



طراحی سیستم های دیجیتال

دکتر حاکم بیت اللهی

---

پاسخنامه تمرین سری اول

---

آبتین بدیعی - تارا ستاره

## سوال اول:

(A)

چهار نوع port mode داریم:

In: مقدار قرار گرفته درون پورت در این حالت تنها قابل خواندن است و همیشه در آن نوشت.

Out: این حالت برعکس حالت اول تنها قابل نوشتن است و غیرقابل خواندن از آن است.

Inout: این حالت هم قابل نوشتن است و هم قابل خواندن

Buffer: مانند inout است ولی تنها از یک source مقدار دهی میشود.

(B)

478 به صورت باینری : 111011110

478 به صورت اوکتال: 736

478 به صورت هگزادسیمال : 1DE

(C)

از identifiers ها جهت نام گذاری آیتم ها استفاده میشود.

- نمی توان همزمان از دو Underline در آنها استفاده کرد.
- نمی توانند با Underline تمام شوند.
- میتواند شامل حروف الفبا (A-Z) یا (a-z) میباشد و یا شامل اعداد (0-9) و یا underline میباشد.
- حرف اول آنها باید از حروف تشکیل شده باشد.

(D)

- طریقه مقدار دهی در آنها متفاوت است:  

$$\text{variable} := \text{signal} \leq$$
- Signal ها به صورت global تعریف میشوند ولی variable ها به صورت local
- Variable ها درون process تعریف میشود ولی signal ها درون architecture ها و قبل از begin
- مقدار اختصاص یافته به variable ها به صورت به صورت لحظه ای و بدون هیچ گونه تاخیر میباشد ولی مقدار اختصاصی به signal ها با تاخیر همراه میباشد.
- برای اجرای برنامه به صورت موازی از signal استفاده میشود ولی برای اجرای برنامه به صورت همروند از variable استفاده میشود.

(E)

- با کمک generics ها میتوان پارامتر های entity را مشخص کرد.
- زمانی کاربرد است که بخواهیم چند نمونه از یک component ایجاد نماییم. و همچنین میتواند در هر نمونه مقدار متفاوت خود را داشته باشد.
- همچنین باید اشاره کرد که قابلیت Parameter Passing را نیز دارد.

(F

```
entity Example is
  Port ( input_signal : in STD_LOGIC;
        output_signal : out STD_LOGIC
        );
end entity Example;

architecture Behavioral of Example is
  signal internal_signal : STD_LOGIC;
begin
  Process
  begin
    if input_signal = '1' then
      internal_signal <= '1';
    else
      internal_signal <= '0';
    end if;
  end process;

  output_signal <= internal_signal;
end Behavioral;
```

## سوال دوم:

```

library IEEE;
use IEEE.STD_LOGIC_1164.all;

entity Comparator is
    port (
        a, b: in std_logic_vector(3 downto 0);
        gt, eq, lt: out std_logic
    );
end Comparator;

architecture behavioral of Comparator is
begin
    gt <= '1' when (a > b) else '0';
    eq <= '1' when (a = b) else '0';
    lt <= '1' when (a < b) else '0';
end behavioral;

```

## سوال سوم:

```

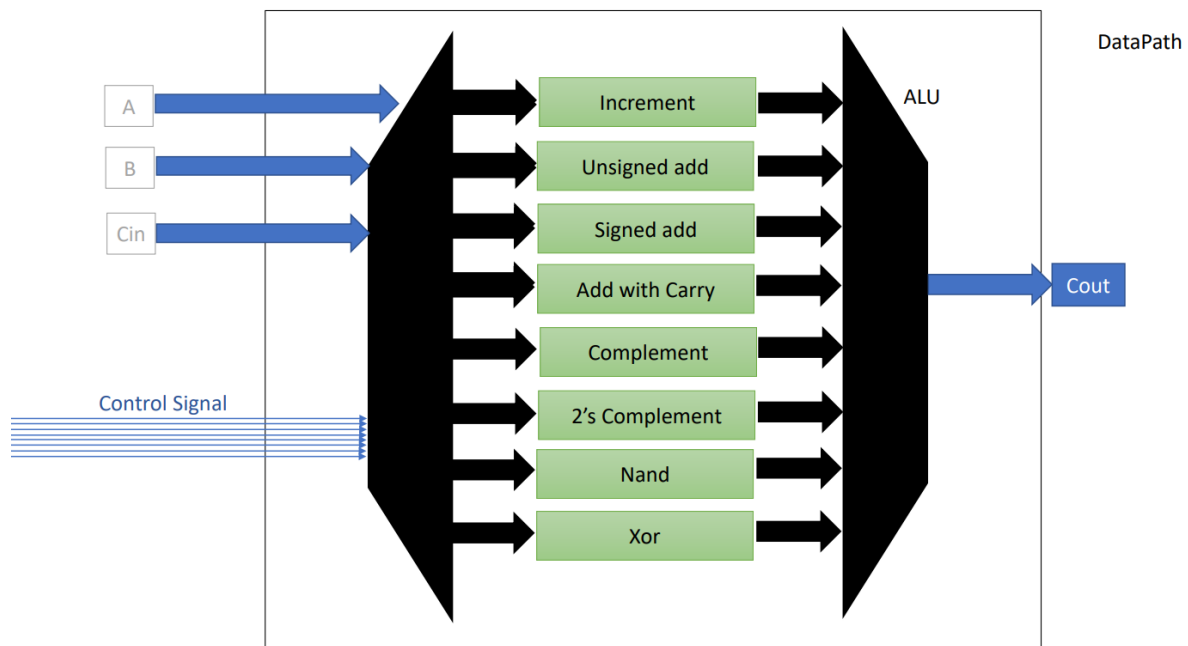
library ieee;
use ieee.std_logic_1164.all;
use ieee.std_logic_unsigned.all;
use ieee.numeric_std.all;

entity ALU is
    port (
        a: in std_logic_vector (7 downto 0);
        b: in std_logic_vector (7 downto 0);
        sel: in std_logic_vector (2 downto 0);
        y: out std_logic_vector (7 downto 0);
        cin: in std_logic
    );
end ALU;

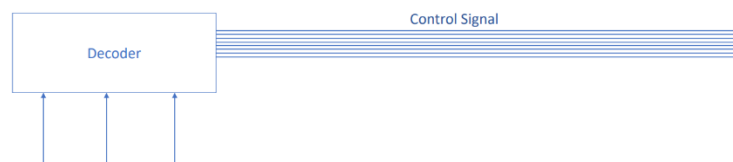
architecture behavioral of ALU is
begin
    Y <= a + '1' when sel = "000" else
        std_logic_vector(unsigned(a) + unsigned(b)) when sel = "001" else
        std_logic_vector(signed(a) + signed(b)) when sel = "010" else
        a + b + cin when sel = "011" else
        (not a) when sel = "100" else
        ((not a) + '1') when sel = "101" else
        a nand b when sel = "110" else
        a xor b when sel = "111" else
        "00000000";

end behavioral;

```



#### Control Unit



## سوال چهارم:

A)

```

ENTITY priority_encoder IS
    PORT (x: IN BIT_VECTOR (7 DOWNTO 1);
          y: OUT BIT_VECTOR (2 DOWNTO 0));
END priority_encoder;

ARCHITECTURE encoder OF priority_encoder IS
BEGIN
    y <= "111" WHEN x(7)='1' ELSE
        "110" WHEN x(6)='1' ELSE
        "101" WHEN x(5)='1' ELSE
        "100" WHEN x(4)='1' ELSE
        "011" WHEN x(3)='1' ELSE
        "010" WHEN x(2)='1' ELSE
        "001" WHEN x(1)='1' ELSE
        "000";
END encoder;

```

B)

```

ENTITY priority_encoder IS
    PORT (x: IN BIT_VECTOR (7 DOWNTO 1);
          y: OUT BIT_VECTOR (2 DOWNTO 0));
END priority_encoder;

ARCHITECTURE encoder OF priority_encoder IS
BEGIN
    y(2) <= x(7) OR x(6) OR x(5) OR x(4);
    y(1) <= x(7) OR x(6) OR ((NOT x(5) AND NOT x(4)) AND (x(3) OR x(2)));
    y(0) <= x(7) OR (NOT x(6) AND (x(5) OR (NOT x(4) AND (x(3) OR (NOT x(2) AND x(1))))));
END encoder;

```