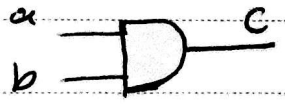
Subject
Date

تعریف سیگنال برای ورودی مدار:



برای ورودی مدار می‌توانیم بصورت زیر سیگنال تعریف کنیم.

a <= '0', '1' after 5ns, '0' after 7ns

* تعریف سیگنال متناوب

~~a <= not b after 5ns~~

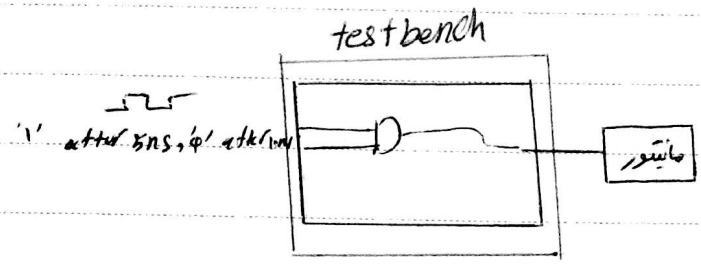
clk <= not clk after half-period;

نکته: اگر بخواهیم چندین بیت را انتقال دهیم.

a <= "0010", "1101" after 5ns, "1110" after 10ns;

تست کردن درست بودن کد نوشته شده

1- نمونه برداری 2- دادن ورودی سیگنال



testbench : محیطی است که درون آن سیگنال تحریک را می‌دهیم و عملکرد مدار را مشاهده می‌کنیم.

نکته: testbench یک مدار منطقی واقعی نیست. یک بستر آزمایش است

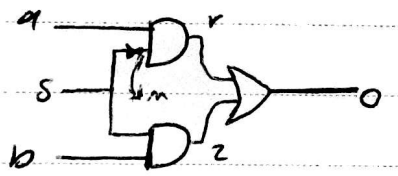
نکته: برای اینکه ارتباطی بین مدارها و دنیای بیرون برقرار کنیم، پورت‌ها را تعریف کردیم، اما درون testbench نیازی به تعریف پورت نیست چون کلیه مدارهایی که طراحی می‌کنیم درون testbench قرار دارند.

```
Entity test bench is
End
Architecture TB of test bench is
    signal a1, b1, c1 : bit
begin
DUT : entity mydesign port map(a => a1, b => b1, c => c1)
    a1 <= '0', '1' after 10ns, '0' after 5ns;
    b1 <= '1', '0' after 13ns, '1' after 10ns;
end TB
```

مراحل یا نمونه برداری

مثال: مدار سالتی پلکسر 2x1 را به همراه testbench آن به زبان VHDL توصیف کنید.

$O = aS' + bS$




```
Entity MUX2 is
port ( a, b, S: in bit ; O : out bit);
end
-----------------------------------
Architecture AR_MUX of MUX2 is
   Signal m, r, z : bit;
begin
      m <= Not S
      r <= m And a
      z <= S And b
      O <= z OR r
end AR_MUX
```

نکته: بدون سیگنال میانی مثال بالا را می‌توان بصورت زیر نوشت.

```
O <= (a And (Not S)) or (b And S);
```

```
Entity testbench is
end

Architecture TB of testbench is
    signal a1, b1, s1, o1 : bit;
    begin

DUT : Entity MUX2 port map (a => a1, b => b1, c => c1);

        a1 <= '0', '1' after 8 ns, '0' after 10 ns;
        b1 <= '1', '0' after 10 ns, '1' after 20 ns;
        s1 <= '0', '1' after 10 ns, '0' after 14 ns;
    end TB
```

مثال ۱: یک مدار جمع کننده یک بیتی به همراه testbench آن طراحی کنید.

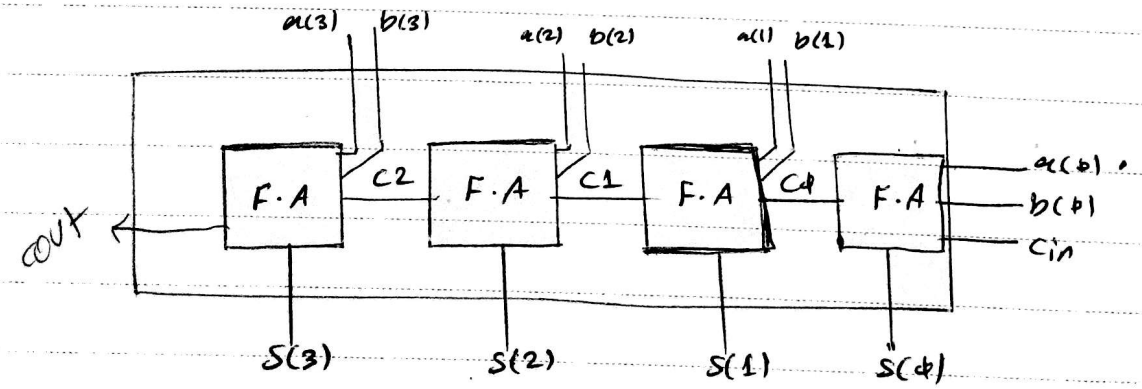
$S = a \oplus b \oplus c$

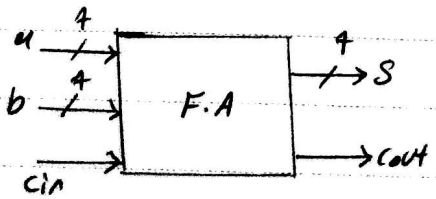
$C = ab + ac + bc$

```
Entity FullAdder is
 port(a, b, c : in bit, s, co : out bit)
 end Entity;

Architecture AR_F of FullAdder is
    begin
      s <= (a xor b) xor c
      c <= ((a And b) or (a And c)) or (b And c)
    end AR_F
```

```
Entity testbench
end

Architecture TB of testbench is
  Signal a1, b1, c1, s1, c2 : bit ;
begin

Full Adder 1 : Entity Full Adder port Map ( a => a1, b => b1 c => c1, Co => c2)

    a1 <= '0', '1' after 10ns , '1' after 12ns ;
    b1 <= '1', '0' after 12ns , '0' after 12ns ;
    c1 <= '0', '1' after 14ns , '0' after 8ns ;
end TB
```

مثال) یک جمع کننده 4 بیتی به همراه testbench بنویسید.





```
Entity Adder_4bit is
  port ( a, b: in bit_vector(3 downto 0); S: out bit_vector(3 downto 0);
         cin: in bit ; cout: out bit )

End Adder_4bit

Architecture Ar_FA of Adder_4bit is
  signal c: bit_vector (2 downto 0);
begin
Ad0 : Entity Adder_1bit port map ( a => a(0), b => b(0), cin => cin
                                   S => S(0), cout => c(0) );

Ad1 : Entity Adder_1bit port map ( a => a(1), b => b(1), cin => c(0)
                                   S => S(1), cout => c(1) );

Ad2 : Entity Adder_1bit port map ( a => a(2), b => b(2), cin => c(1)
                                   S => S(2), cout => c(2) );

Ad3 : Entity Adder_1bit port map ( a => a(3), b => b(3), cin => c(2)
                                   S => S(3), cout => c(3) );

End Ar_FA
```

```
Entity testbench
End

Architecture TB of testbench is
    Signal a1,b1,S1 : bit_vector (3 downto 0);
    Signal Cin,Cout1 : bit;
begin
    DUT : Entity work.Adder_4bit Port Map (a => a1, b => b1, Cin => Cin, S => S1,
                                           Cout => Cout1);

    Cin1 <= '0', '1' after 5ns, '0' after 10ns;
    a1 <= "1001", "0000" after 6ns, "0001" after 10ns;
    b1 <= "1010", "1101" after 10ns, "1110" after 5ns;

End TB
```