

گزارشکار آزمایش ۷ معماری کامپیوتر

محمد مهدی نظری ۹۹۳۱۰۶۱ – آرمین ابراهیمی صبا ۹۹۳۱۰۸۶

گزارشکار این آزمایش از دوبرخش تشکیل شده که حافظه ROM در این بخش و حافظه RAM در بخش دیگر توضیح داده میشوند.

برای پر کردن اولیه خانه های رام از الگوی گری استفاده شده و سایز حافظه ۱۶ تا خانه با گنجایش ۸ بیت در نظر گرفته شده که باتوجه به استفاده از generic قابل تغییر هستند.

```
33
34 entity ROM is
35     generic(
36         D : integer := 16; -- tedade khane hafeze
37         C : integer := 4; -- logD
38         W : integer := 8 -- toole har block
39     );
40 port(
41     addr : in std_logic_vector(C-1 downto 0);
42     data : out std_logic_vector(W-1 downto 0);
43     Clk : in std_logic
44 );
45 end ROM;
46
47 architecture Behavioral of ROM is
48
49     type rom_type is array (0 to D-1) of std_logic_vector(W-1 downto 0);
50
51     -- gray code
52     signal RD : rom_type := (
53         "00000000", -- 0
54         "00000001", -- 1
55         "00000011", -- 2
56         "00000010", -- 3
57         "00000110", -- 4
58         "00000111", -- 5
59         "00000101", -- 6
60         "00000100", -- 7
61         "00001100", -- 8
62         "00001101", -- 9
63         "00001111", -- 10
64         "00001110", -- 11
65         "00001010", -- 12
66         "00001011", -- 13
67         "00001001", -- 14
68         "00001000" -- 15
```

```
70 begin
71
72     process(Clk, RD)
73     begin
74         if (rising_edge(Clk)) then
75             data <= RD(conv_integer(addr));
76         end if ;
77     end process ;
78
79 end Behavioral;
80
```

گزارشکار آزمایش ۷ معماری کامپیوتر

تست پنچ :

```

32 entity ROM_tb is
33 end ROM_tb;
34
35 architecture test of ROM_tb is
36 component ROM is
37 port(
38     addr : in std_logic_vector(3 downto 0);
39     data : out std_logic_vector(7 downto 0);
40     Clk :in std_logic
41 );
42 end component;
43
44 signal addr : std_logic_vector(3 downto 0) := "0000" ;
45 signal data : std_logic_vector(7 downto 0) ;
46 signal clk : std_logic := '0';
47
48 begin
49     clk_process :process
50     begin
51         clk <= '0';
52         wait for 5 ns;
53         clk <= '1';
54         wait for 5 ns;
55     end process;
56
57     rom_example : ROM port map (
58         addr => addr ,
59         data => data ,
60         clk => clk
61     );
62
63

```

```

64 process
65 begin
66
67     addr <= "0000" ;
68
69     wait for 20 ns ;
70
71     addr <= "0001" ;
72
73     wait for 20 ns ;
74
75     addr <= "0010" ;
76
77     wait for 20 ns ;
78
79     addr <= "0100" ;
80
81     wait for 20 ns ;
82
83     addr <= "1000" ;
84
85     wait for 20 ns ;
86
87 end process ;
88
89 end test;

```

