HDL Memory – ROM MW

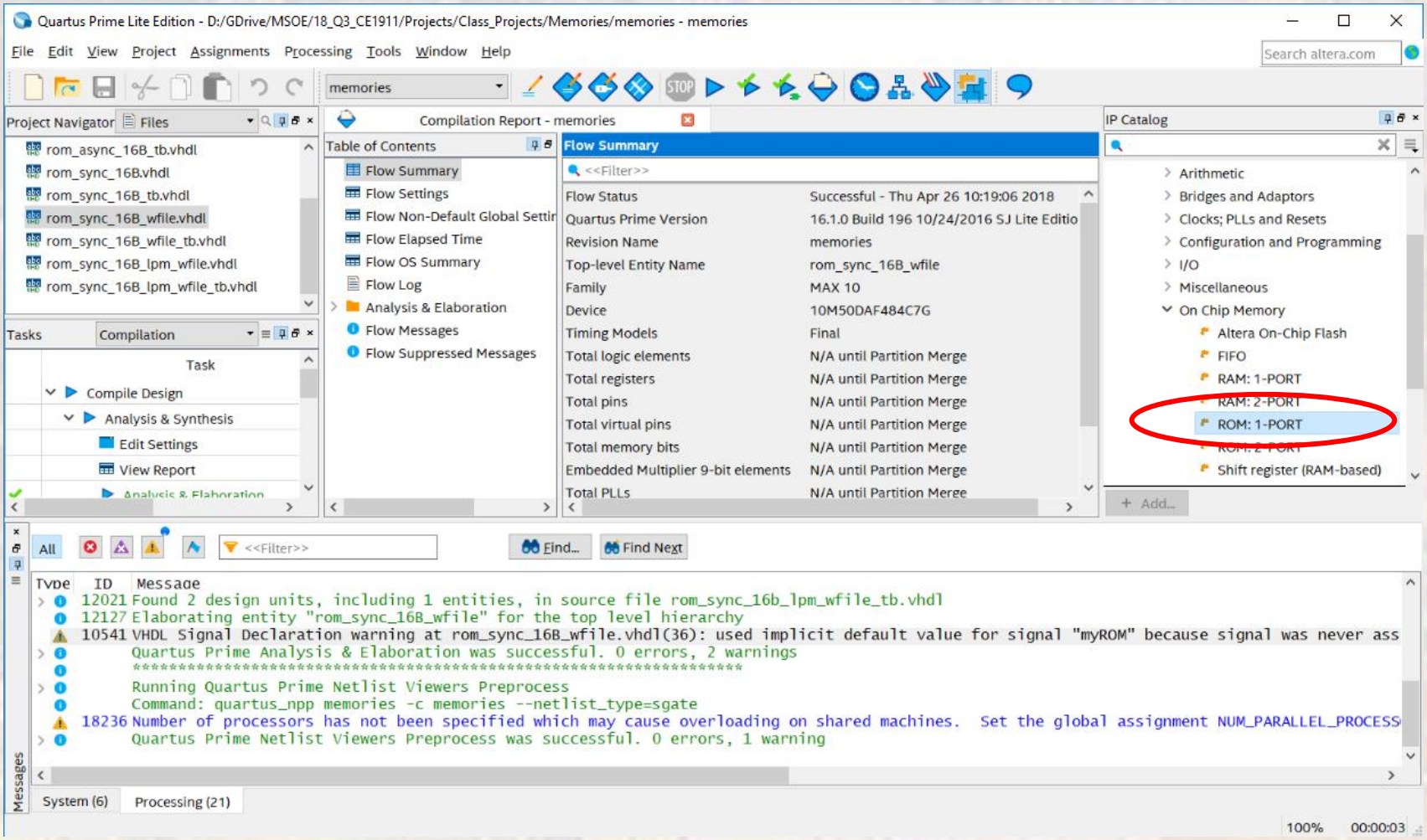
Last updated 4/14/20

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

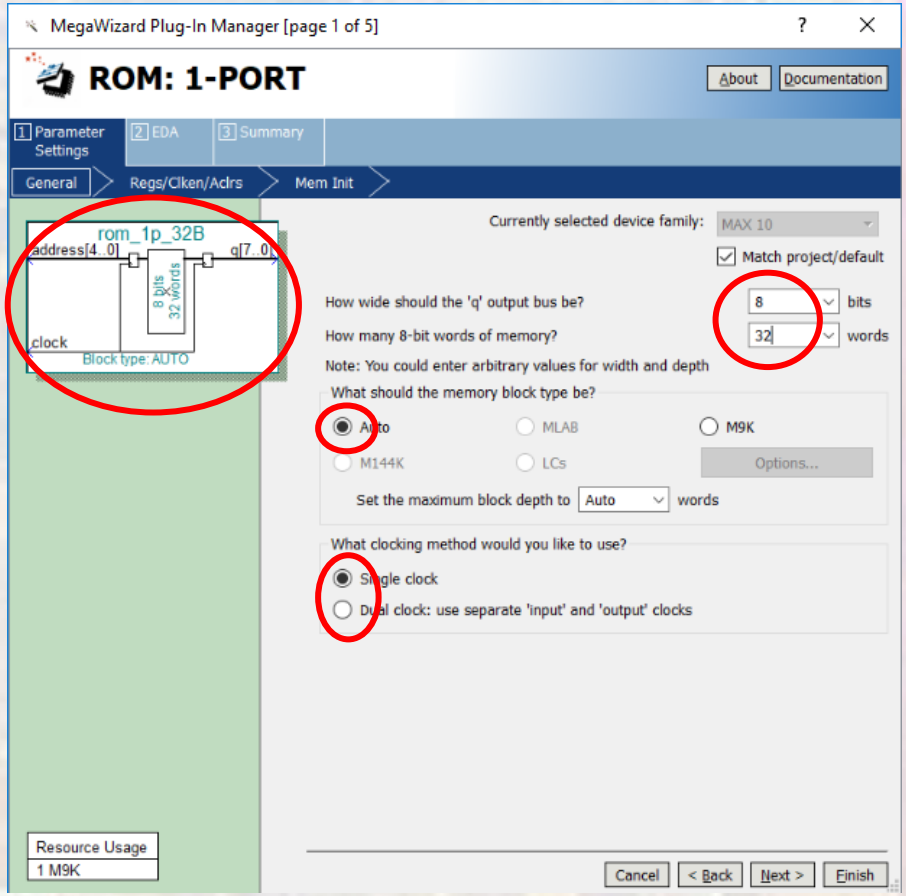
HDL Memory – ROM MW

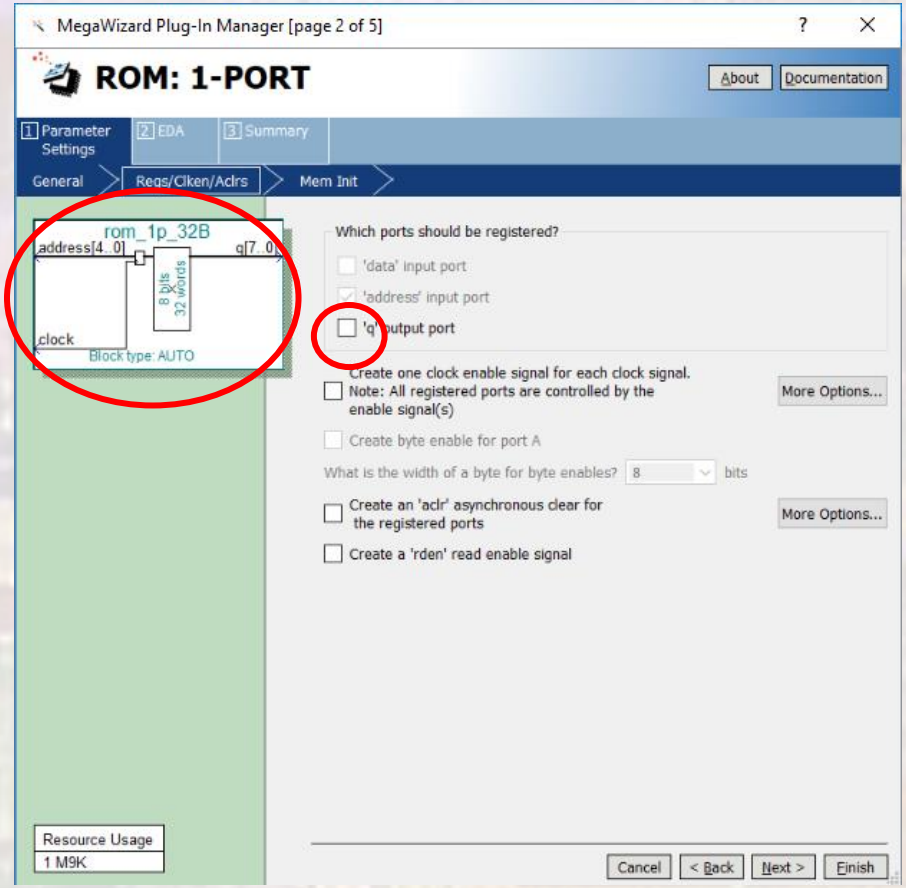
- ROM HDL - MegaWizard



HDL Memory – ROM MW

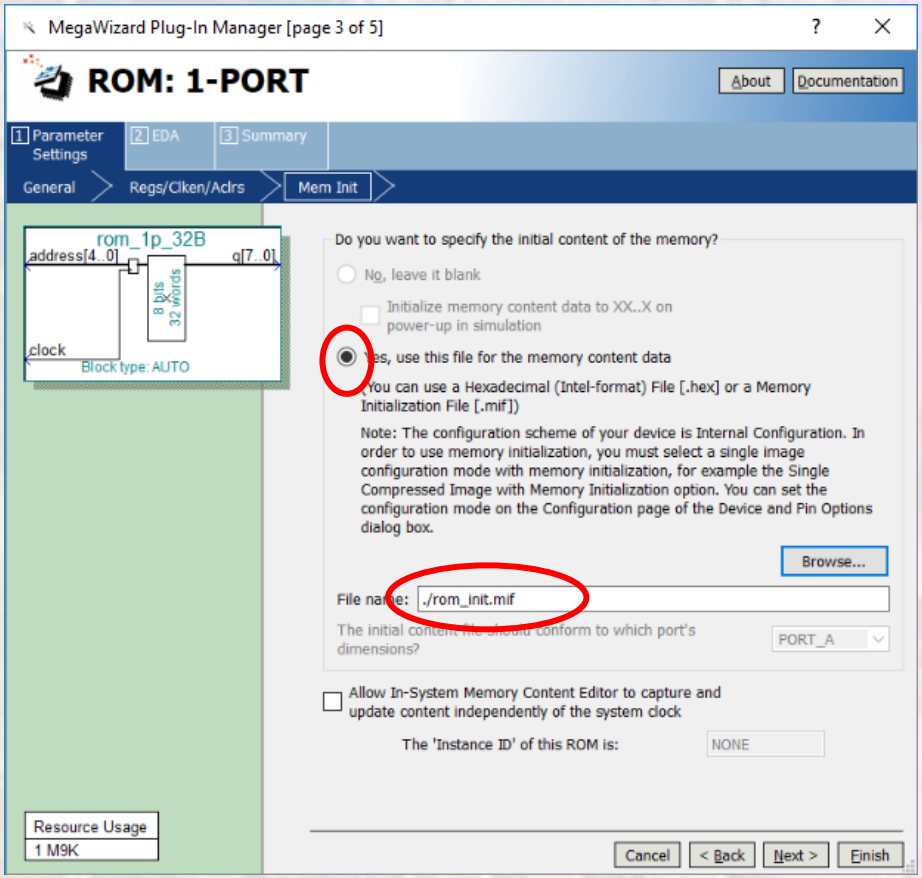
- ROM HDL – MegaWizard

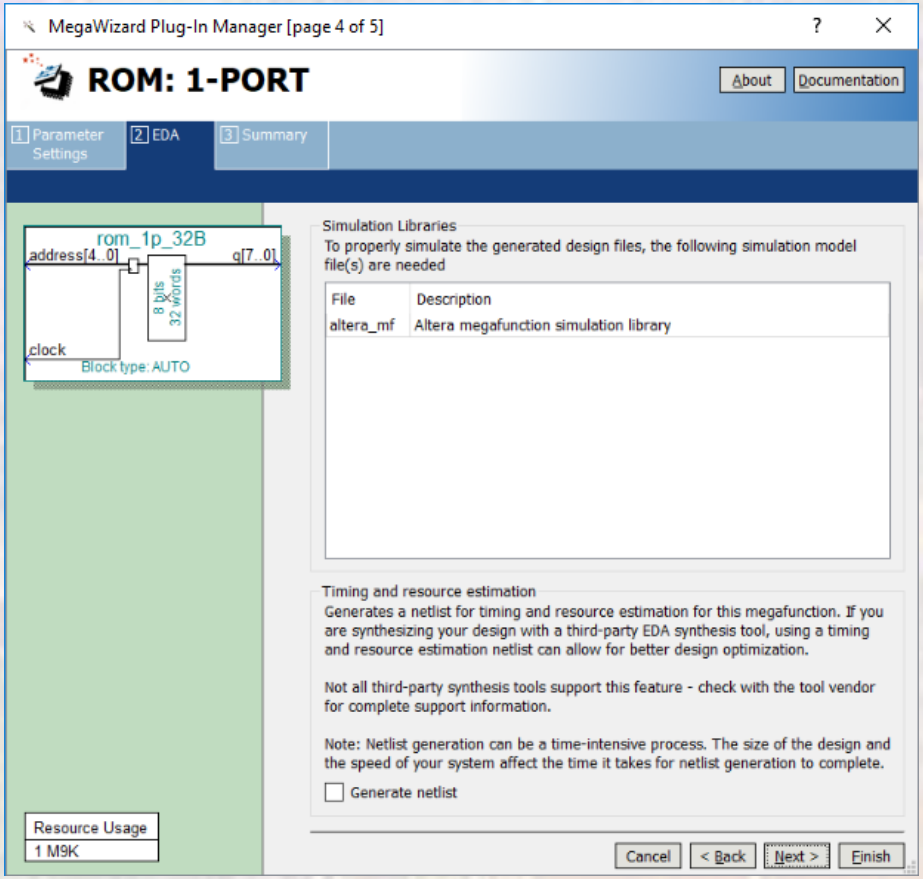




HDL Memory – ROM MW

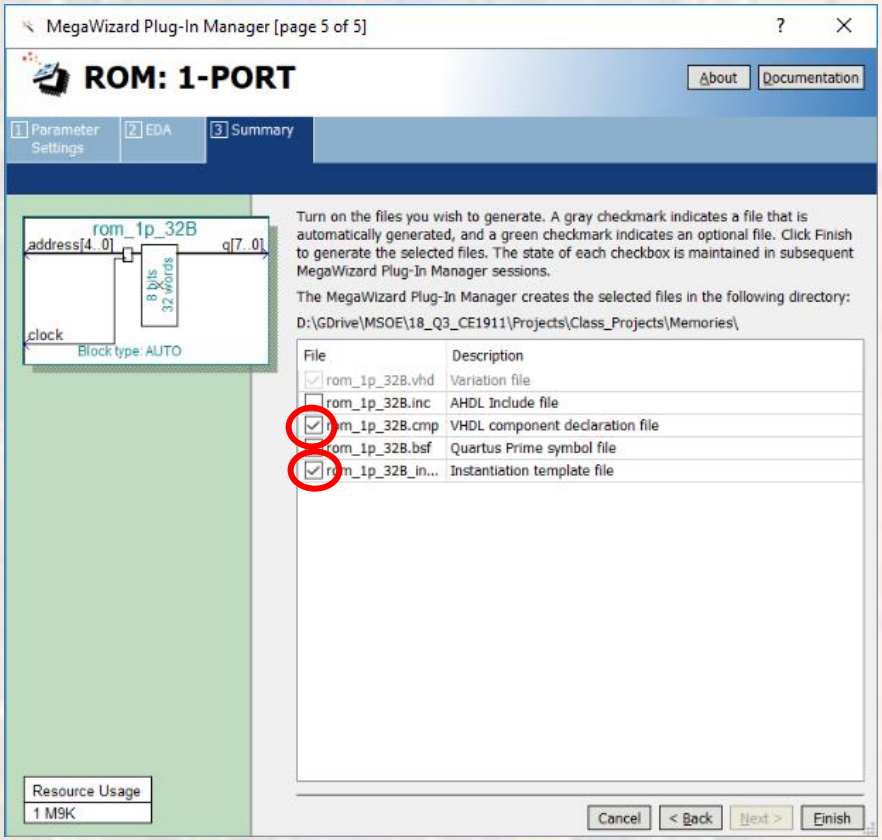
- ROM HDL - MegaWizard

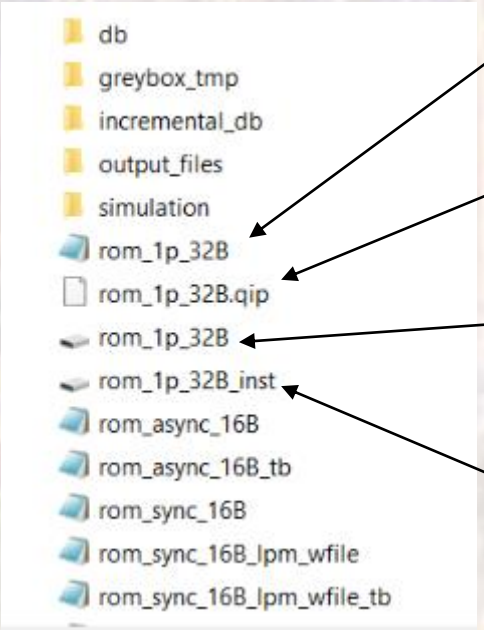




HDL Memory – ROM MW

- ROM HDL - MegaWizard





Component file

Configuration file

HDL file

Instantiation template

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

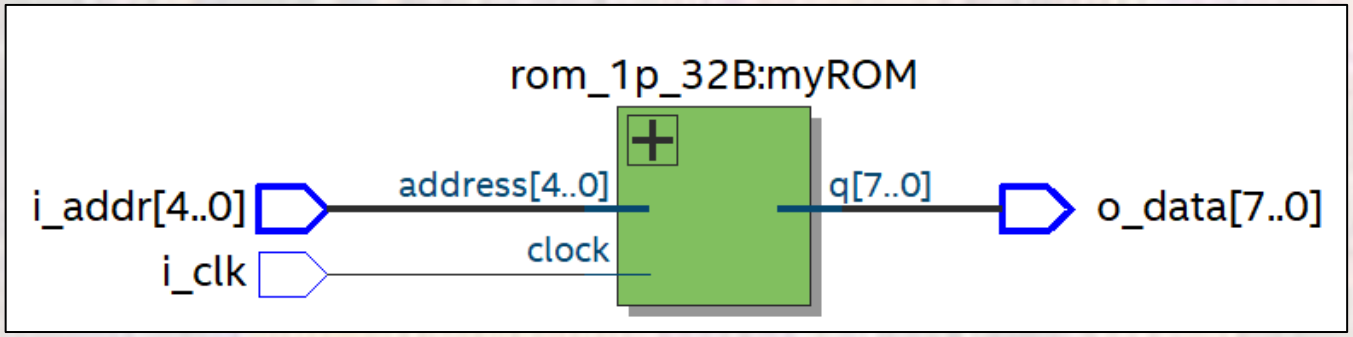
- ROM HDL - MegaWizard

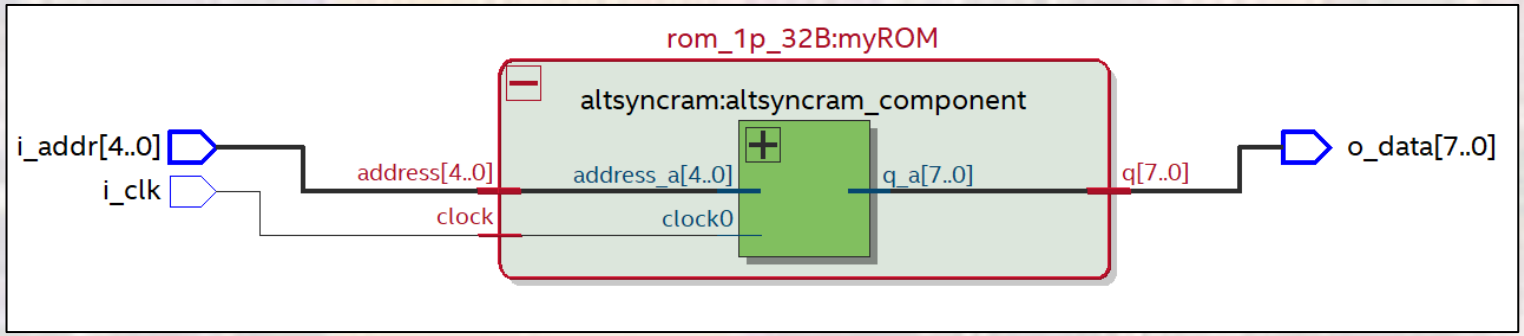
```
---------------------------------------
--
-- rom_sync_32B_MW_wfile.vhdl
--
-- created 4/25/17
-- tj
--
-- rev 0
---------------------------------------
--
-- 32B synchronous ROM from MegaWizard loaded from file
--
---------------------------------------
--
-- Inputs:  clk, addr
-- Outputs: data
--
---------------------------------------
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;
```

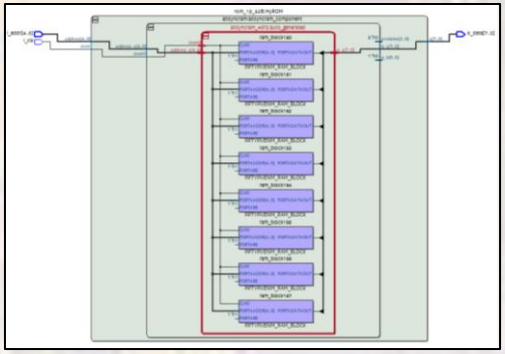
```
architecture behavioral of rom_sync_32B_MW_wfile is
   --
   -- no signals
   --
component rom_1p_32B
   PORT
   (
      address     : IN STD_LOGIC_VECTOR (4 DOWNTO 0);
      clock       : IN STD_LOGIC  := '1';
      q           : OUT STD_LOGIC_VECTOR (7 DOWNTO 0)
   );
end component;

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
      -- Initalize 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