

HDL Memory – ROM MW

Last updated 4/14/20

HDL Memory – ROM MW

- ROM HDL
 - Quartus has a series of pre-defined blocks
 - They can be created in Quartus - **MegaWizard**
 - They need to be instantiated in our design

HDL Memory – ROM MW

- ROM HDL - MegaWizard

The screenshot displays the Quartus Prime Lite Edition interface. The main window shows the 'Compilation Report - memories' with the 'Flow Summary' tab selected. The 'IP Catalog' on the right side is open, and the 'ROM: 1-PORT' entry is highlighted with a red circle. The 'Messages' window at the bottom shows the compilation output, including a warning about the signal 'myROM' and a message about the number of processors.

Compilation Report - memories

Table of Contents	Flow Summary
Flow Summary	Flow Status: Successful - Thu Apr 26 10:19:06 2018
Flow Settings	Quartus Prime Version: 16.1.0 Build 196 10/24/2016 SJ Lite Editio
Flow Non-Default Global Sett	Revision Name: memories
Flow Elapsed Time	Top-level Entity Name: rom_sync_16B_wfile
Flow OS Summary	Family: MAX 10
Flow Log	Device: 10M50DAF484C7G
Analysis & Elaboration	Timing Models: Final
Flow Messages	Total logic elements: N/A until Partition Merge
Flow Suppressed Messages	Total registers: N/A until Partition Merge
	Total pins: N/A until Partition Merge
	Total virtual pins: N/A until Partition Merge
	Total memory bits: N/A until Partition Merge
	Embedded Multiplier 9-bit elements: N/A until Partition Merge
	Total PLLs: N/A until Partition Merge

IP Catalog

- > Arithmetic
- > Bridges and Adaptors
- > Clocks; PLLs and Resets
- > Configuration and Programming
- > I/O
- > Miscellaneous
- > On Chip Memory
 - Altera On-Chip Flash
 - FIFO
 - RAM: 1-PORT
 - RAM: 2-PORT**
 - ROM: 1-PORT**
 - ROM: 2-PORT
 - Shift register (RAM-based)

Messages

```
System (6) Processing (21)
> 12021 Found 2 design units, including 1 entities, in source file rom_sync_16b_lpm_wfile_tb.vhdl
> 12127 Elaborating entity "rom_sync_16B_wfile" for the top level hierarchy
> 10541 VHDL Signal Declaration warning at rom_sync_16B_wfile.vhdl(36): used implicit default value for signal "myROM" because signal was never ass
> Quartus Prime Analysis & Elaboration was successful. 0 errors, 2 warnings
> Running Quartus Prime Netlist Viewers Preprocess
> Command: quartus_npp memories -c memories --netlist_type=sgate
> 18236 Number of processors has not been specified which may cause overloading on shared machines. Set the global assignment NUM_PARALLEL_PROCESS
> Quartus Prime Netlist Viewers Preprocess was successful. 0 errors, 1 warning
```

HDL Memory – ROM MW

- ROM HDL – MegaWizard

MegaWizard Plug-In Manager [page 1 of 5]

ROM: 1-PORT

Parameter Settings | EDA | Summary

General | Regs/Clken/Aclrs | Mem Init

Currently selected device family: MAX 10

Match project/default

How wide should the 'q' output bus be? 8 bits

How many 8-bit words of memory? 32 words

Note: You could enter arbitrary values for width and depth

What should the memory block type be?

Auto MLAB M9K
 M144K LCs

Set the maximum block depth to Auto words

What clocking method would you like to use?

Single clock Dual clock: use separate 'input' and 'output' clocks

Resource Usage: 1 M9K

Cancel < Back Next > Finish

MegaWizard Plug-In Manager [page 2 of 5]

ROM: 1-PORT

Parameter Settings | EDA | Summary

General | Regs/Clken/Aclrs | Mem Init

Which ports should be registered?

'data' input port

'address' input port

'q' output port

Create one clock enable signal for each clock signal.
Note: All registered ports are controlled by the enable signal(s)

Create byte enable for port A

What is the width of a byte for byte enables? 8 bits

Create an 'aclr' asynchronous clear for the registered ports

Create a 'rden' read enable signal

Resource Usage: 1 M9K

Cancel < Back Next > Finish

HDL Memory – ROM MW

- ROM HDL - MegaWizard

MegaWizard Plug-In Manager [page 3 of 5]

ROM: 1-PORT

Parameter Settings | EDA | Summary

General > Regs/Cken/Aclrs > Mem Init

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:

The initial content dimensions should conform to which port's dimensions?

Allow In-System Memory Content Editor to capture and update content independently of the system clock

The 'Instance ID' of this ROM is:

Resource Usage: 1 M9K

Cancel < Back Next > Finish

MegaWizard Plug-In Manager [page 4 of 5]

ROM: 1-PORT

Parameter Settings | EDA | 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.

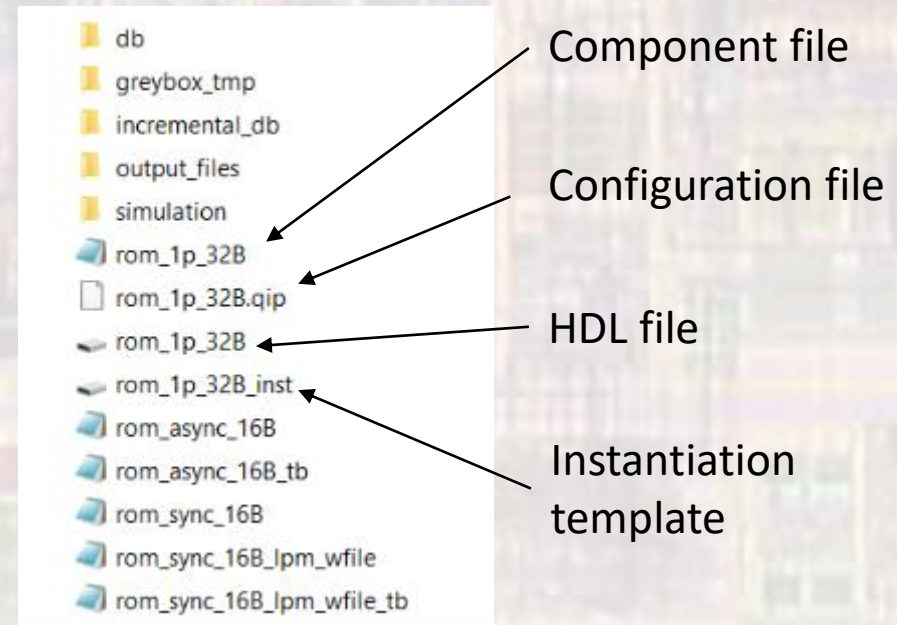
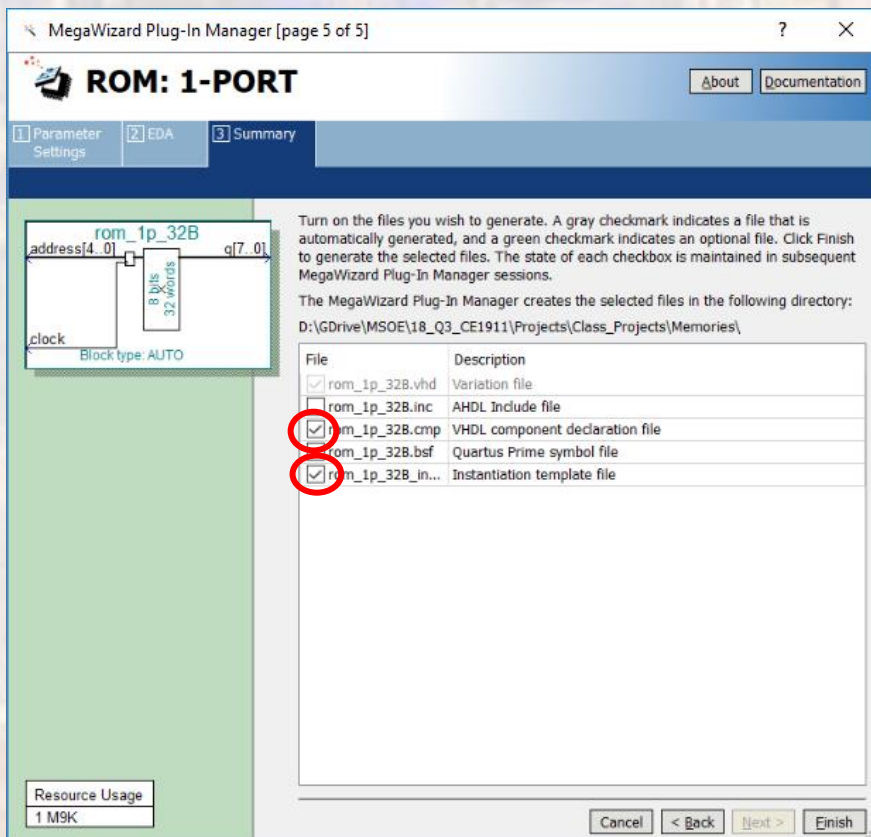
Generate netlist

Resource Usage: 1 M9K

Cancel < Back Next > Finish

HDL Memory – ROM MW

- ROM HDL - MegaWizard



HDL Memory – ROM MW

- ROM HDL - MegaWizard

System created HDL file

```
LIBRARY ieee;
USE ieee.std_logic_1164.all;

LIBRARY altera_mf;
USE altera_mf.altera_mf_components.all;

ENTITY rom_1p_32B IS
  PORT
  (
    address      : IN STD_LOGIC_VECTOR (4 DOWNTO 0);
    clock        : IN STD_LOGIC := '1';
    q            : OUT STD_LOGIC_VECTOR (7 DOWNTO 0)
  );
END rom_1p_32B;

ARCHITECTURE SYN OF rom_1p_32b IS

  SIGNAL sub_wire0 : STD_LOGIC_VECTOR (7 DOWNTO 0);
```

```
BEGIN
  q <= sub_wire0(7 DOWNTO 0);

  altsyncram_component : altsyncram
  GENERIC MAP (
    address_aclr_a => "NONE",
    clock_enable_input_a => "BYPASS",
    clock_enable_output_a => "BYPASS",
    init_file => "rom_init.mif",
    intended_device_family => "MAX 10",
    lpm_hint => "ENABLE_RUNTIME_MOD=NO",
    lpm_type => "altsyncram",
    numwords_a => 32,
    operation_mode => "ROM",
    outdata_aclr_a => "NONE",
    outdata_reg_a => "UNREGISTERED",
    widthad_a => 5,
    width_a => 8,
    width_byteena_a => 1
  )
  PORT MAP (
    address_a => address,
    clock0 => clock,
    q_a => sub_wire0
  );

END SYN;
```

HDL Memory – ROM MW

- ROM HDL - MegaWizard

Component file

```
component rom_1p_32B
  PORT
  (
    address      : IN STD_LOGIC_VECTOR (4 DOWNTO 0);
    clock        : IN STD_LOGIC := '1';
    q            : OUT STD_LOGIC_VECTOR (7 DOWNTO 0)
  );
end component;
```

Instantiation template

```
rom_1p_32B_inst : rom_1p_32B PORT MAP (
  address => address_sig,
  clock   => clock_sig,
  q       => q_sig
);
```


HDL Memory – ROM MW

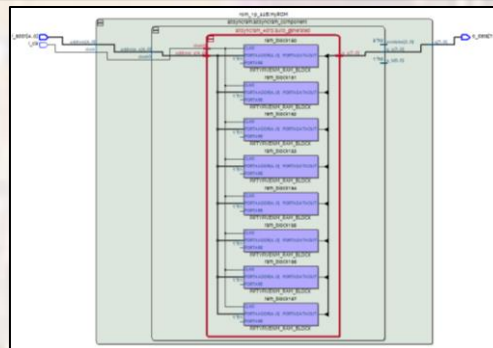
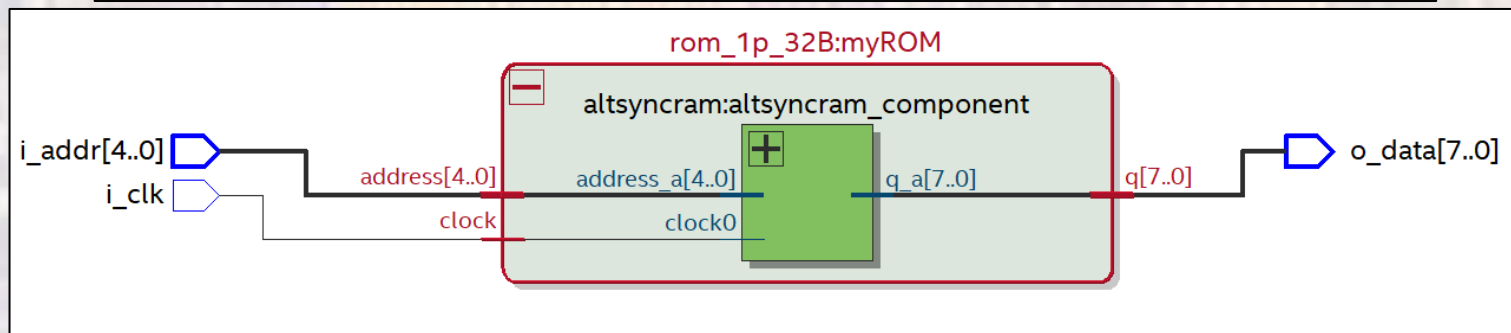
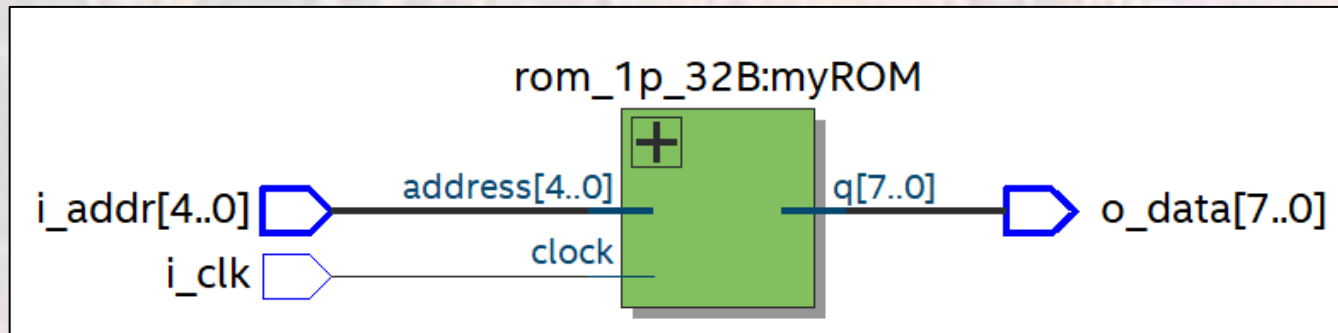
- ROM HDL - MegaWizard

```
-----  
--  
-- rom_sync_32B_MW_wfile.vhd1  
--  
-- created 4/25/17  
-- tj  
--  
-- rev 0  
-----  
--  
-- 32B synchronous ROM from MegaWizard loaded from file  
--  
-----  
--  
-- Inputs:  clk, addr  
-- Outputs: data  
--  
-----  
library ieee;  
use ieee.std_logic_1164.all;  
use ieee.numeric_std.all;  
  
entity rom_sync_32B_MW_wfile is  
port(  
    i_clk:    in std_logic;  
    i_addr:   in std_logic_vector(4 downto 0);  
    o_data:   out std_logic_vector(7 downto 0)  
);  
end;
```

```
architecture behavioral of rom_sync_32B_MW_wfile is  
    --  
    -- no signals  
    --  
    component rom_1p_32B  
        PORT  
        (  
            address      : IN STD_LOGIC_VECTOR (4 DOWNTO 0);  
            clock         : IN STD_LOGIC      := '1';  
            q              : OUT STD_LOGIC_VECTOR (7 DOWNTO 0)  
        );  
    end component;  
  
begin  
  
    myROM: rom_1p_32B PORT MAP (  
        address => i_addr,  
        clock   => i_clk,  
        q       => o_data  
    );  
  
    --  
    -- Output logic  
    --  
end behavioral;
```

HDL Memory – ROM MW

- ROM HDL - MegaWizard



HDL Memory – ROM MW

- ROM HDL - testbench

```
-----  
--  
-- rom_sync_32B_MW_wfile_tb.vhdl  
--  
-- created: 4/10/18  
-- by: johnsontimoj  
-- rev: 0  
--  
-- testbench for sync 32B Megawizard from file example  
-- of rom_sync_32B_MW_wfile.vhdl  
-----  
library ieee;  
use ieee.std_logic_1164.all;  
use ieee.numeric_std.all;  
  
entity rom_sync_32B_MW_wfile_tb is  
  -- no entry - testbench  
end entity;  
  
architecture testbench of rom_sync_32B_MW_wfile_tb is  
  
  signal CLK      : std_logic;  
  signal ADDR     : std_logic_vector(4 downto 0);  
  
  signal DATA    : std_logic_vector(7 downto 0);  
  
  constant PER    : time := 20 ns;  
  
  -----  
  -- Component prototype  
  -----  
  COMPONENT rom_sync_32B_MW_wfile  
  port(  
    i_clk:  in std_logic;  
    i_addr: in std_logic_vector(4 downto 0);  
    o_data: out std_logic_vector(7 downto 0)  
  );  
END COMPONENT;
```

```
begin  
  
-----  
-- Device under test (DUT)  
-----  
DUT: rom_sync_32B_MW_wfile  
  port map(  
    i_clk  => CLK,  
    i_addr => ADDR,  
    o_data => DATA  
  );  
  
-----  
-- Test processes  
-----  
  
-- Clock process  
clock: process -- note - no sensitivity list allowed  
begin  
  CLK <= '0';  
  wait for PER/2;  
  infinite: loop  
    CLK <= not CLK; wait for PER/2;  
  end loop;  
end process;  
  
-- Reset process  
--no reset  
  
-- Run Process  
run: Process -- note - no sensitivity list allowed  
begin  
  -- Initialize values  
  ADDR <= (others => '0');  
  -- test all values  
  for i in 0 to 15 loop  
    wait for 2*PER;  
    ADDR <= std_logic_vector(to_unsigned(i,5));  
  end loop;  
end process run;  
  
-----  
-- End test processes  
-----  
  
end architecture;
```

HDL Memory – ROM MW

- ROM HDL - MegaWizard

