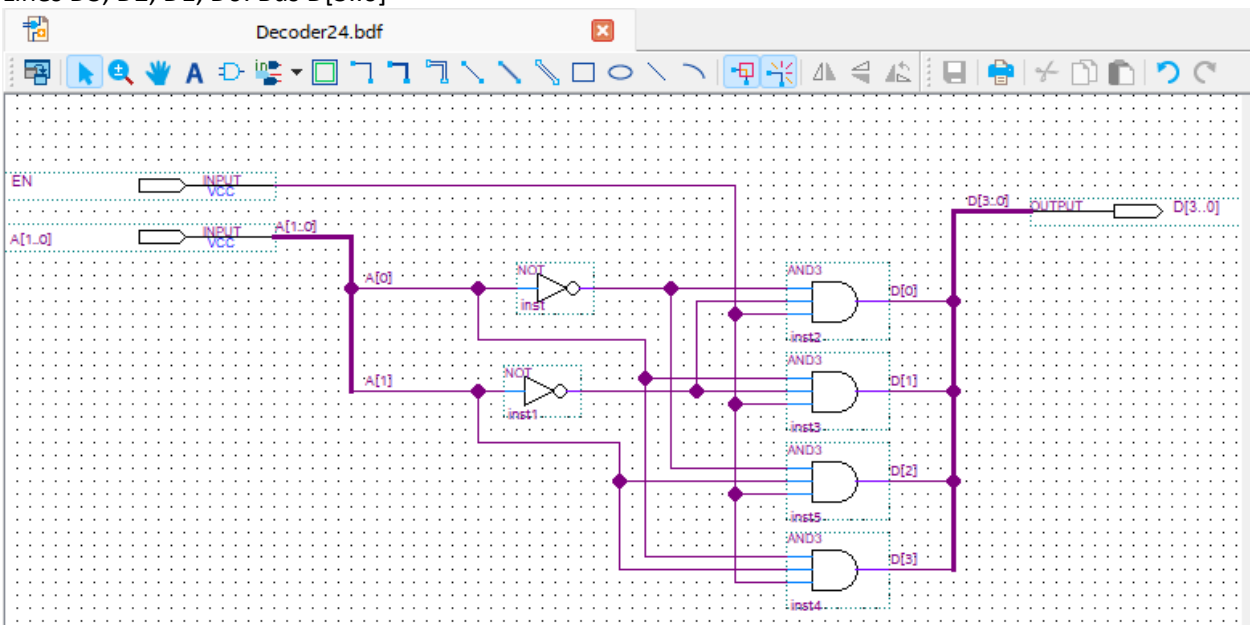
1. Truth table

EN	A1	A0	D0	D1	D2	D3
0	x	x	0	0	0	0
1	0	0	1	0	0	0
1	0	1	0	1	0	0
1	1	0	0	0	1	0
1	1	1	0	0	0	1

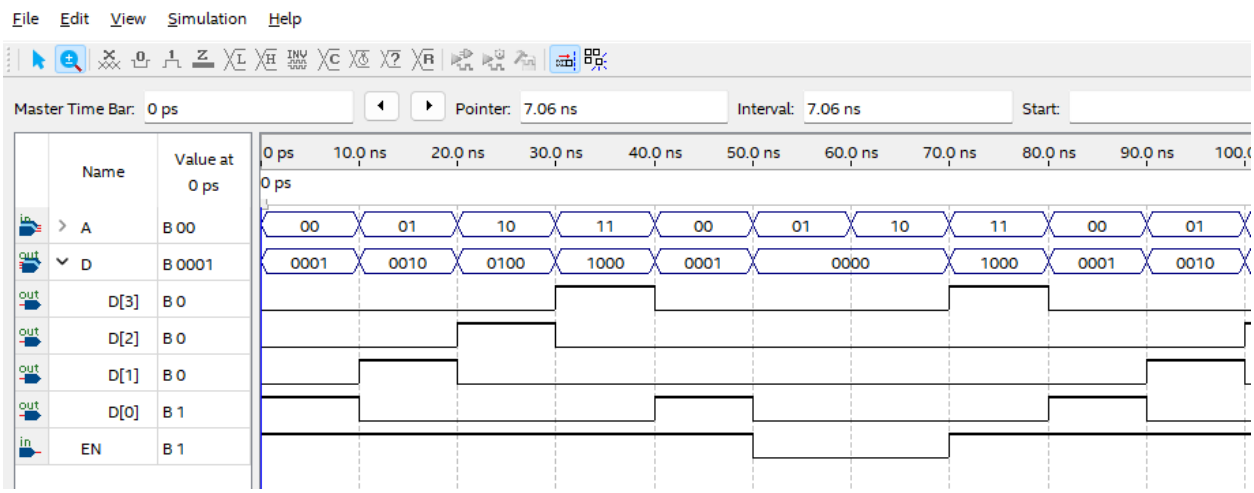
2. Design the 2-4 decoder's logic diagram

Lines A1, A0: Bus A[1..0]

Lines D3, D2, D1, D0: Bus D[3..0]



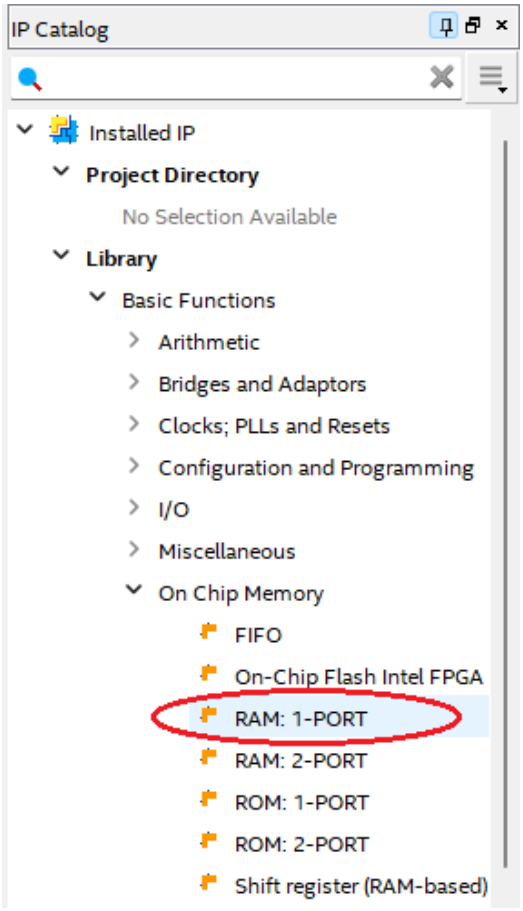
3. Simulate the circuit



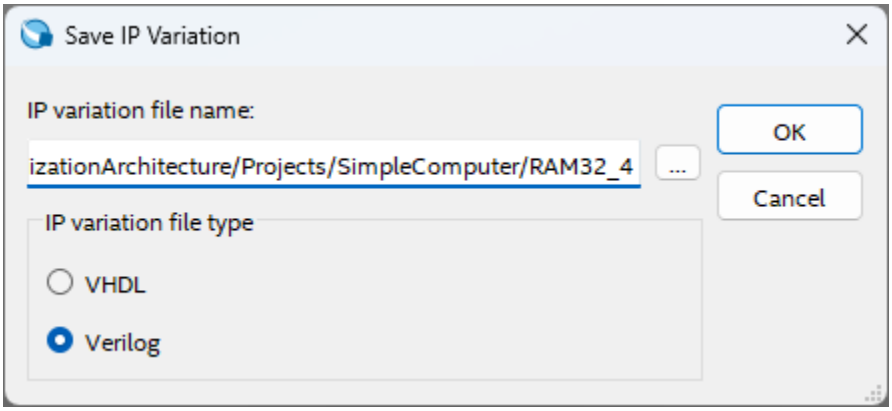
Assignment 2 (10 Points)

Create a 128-bit RAM organized as 32-word by 4-bit array.

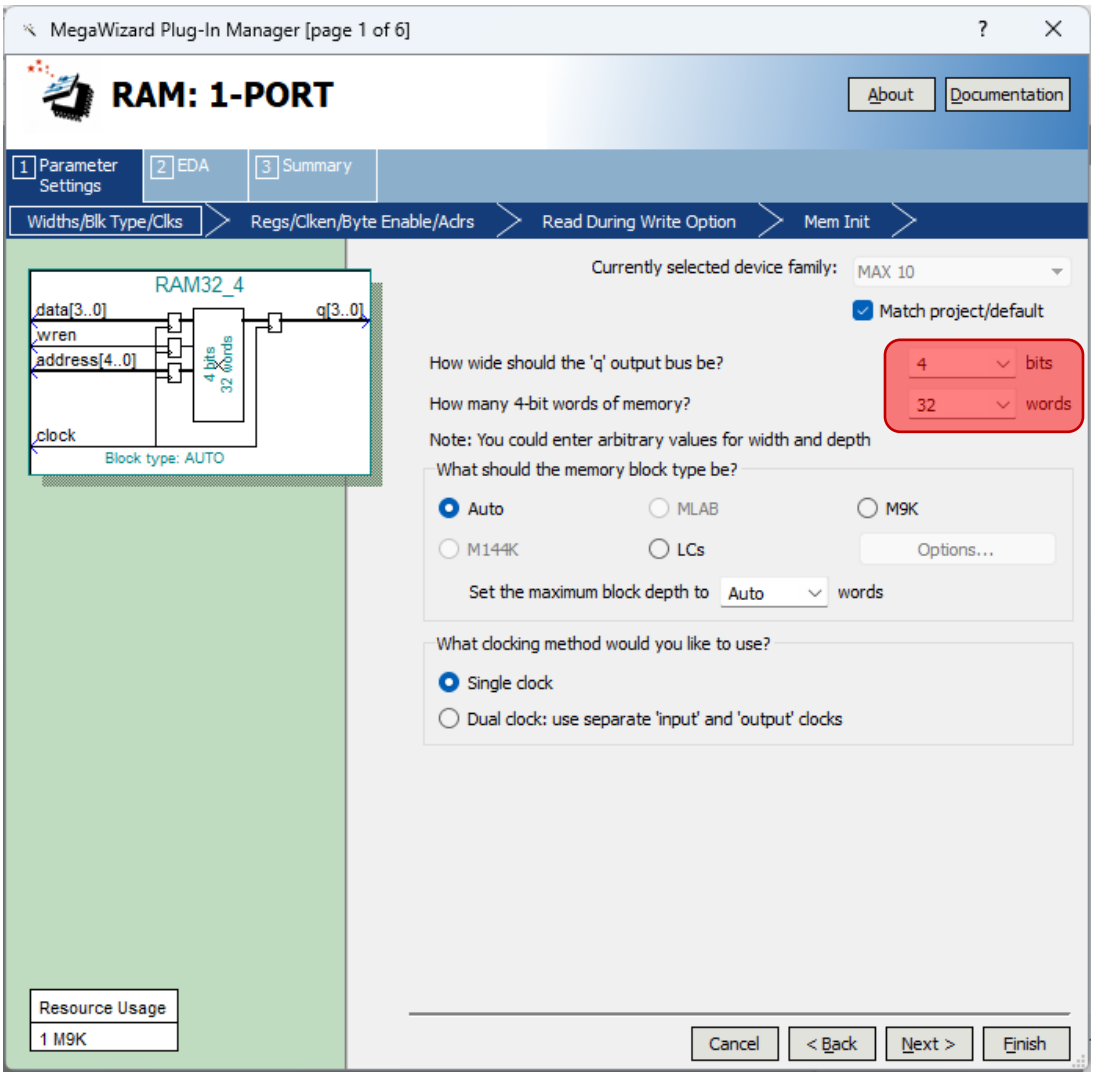
1. Create the 32-4 RAM (32 by 4)

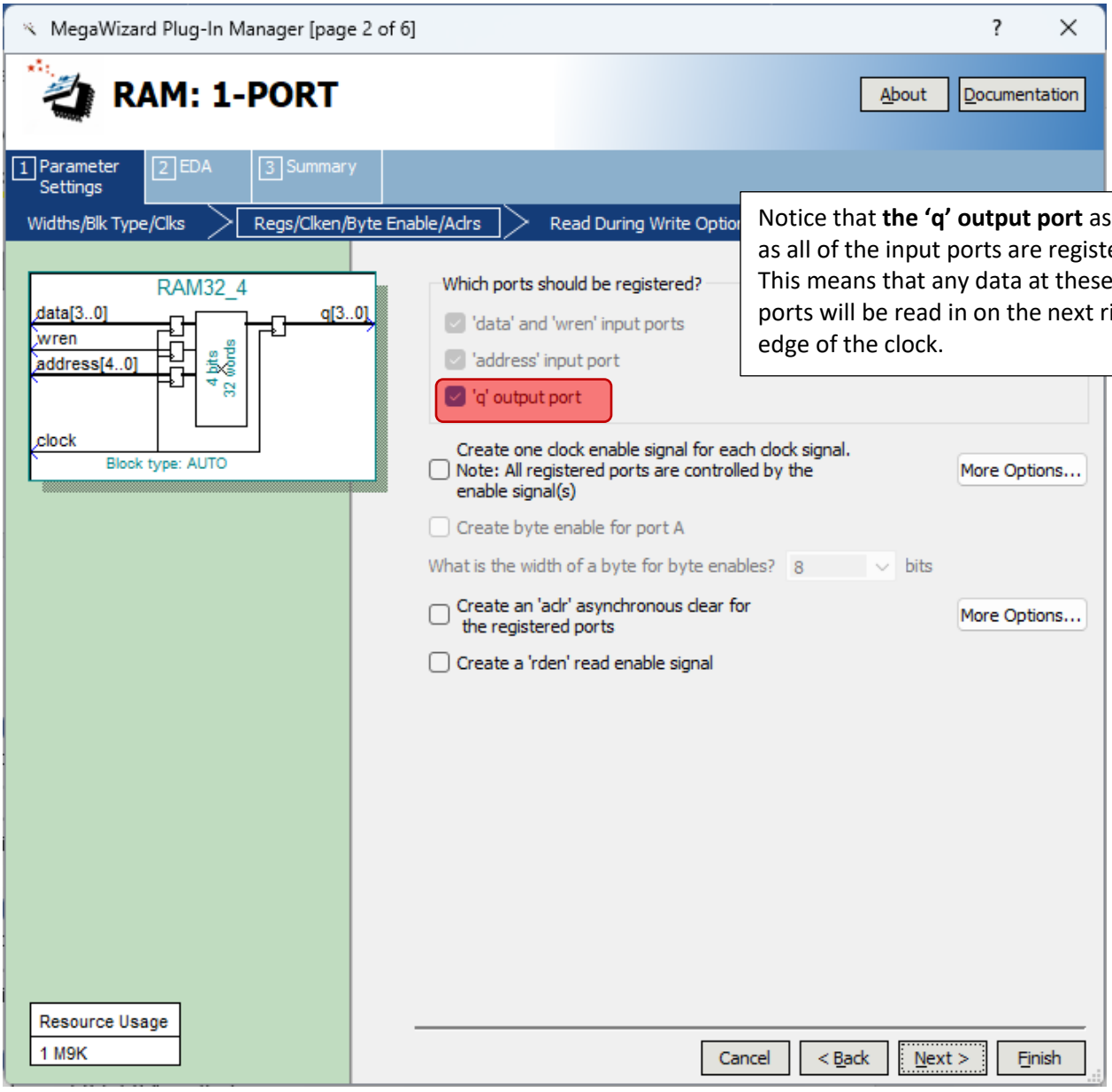


- Name the RAM as RAM32_4:

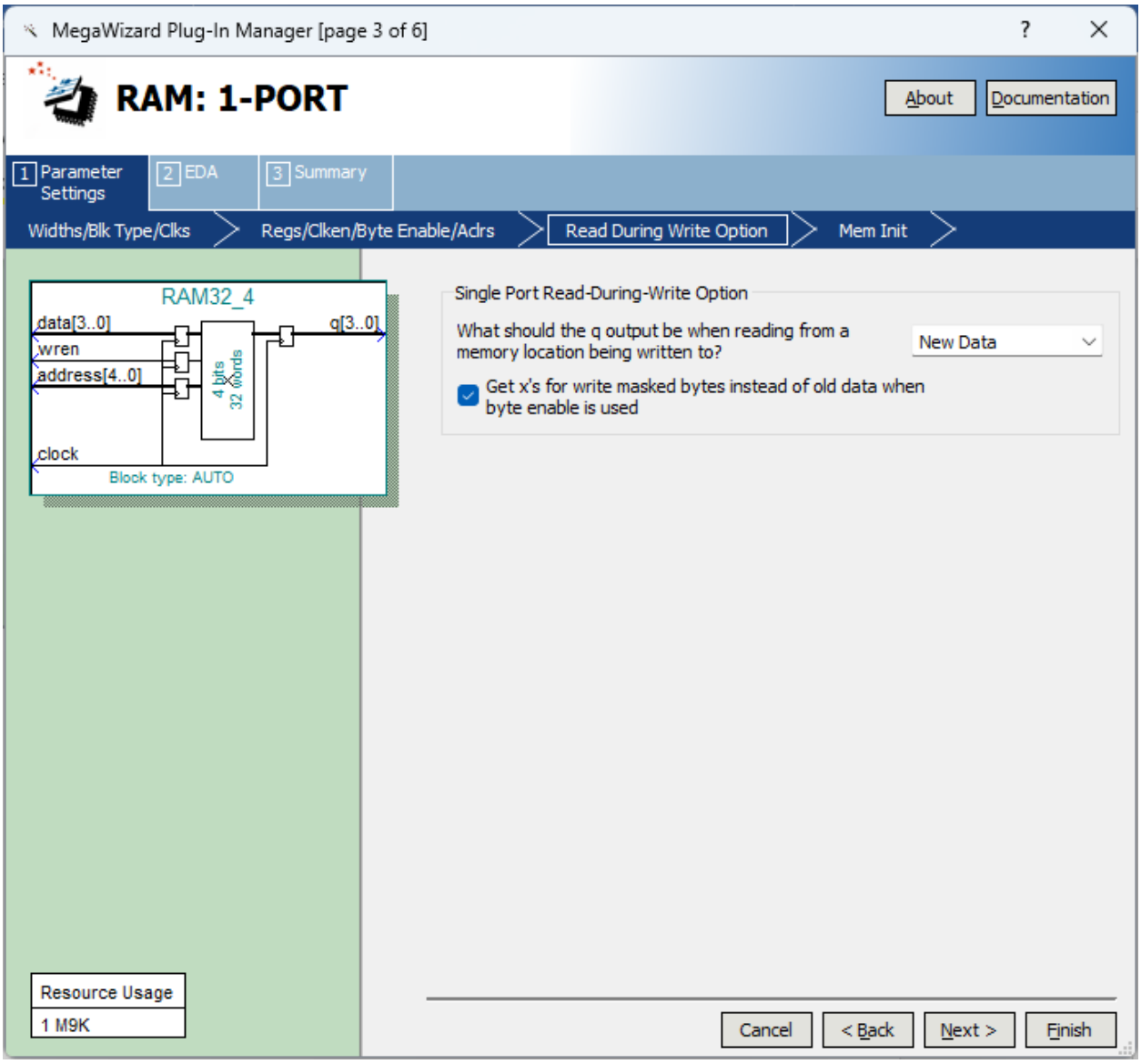


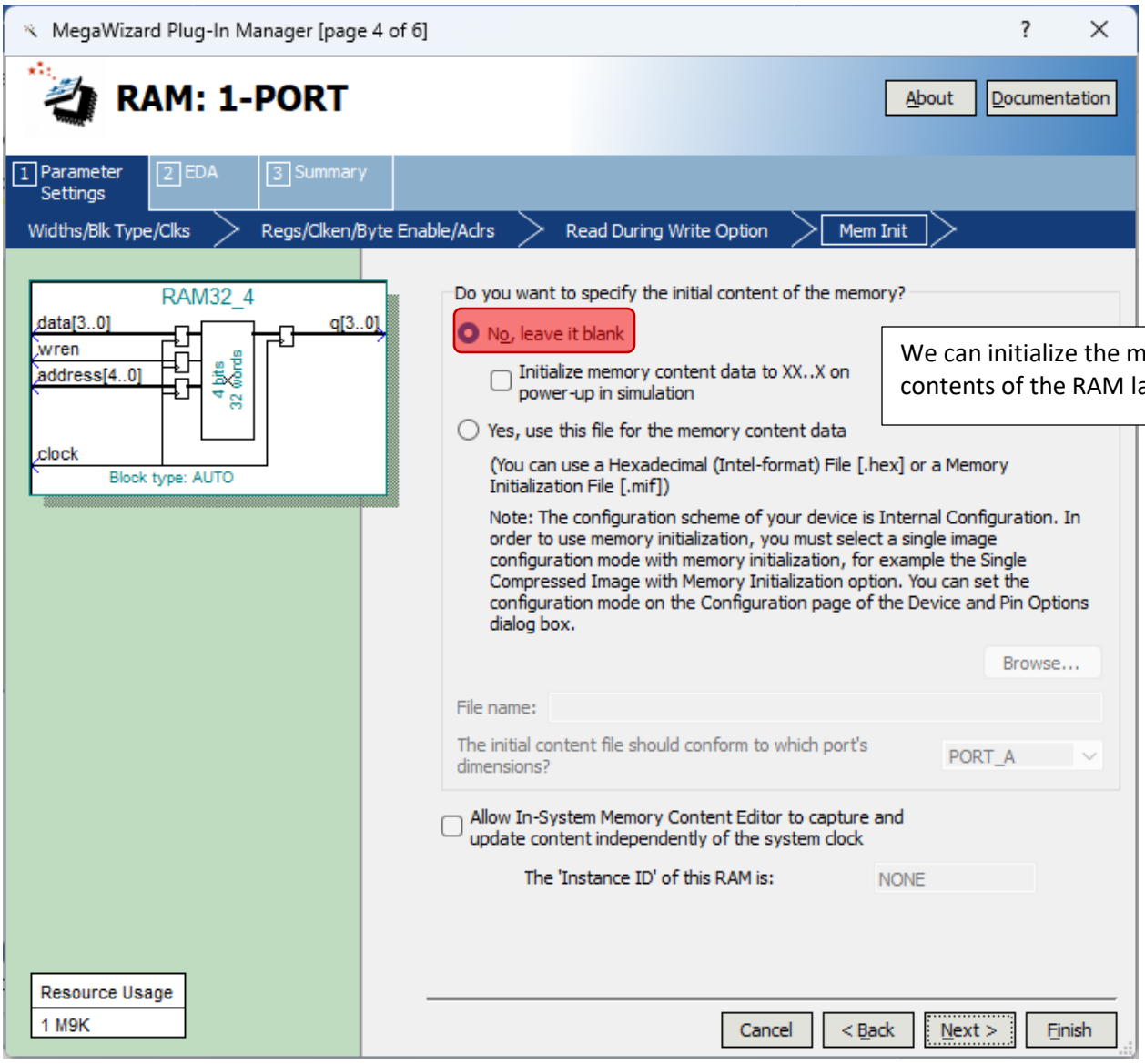
- The **MegaWizard Plug-In Manager** window is shown as in the following screen.





Notice that **the 'q' output port** as well as all of the input ports are registered. This means that any data at these ports will be read in on the next rising edge of the clock.


MegaWizard Plug-In Manager [page 3 of 6]
RAM: 1-PORT
About
Documentation
1 Parameter Settings
2 EDA
3 Summary
Widths/Blk Type/Clks
Regs/Clken/Byte Enable/Adrs
Read During Write Option
Mem Init
RAM32_4
data[3..0]
wren
address[4..0]
clock
q[3..0]
4 bits
32 words
Block type: AUTO
Single Port Read-During-Write Option
What should the q output be when reading from a memory location being written to?
New Data
Get x's for write masked bytes instead of old data when byte enable is used
Resource Usage
1 M9K
Cancel
< Back
Next >
Finish



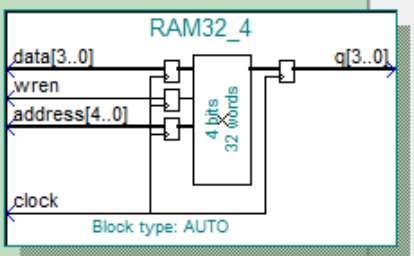
We can initialize the memory contents of the RAM later.

MegaWizard Plug-In Manager [page 5 of 6]

RAM: 1-PORT

About | Documentation

1 Parameter Settings | 2 EDA | 3 Summary



Simulation Libraries

To properly simulate the generated design files, the following simulation model file(s) are needed

File	Description
altera_mf	Altera megafunction simulation library

Timing and resource estimation

Generates a netlist for timing and resource estimation for this megafunction. If you are synthesizing your design with a third-party EDA synthesis tool, using a timing and resource estimation netlist can allow for better design optimization.

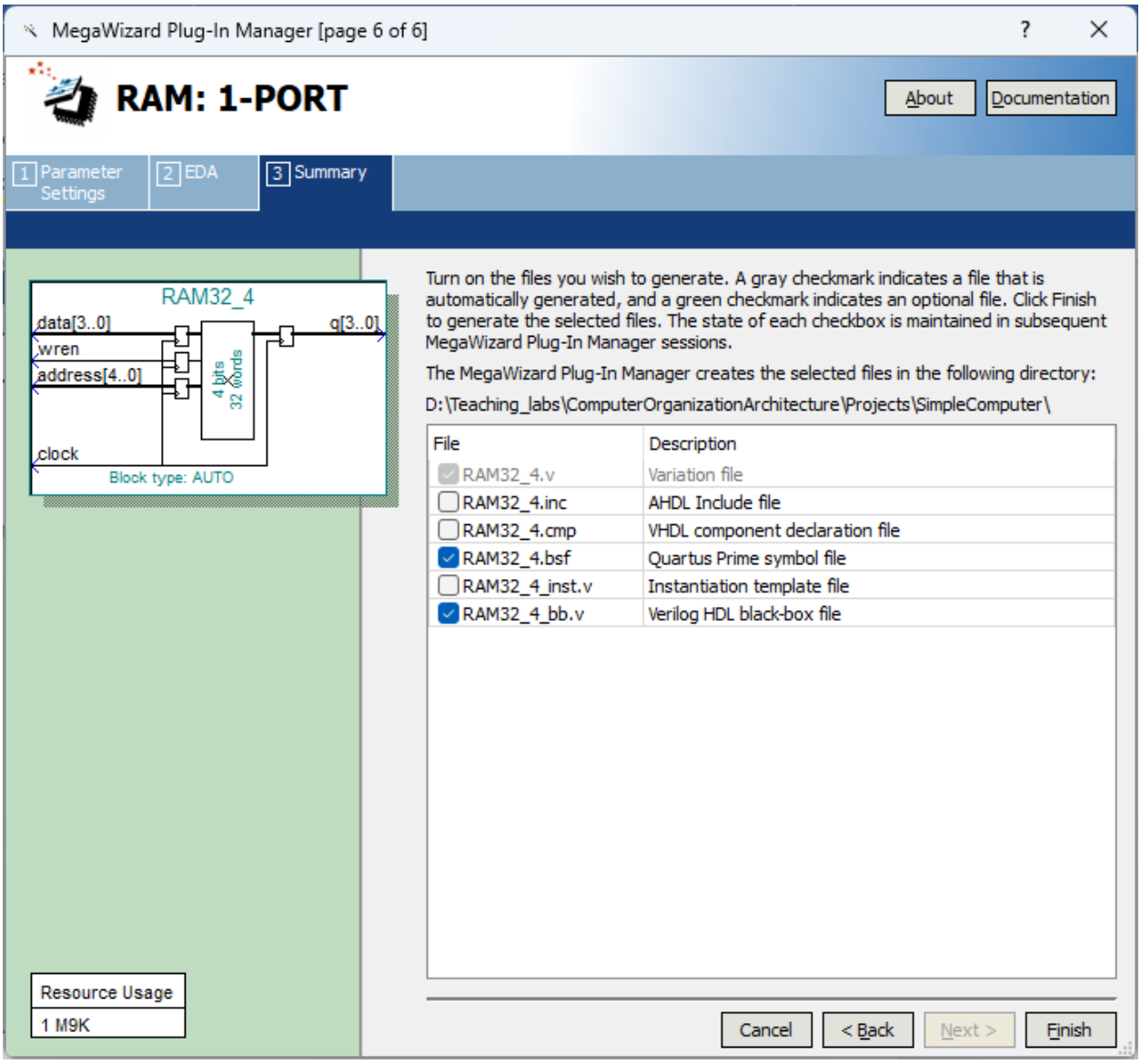
Not all third-party synthesis tools support this feature - check with the tool vendor for complete support information.

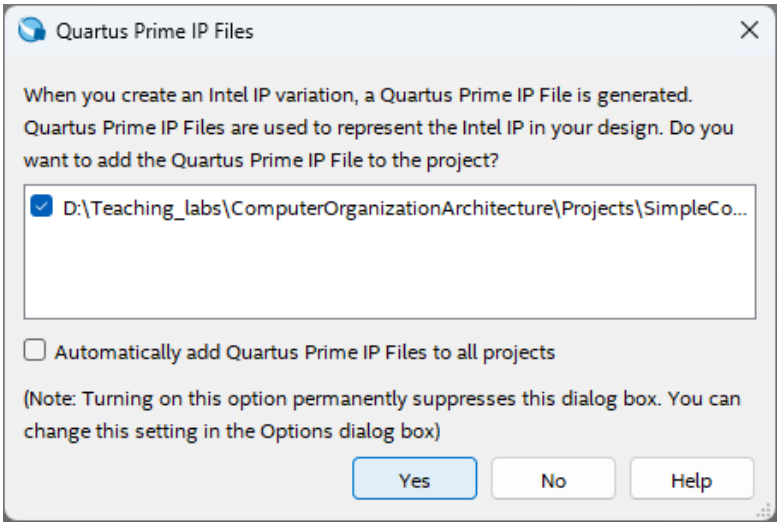
Note: Netlist generation can be a time-intensive process. The size of the design and the speed of your system affect the time it takes for netlist generation to complete.

☐ Generate netlist

Resource Usage
1 M9K

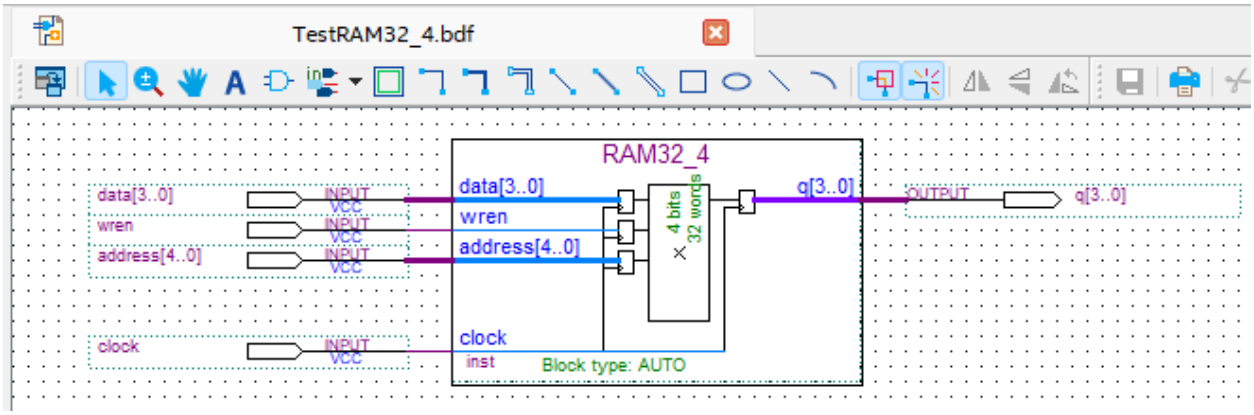
Cancel | < Back | Next > | Finish


MegaWizard Plug-In Manager [page 6 of 6]
RAM: 1-PORT
About
Documentation
1 Parameter Settings
2 EDA
3 Summary
RAM32_4
data[3..0]
wren
address[4..0]
clock
q[3..0]
4 bits
32 words
Block type: AUTO
Turn on the files you wish to generate. A gray checkmark indicates a file that is automatically generated, and a green checkmark indicates an optional file. Click Finish to generate the selected files. The state of each checkbox is maintained in subsequent MegaWizard Plug-In Manager sessions.
The MegaWizard Plug-In Manager creates the selected files in the following directory:
D:\Teaching_labs\ComputerOrganizationArchitecture\Projects\SimpleComputer\
File
Description
RAM32_4.v
Variation file
RAM32_4.inc
AHDL Include file
RAM32_4.cmp
VHDL component declaration file
RAM32_4.bsf
Quartus Prime symbol file
RAM32_4_inst.v
Instantiation template file
RAM32_4_bb.v
Verilog HDL black-box file
Resource Usage
1 M9K
Cancel
< Back
Next >
Finish


Quartus Prime IP Files
When you create an Intel IP variation, a Quartus Prime IP File is generated. Quartus Prime IP Files are used to represent the Intel IP in your design. Do you want to add the Quartus Prime IP File to the project?
D:\Teaching_labs\ComputerOrganizationArchitecture\Projects\SimpleCo...
Automatically add Quartus Prime IP Files to all projects
(Note: Turning on this option permanently suppresses this dialog box. You can change this setting in the Options dialog box)
Yes
No
Help

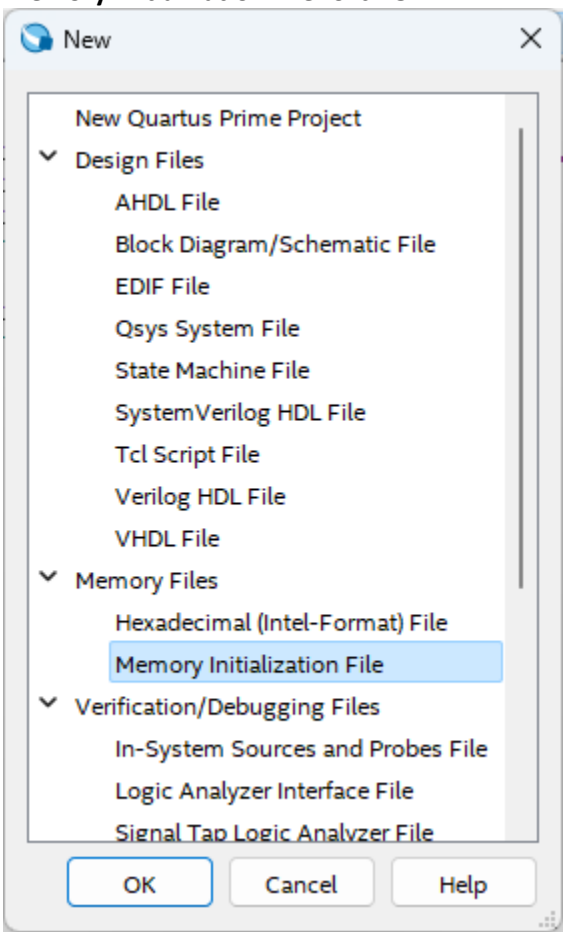
You should now see the symbol for the RAM we just created. Click **Yes** and place the RAM in your design.

2. Use the RAM32_4

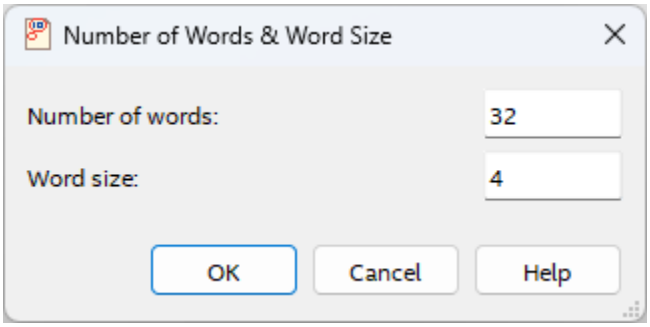


3. We now initialize the memory contents of our RAM.

We will use a **Memory Initialization File (.MIF)**. Select **File > New** from the menu and then select **Memory Initialization File**. Click **OK**.



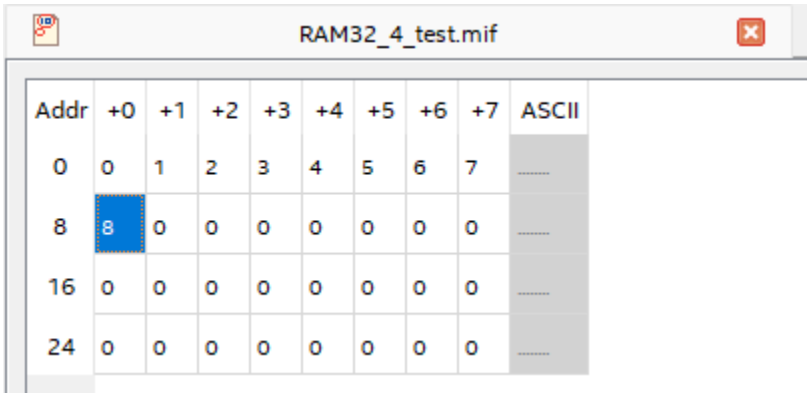
Select 16 words and 4 bit word size to match our RAM block.



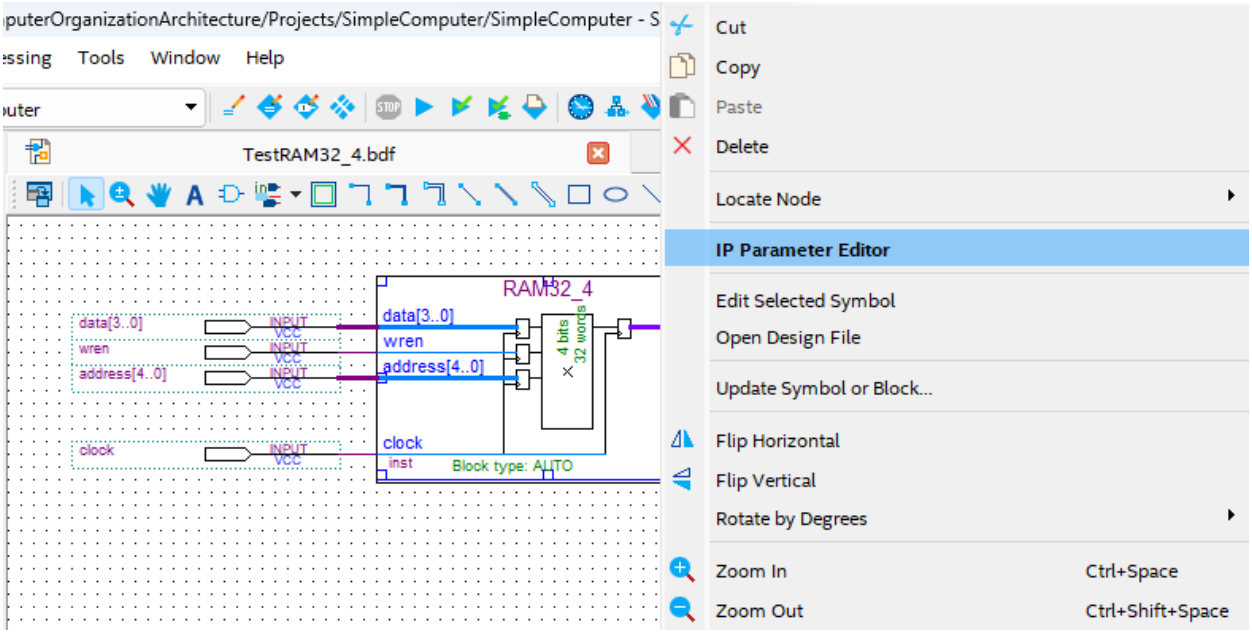
You can now fill the memory with any values from 0 to 15 (15 is the maximum because our word size is 4 bits).

Right clicking on a cell or range of cells will allow us to rapidly fill multiple memory addresses with custom values.

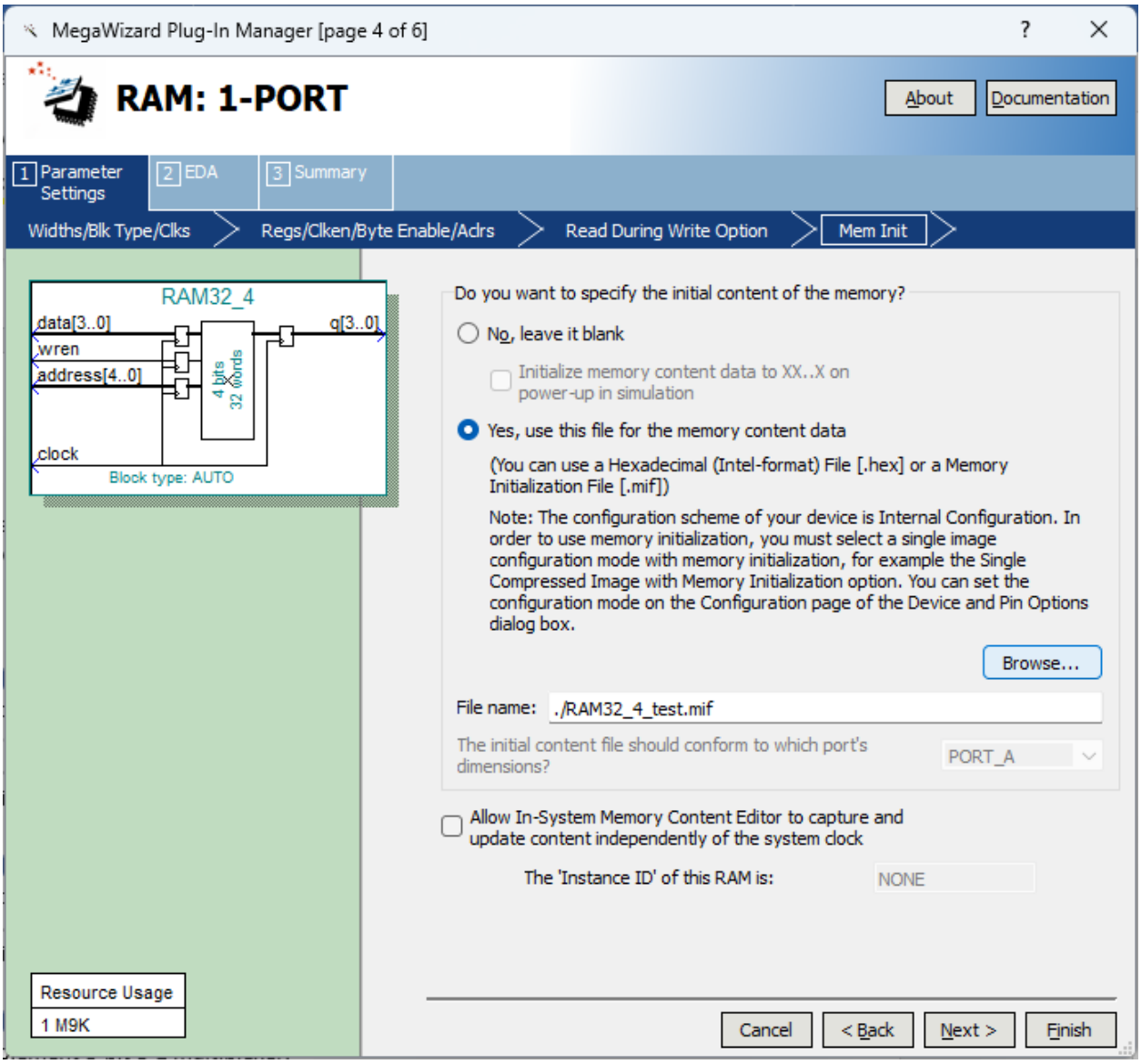
Remember to save the .imf file. The file name should default to RAM32_4_test.mif

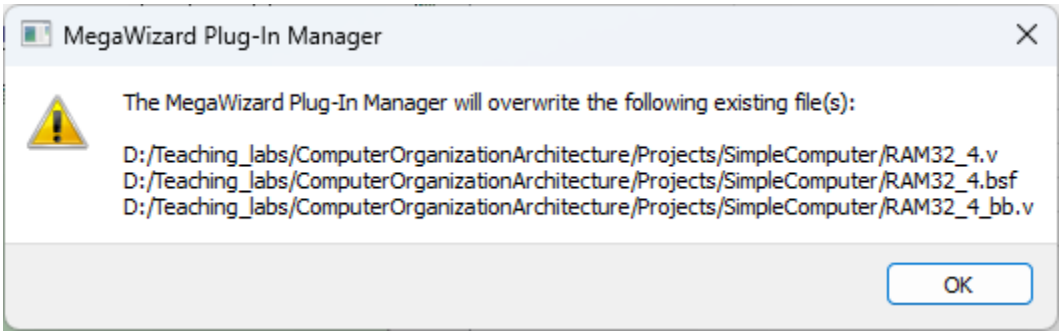


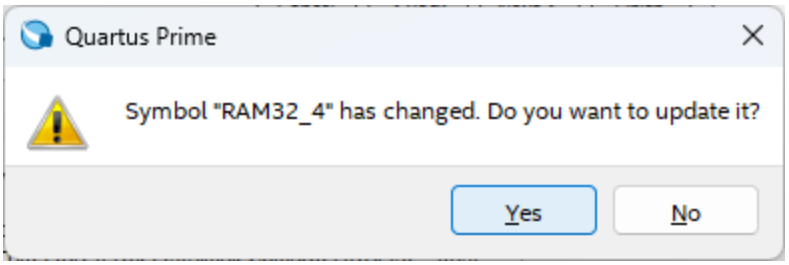
Now we must update our RAM to initialize itself to our memory initialization file. Right click the RAM component in the schematic and select **IP Parameter Editor**.

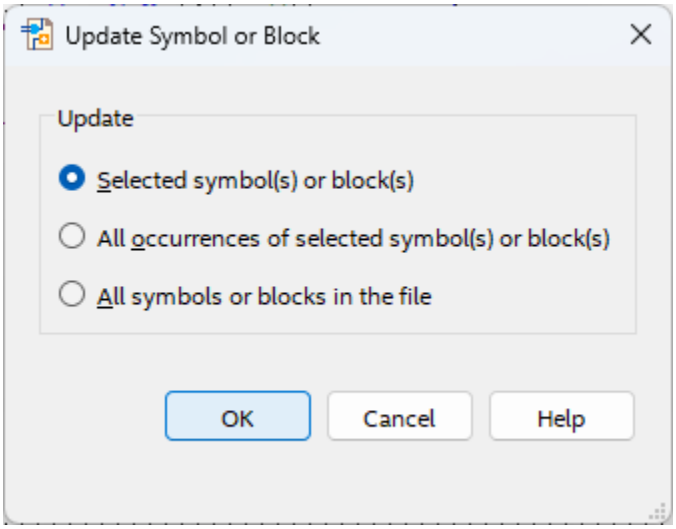


Click Next until you reach the **Mem Init** tab (or just click the **Mem Init** tab). Select **Yes, use this file for the memory content data** to specify a file and then select browse and find your .mif file.
Select the file and then click Finish.

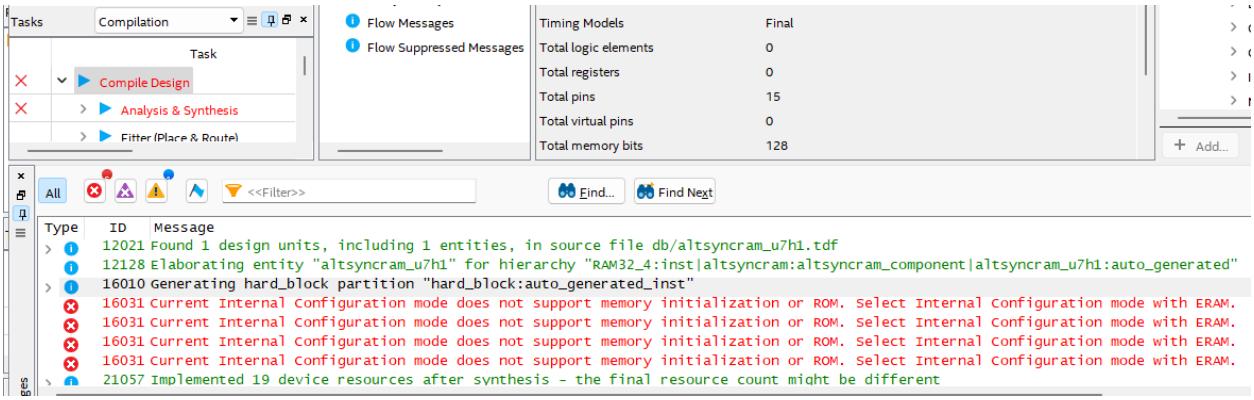

MegaWizard Plug-In Manager [page 4 of 6]
RAM: 1-PORT
About
Documentation
1 Parameter Settings
2 EDA
3 Summary
Widths/Blk Type/Clks
Regs/Clken/Byte Enable/Adrs
Read During Write Option
Mem Init
RAM32_4
data[3..0]
wren
address[4..0]
clock
q[3..0]
4 bits
32 words
Block type: AUTO
Do you want to specify the initial content of the memory?
No, leave it blank
Initialize memory content data to XX..X on power-up in simulation
Yes, use this file for the memory content data
(You can use a Hexadecimal (Intel-format) File [.hex] or a Memory Initialization File [.mif])
Note: The configuration scheme of your device is Internal Configuration. In order to use memory initialization, you must select a single image configuration mode with memory initialization, for example the Single Compressed Image with Memory Initialization option. You can set the configuration mode on the Configuration page of the Device and Pin Options dialog box.
Browse...
File name: ./RAM32_4_test.mif
The initial content file should conform to which port's dimensions?
PORT_A
Allow In-System Memory Content Editor to capture and update content independently of the system clock
The 'Instance ID' of this RAM is:
NONE
Resource Usage
1 M9K
Cancel
< Back
Next >
Finish


MegaWizard Plug-In Manager
The MegaWizard Plug-In Manager will overwrite the following existing file(s):
D:/Teaching_labs/ComputerOrganizationArchitecture/Projects/SimpleComputer/RAM32_4.v
D:/Teaching_labs/ComputerOrganizationArchitecture/Projects/SimpleComputer/RAM32_4.bsf
D:/Teaching_labs/ComputerOrganizationArchitecture/Projects/SimpleComputer/RAM32_4_bb.v
OK



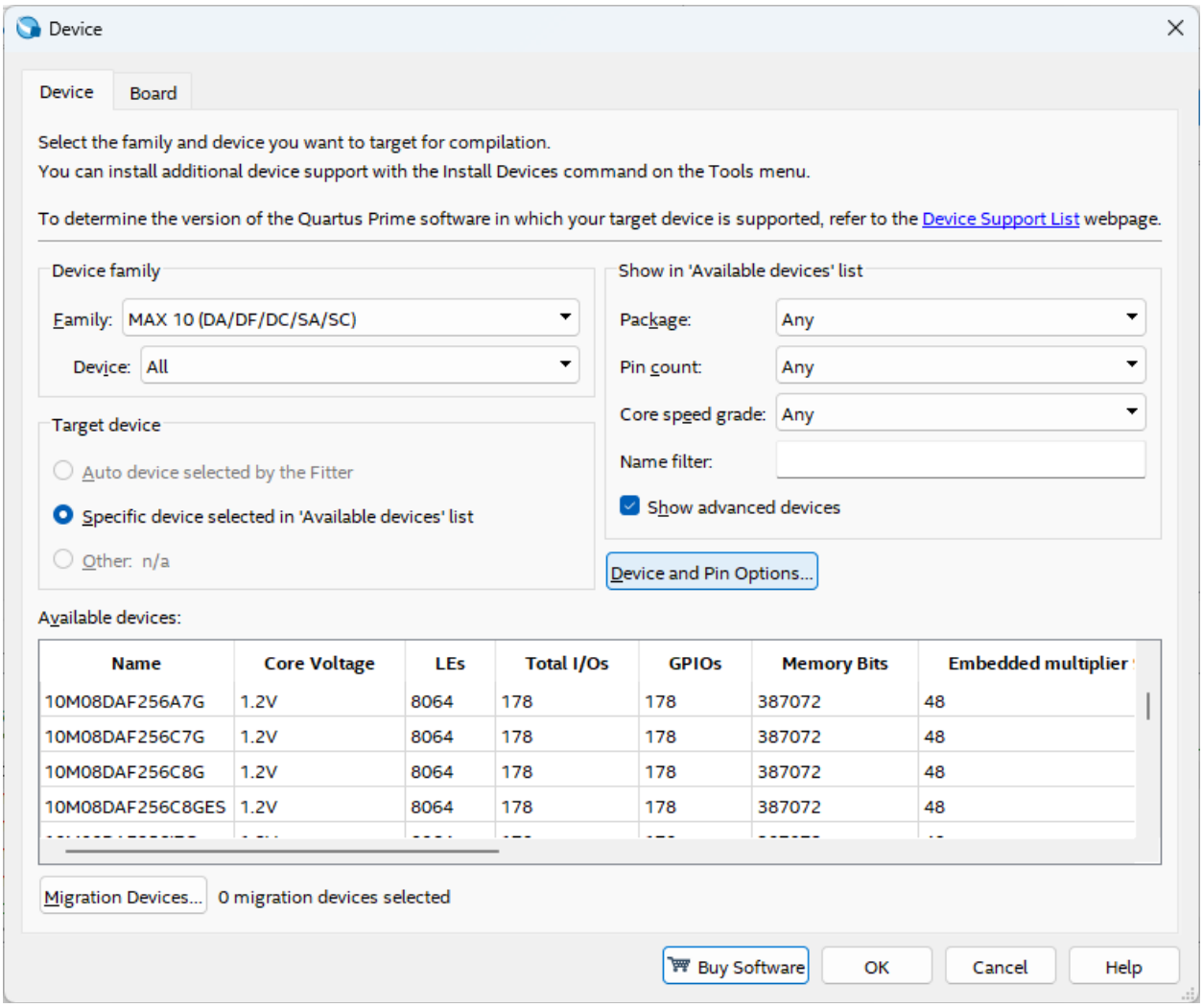


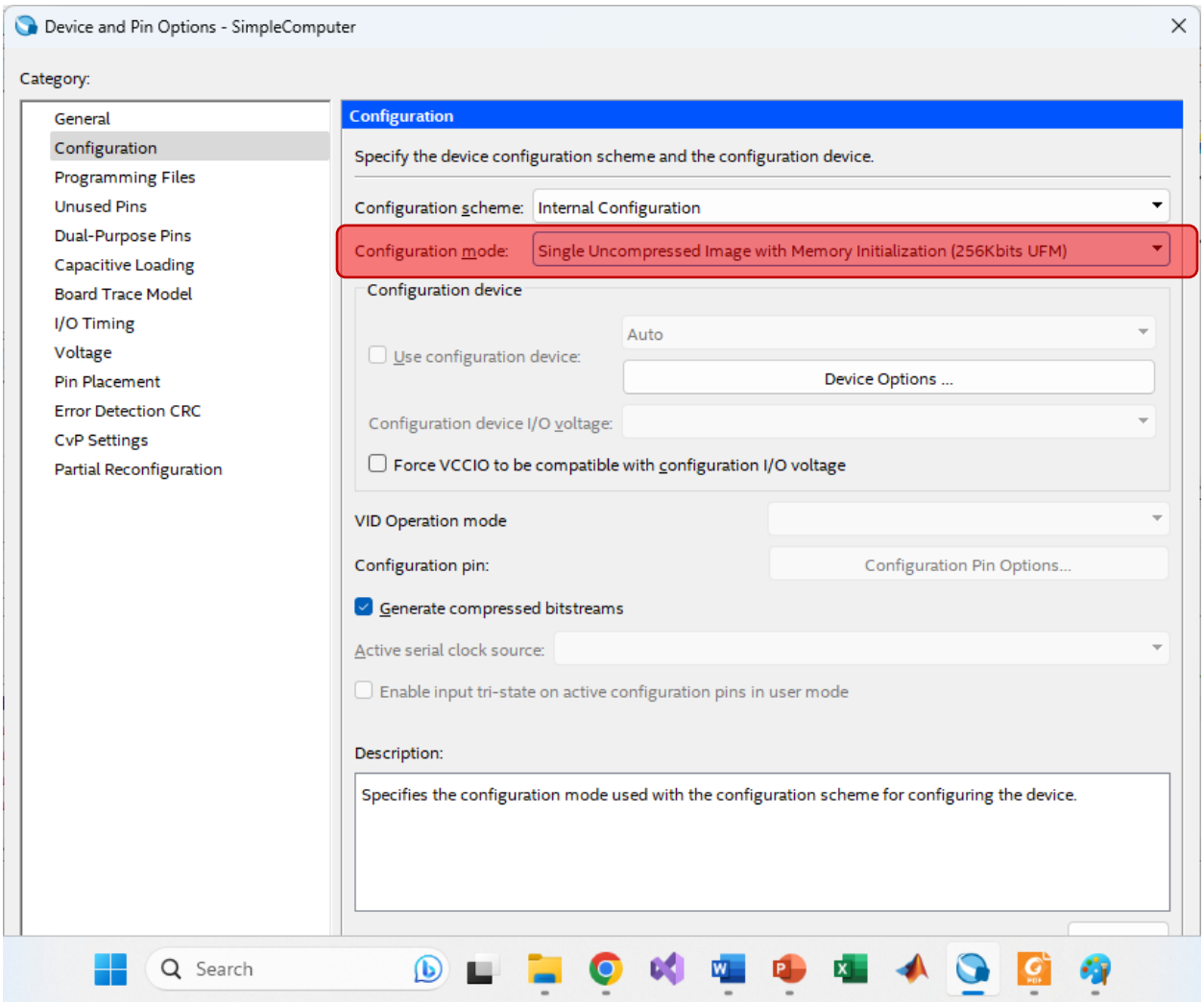
Compile the design



During compilation if you get the error message:
"16031 Current Internal Configuration mode does not support memory initialization or ROM. Select Internal Configuration mode with ERAM.",

Select **Assignments -> Device -> Device and Pin Options -> Configuration -> Configuration Mode: Single uncompressed image with Memory Initialization**

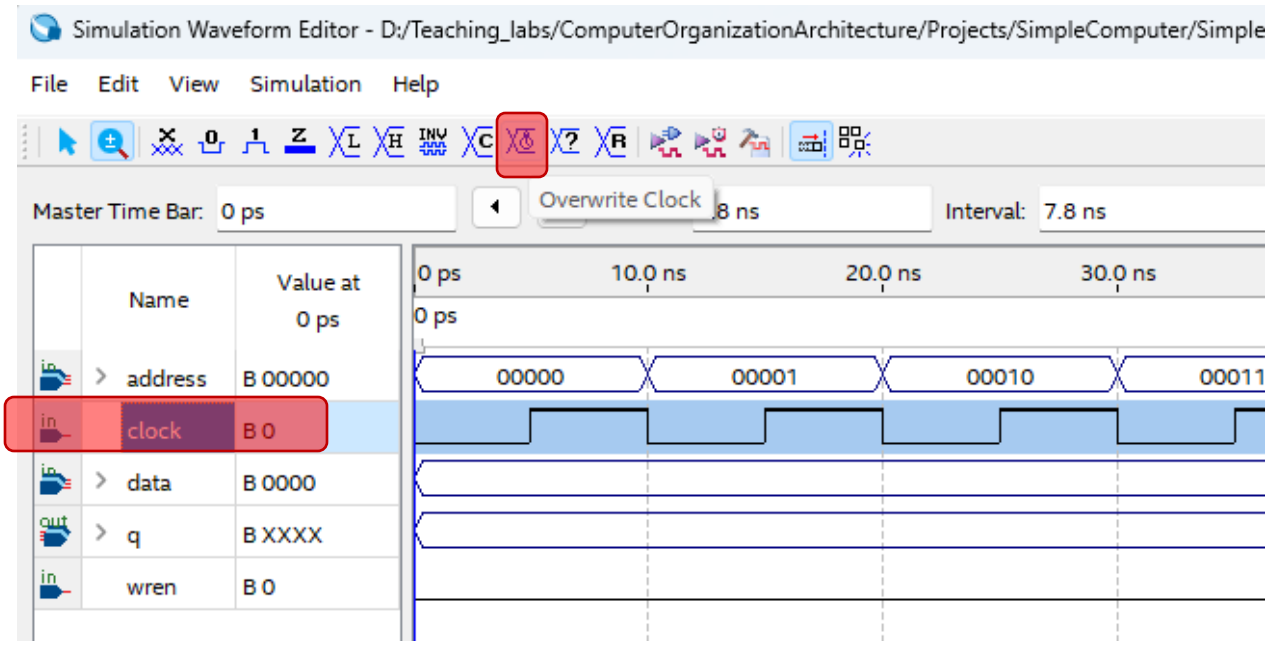

Device
Device
Board
Select the family and device you want to target for compilation.
You can install additional device support with the Install Devices command on the Tools menu.
To determine the version of the Quartus Prime software in which your target device is supported, refer to the Device Support List webpage.
Device family
Family: MAX 10 (DA/DF/DC/SA/SC)
Device: All
Target device
Auto device selected by the Fitter
Specific device selected in 'Available devices' list
Other: n/a
Show in 'Available devices' list
Package: Any
Pin count: Any
Core speed grade: Any
Name filter:
Show advanced devices
Device and Pin Options...
Available devices:
Name
Core Voltage
LEs
Total I/Os
GPIOs
Memory Bits
Embedded multiplier
10M08DAF256A7G 1.2V 8064 178 178 387072 48
10M08DAF256C7G 1.2V 8064 178 178 387072 48
10M08DAF256C8G 1.2V 8064 178 178 387072 48
10M08DAF256C8GES 1.2V 8064 178 178 387072 48
Migration Devices...
0 migration devices selected
Buy Software
OK
Cancel
Help



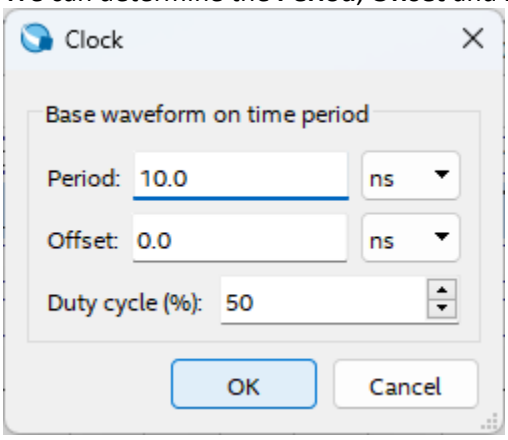
Recompile the project again.

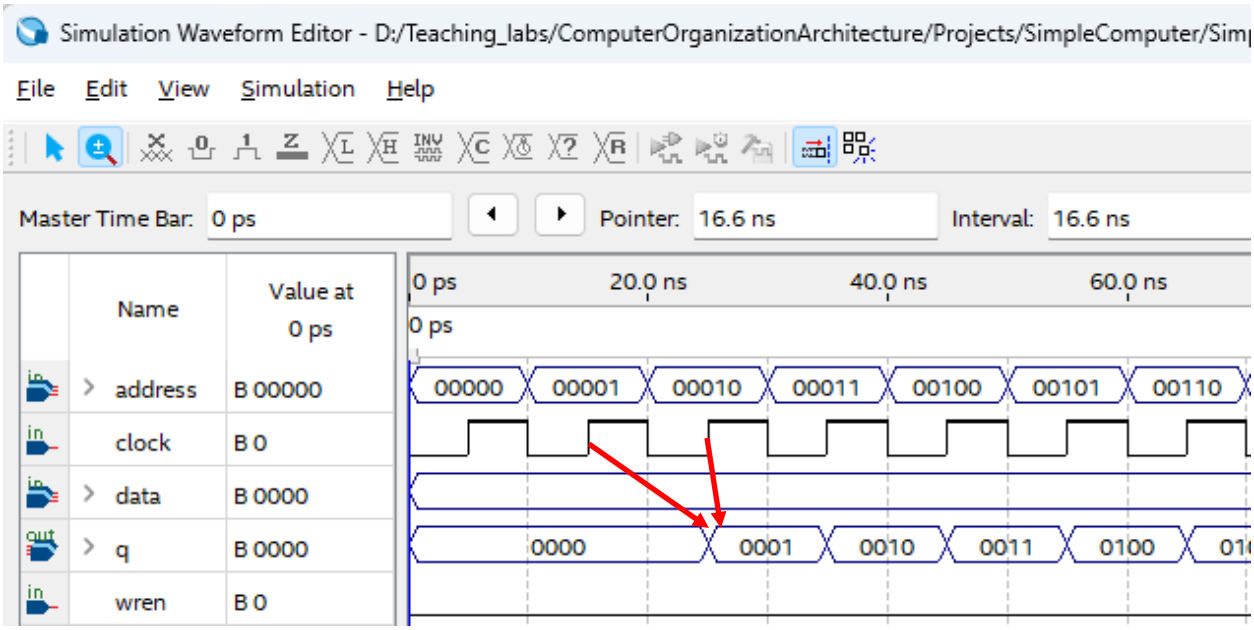
4. Simulate the circuit

- Create the clock signal for the **clock** line:
    + Select the **clock** line and select **Overwrite Clock**.



We can determine the **Period**, **Offset** and **Duty cycle** of the clock signal.





You should see that the **q** output matches the data values we stored in the RAM using our memory initialization file.
Notice that the output is updated on the second rising edge of the clock after a new address is presented to the RAM. This is because we selected to register both the input and output ports of our RAM in the MegaWizard Plug-In Manager. On the first rising edge, the RAM receives whatever address is presented at the address pins. The RAM reads the data value and presents it to the output port, but the data at the output port is not presented to the **q** pins until the next rising clock edge.

Assignment 3 (10 Points)

Design a logic diagram to implement 3-8 decoder with an enable input.

1. Derive the decoder's truth table.
2. Design the logic diagram to implement the decoder.
3. Simulate the circuit and check the result with the truth table.

Assignment 4 (10 Points)

Design a logic diagram to implement 1-bit 8-1 multiplexer.

1. Design the logic diagram to implement the multiplexer.
2. Simulate the circuit and check the result.

Assignment 5 (10 Points)

Design a logic diagram to implement 8-bit 8-1 multiplexer.

1. Design the logic diagram to implement the multiplexer.
2. Simulate the circuit and check the result.

Assignment 6 (20 Points)

Design a logic diagram to implement an 8-bit parallel load register with enable input.

1. Design the logic diagram of the register.
2. Simulate the circuit and check the result.

Assignment 7 (10 Points)

Design a logic diagram of a 32 x 8 RAM using two 32 x 4 RAM chips.

1. Design the logic diagram.
2. Simulate the circuit and check the result.

Assignment 8 (20 Points)

Design a logic diagram of a 64 x 4 RAM using two 32 x 4 RAM chips.

1. Design the logic diagram.
2. Simulate the circuit and check the result.