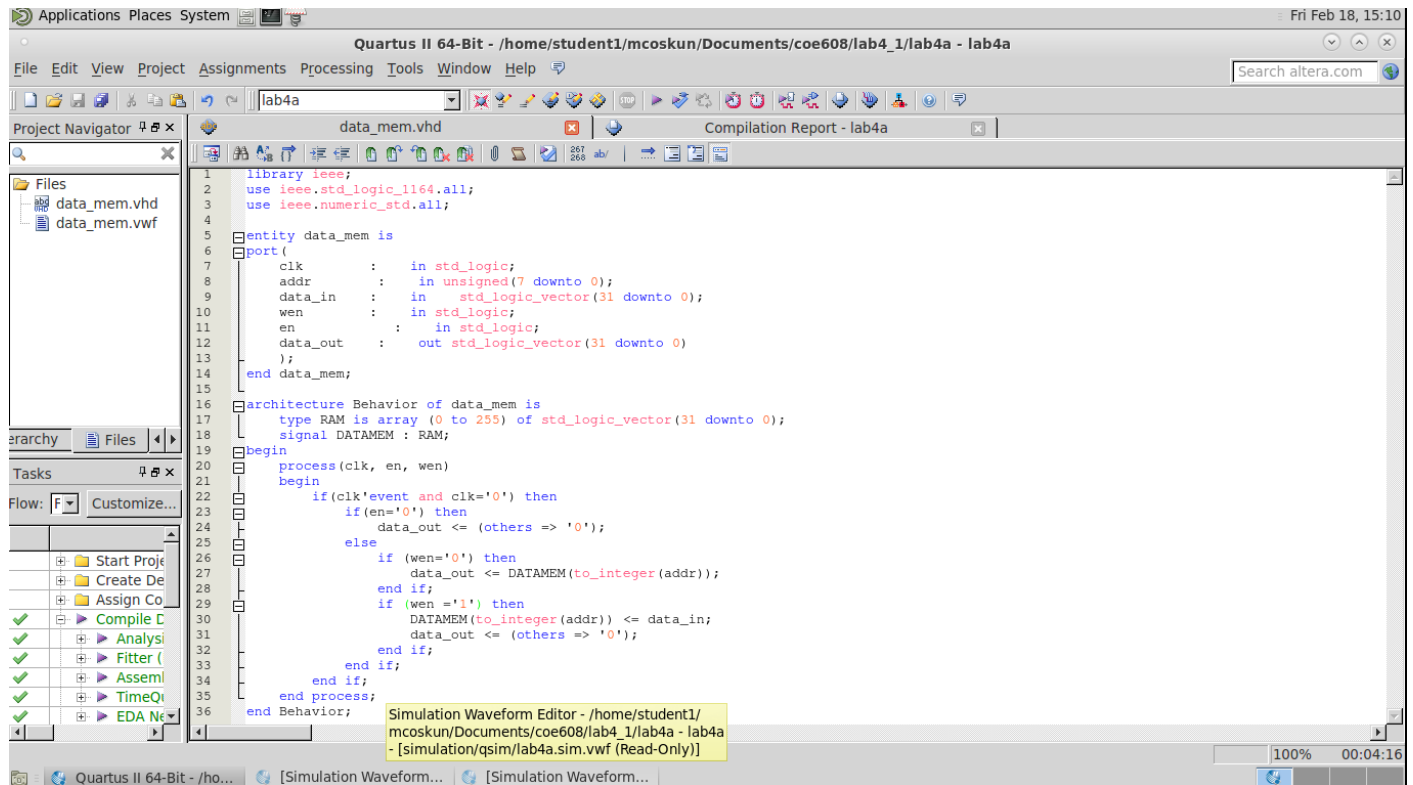


In this lab, we will be implementing a simple memory unit that can read and write within a clock cycle. The unit consists of 8 bit-address input (addr), 32 bit-data input and output (data\_in and data\_out), enable(en), write enable(wen), and clock(clk).

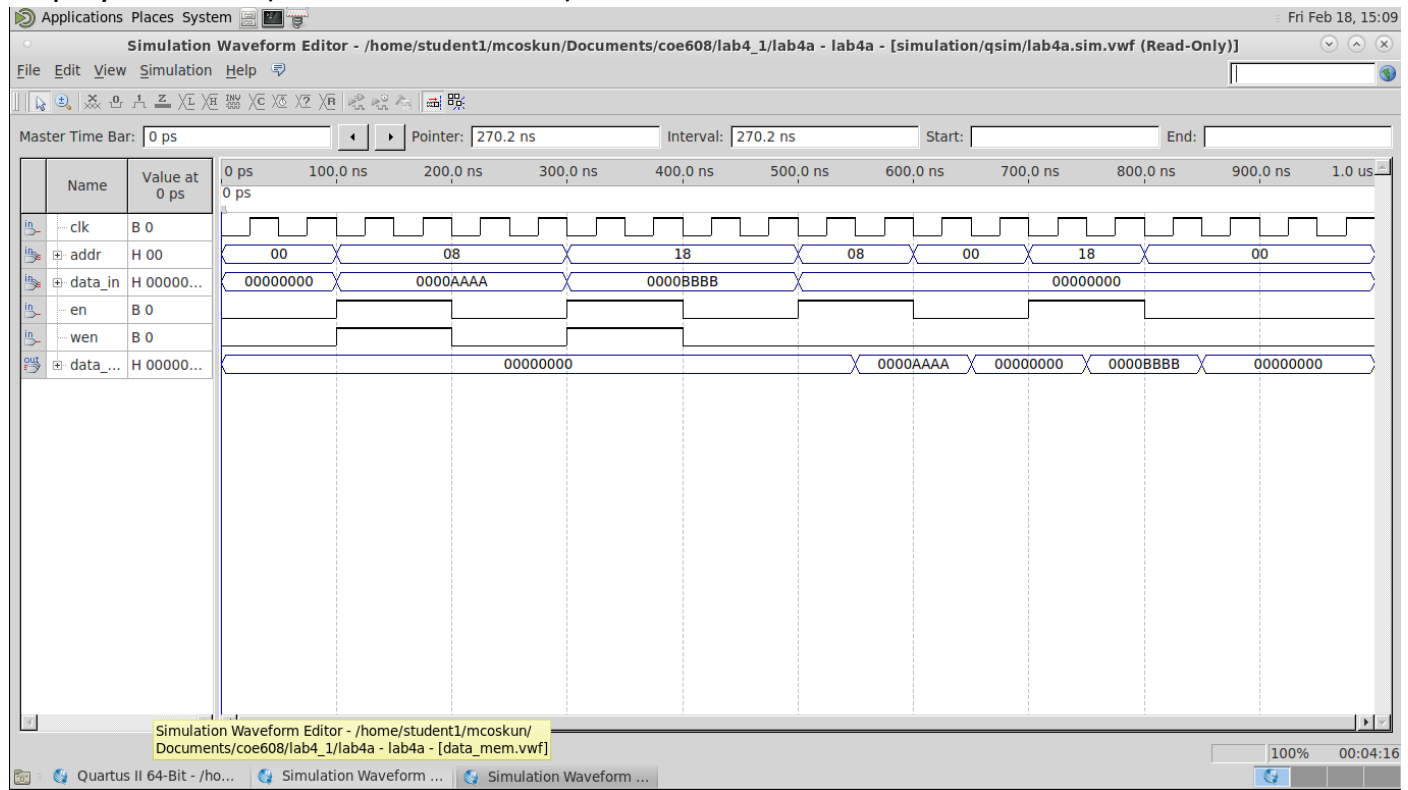
The VHDL code of the unit can be seen below.



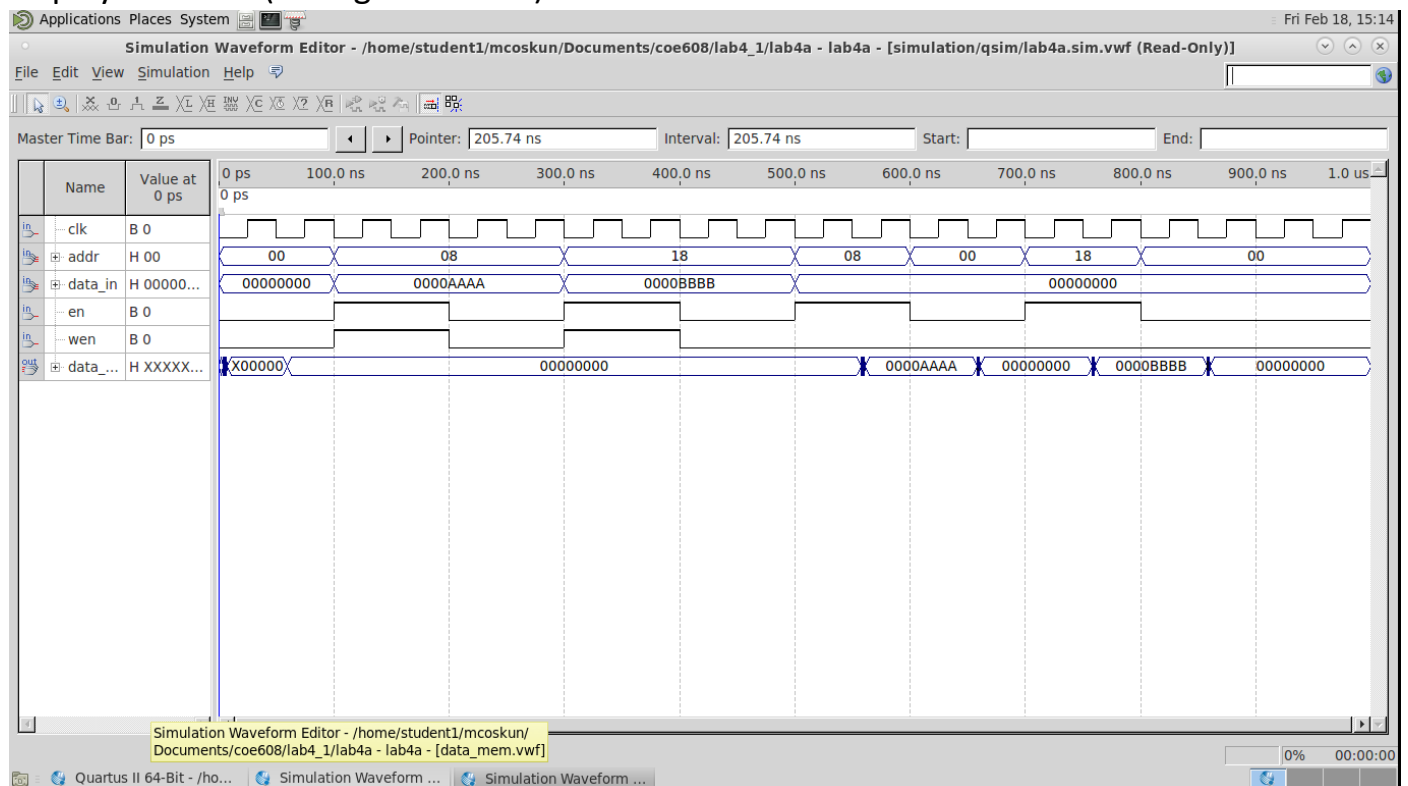
```
1 library ieee;
2 use ieee.std_logic_1164.all;
3 use ieee.numeric_std.all;
4
5 entity data_mem is
6 port (
7     clk      : in std_logic;
8     addr     : in unsigned(7 downto 0);
9     data_in  : in std_logic_vector(31 downto 0);
10    wen      : in std_logic;
11    en       : in std_logic;
12    data_out  : out std_logic_vector(31 downto 0);
13 );
14 end data_mem;
15
16 architecture Behavior of data_mem is
17     type RAM is array (0 to 255) of std_logic_vector(31 downto 0);
18     signal DATAMEM : RAM;
19 begin
20     process(clk, en, wen)
21     begin
22         if (clk'event and clk='0') then
23             if (en='0') then
24                 data_out <= (others => '0');
25             else
26                 if (wen='0') then
27                     data_out <= DATAMEM(to_integer(addr));
28                 end if;
29                 if (wen='1') then
30                     DATAMEM(to_integer(addr)) <= data_in;
31                     data_out <= (others => '0');
32                 end if;
33             end if;
34         end if;
35     end process;
36 end Behavior;
```

Simulation Waveform Editor - /home/student1/mcoskun/Documents/coe608/lab4\_1/lab4a - lab4a  
- [simulation/qsim/lab4a.sim.vwf (Read-Only)]

After compiling successfully, the waveform of the unit created following the lab manual is displayed below (function waveform).



After compiling successfully, the waveform of the unit created following the lab manual is displayed below (timing waveform).



In order to find the worst-case delays for reads and writes of various inputs, we refer to the compilation report of lab4a, "TimeQuest Timing Analyzer>Multicorner Datasheet>Clock to Output Times" section. The average of data\_out[\*] Rise and Fall gives us an approximate for the worst-case scenario delays. In our case, it is 10.0065.