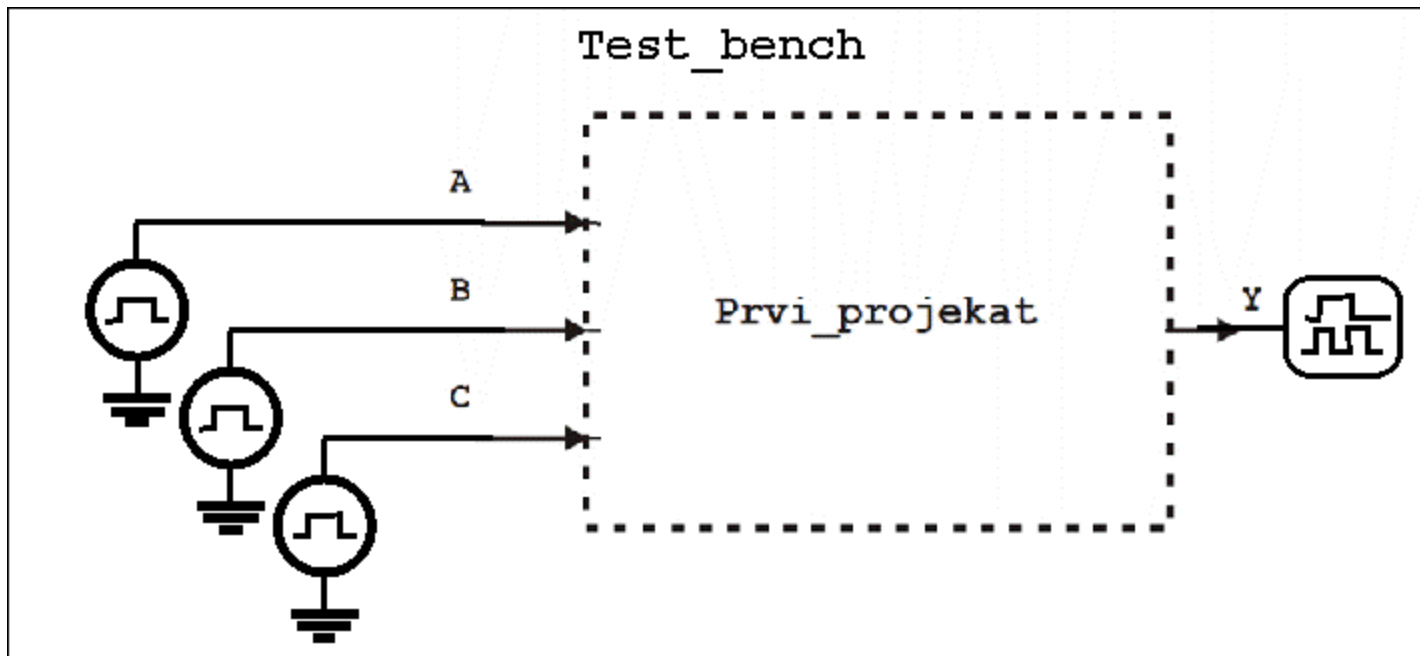


9. Verifikacija projekta (Test Bench)

- a. *Vremenska funkcija* (**after**)
- b. *Konstanta* (**constant**)
- c. *Čekanje* (**wait**)
- d. *Upozorenje* (**assert**)
- e. *Zapis* (**record**);
- f. *Polje* (**array**);
- g. *Petlja* (**loop**);
- h. *Promenljive* (**variable**);
- i. *Zapisivanje u fajl* (**write** i **writeline**);
- j. *Vremenska funkcija* (**now**)
- k. *Čitanje iz fajla* (**read** i **readline**);

9. Verifikacija projekta – (*Test bench*)



9. Verifikacija projekta – (*Test bench*)

Test bench (TB) mora da sadrzi tri celine:

- ◆ - komponentu koja se ispituje (UUT, Unit Under Test)
- ◆ - pobudu (generatori test sekvence)
- ◆ - monitor („instrument“ na kome se prati odziv).

9. Verifikacija projekta – (*Test bench*)

Generatori pobude i monitor satsavni su deo TB, tako da se radi o jedinstvenom entitetu koji ne komunicira sa okolnim svetom preko portova.

Saglasno definiciji entiteta, opis TB entiteta **ne zahteva definisanje ni portova ni opštih konstanti - generika.**

UUT je posebna celina – komponenta opisana entitetom i arhitekturom.

Sa pobudnim signalima i monitorom UUT je vezana preko sopstvenih portova koji **u odnosu na TB predstavljaju interne signale.**

Arhitektura TB opisuje se na strukturnom nivou.



9. Verifikacija projekta – (*Test bench*)

```
entity TestBench is  
end entity TestBench;
```

} "prazan" entitet

```
architecture TB_Arhitektura of TestBench is
```

```
component UUT (Arhitektura_UUT)  
  port(  
end component UUT;
```

} Deklaracija komponente za UUT

```
-- deklarisanje lokalnih signala i konstanti
```

```
begin ← Početak opisa arhitekture
```

```
u1: UUT port map (  
);
```

} Instanciranje komponente
koja se testira

```
konkurentno:
```

```
  sekvencionalno:  
  seq: process
```

```
end process seq;
```

} Testiranje

```
end architecture TB_Arhitektura; ← Kraj opisa arhitekture
```

9. Verifikacija projekta – (*Test bench*)

Definisanje pobude i željenog odziva

Definisanje tabele pobude u okviru jednog procesa pogodno je kod manjih projekata, kada je dužina testnog vektora mala.

Testni vektor predstavlja sve kombinacije promene ulaznih signala koje su neophodne da bi se kolo testiralo.

Redosled ulaznih signala može da se zadaje konkurentno, direktnim dodeljivanjem vrednosti pojedinim signalima ili sekvencijalno, u okviru procesa.

Za veća kola, međutim, mnogo je zgodnije zapisati povorku testnih impulsa u fajl, pa je odatle iščitati.



9. Verifikacija projekta – (*Test bench*)

Definisanje pobude i željenog odziva

Verifikacija => modeli sa vremenskim parametrima.

Podrazumevaju definisanje talasnih oblika ulaznih signala, a takođe, mogu da obuhvataju i sva kašnjenja signala na putu od ulaza do izlaza.

O definisanju kašnjenja kroz pojedine komponente biće reči nešto kasnije; sada ćemo se posvetiti samo definisanju talasnih oblika ulaznog vektora.

9. Verifikacija projekta – (*Test bench*)

Definisanje pobude i željenog odziva

Što se obrade informacija o odzivu tiče, ukoliko povorka izlaznih signala nije velika, najbrži način provere jeste posmatranje talasnih oblika signala.

Međutim, u slučajevima testiranja složenijih kola, kada su povorke izlaznih signala duge, mnogo je povoljnije da se informacija o odzivu upisuje u izlazni fajl.

9. Verifikacija projekta – (*Test bench*)

Definisanje pobude i željenog odziva

Poređenjem željenog i dobijenog talasnog oblika utvrđuje se ispravnost kola, pri čemu je pogodno generisati i odgovarajuće poruke koje projektantu mogu pomoći prilikom otkrivanja i otklanjanja eventualnih nepravilnosti.

Kod složenih kola jako je korisno da se takve poruke upisuju u fajl, i da njihov sastavni deo bude i tačan trenutak na vremenskoj osi u kome je zapaženo odstupanje.

9. Verifikacija projekta – (*Test bench*)

Definisanje pobude i željenog odziva

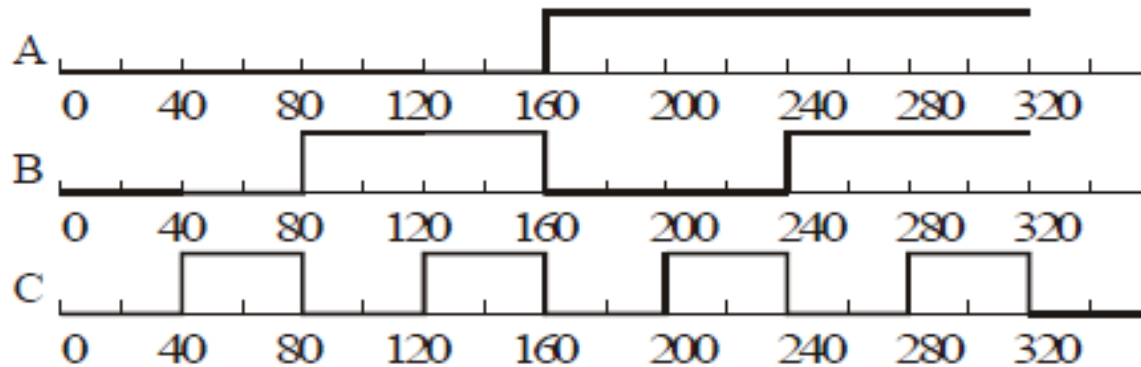
Da bi se ispitao odziv kola **Prvi_projekat** za sve moguće kombinacije ulaznih signala potrebno je da se kolo pobudi sa tačno osam kombinacija:

A	B	C	Y
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	1

9. Verifikacija projekta – (*Test bench*)

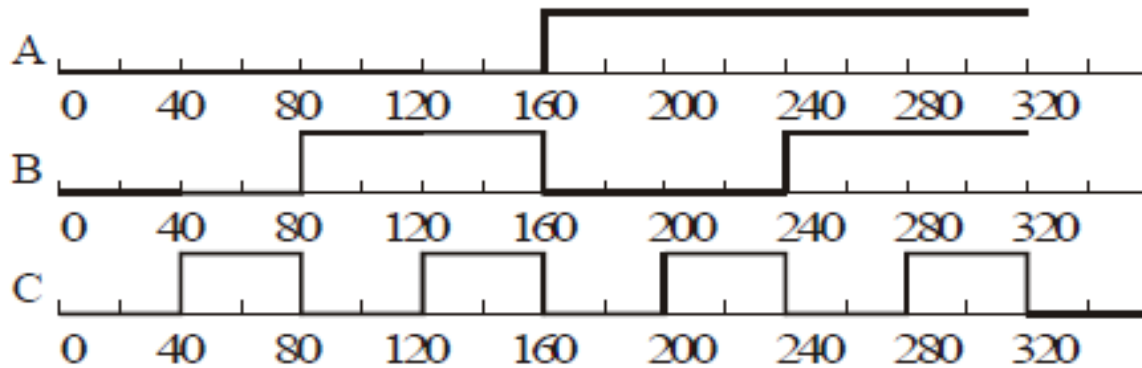
Definisanje pobude i željenog odziva

Da bi se ispitao odziv kola **Prvi_projekat** za sve moguće kombinacije ulaznih signala potrebno je da se kolo pobudi sa tačno osam kombinacija:

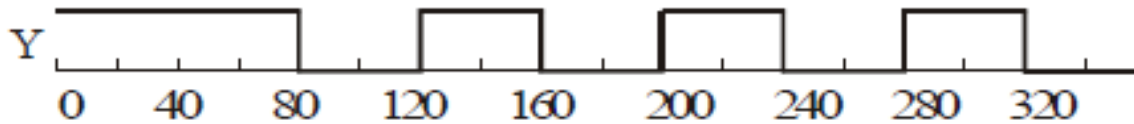


9. Verifikacija projekta – (*Test bench*)

Definisanje pobude



i željenog odziva



9. Verifikacija projekta – (*Test bench*)

Definisanje pobude i željenog odziva (1)

```
1  library IEEE;
2  use IEEE.std_logic_1164.all;
3
4  entity TestBench is
5  end entity TestBench;
6
7  architecture Prvi_projekat_TB of TestBench is
8  -- deklarisanje komponente
9  component Prvi_projekat
10     port (A, B, C: in std logic;
11           Y: out std logic);
12 end component;
13
14 signal A, B, C, Y: std logic; -- deklarisanje signala
15 begin
```

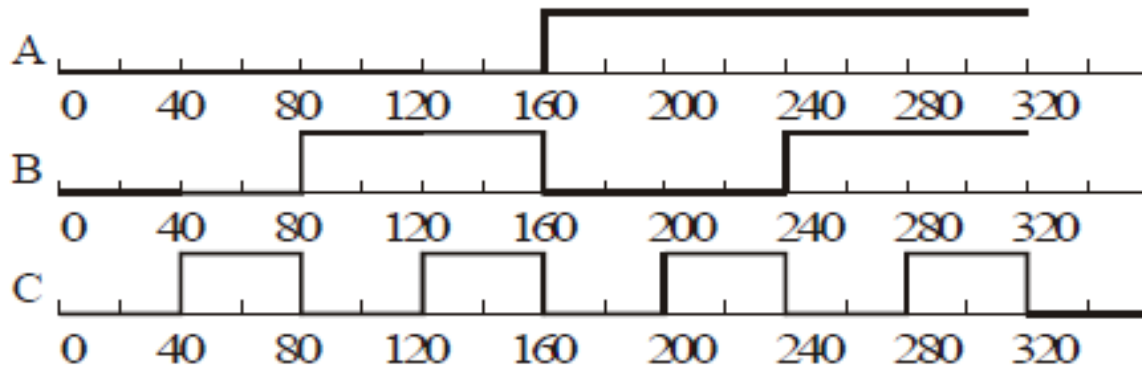
9. Verifikacija projekta – (*Test bench*)

Definisanje pobude i željenog odziva

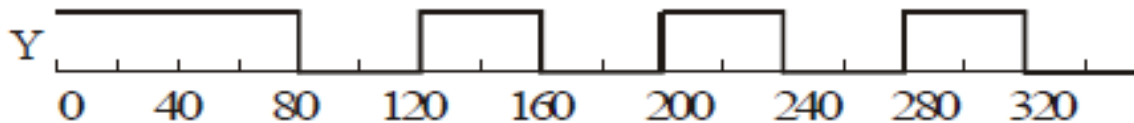
```
15 begin
16   -- instanciranje komponente
17   UUT: Prvi_projekat
18       port map ( A => A, B => B, C => C, Y => Y);
19   -- Zadavanje pobude
20       C <= '0',      '1' after 40ns,
21           '0' after 80ns,
22           '1' after 120ns,
23           '0' after 160ns,
24           '1' after 200ns,
25           '0' after 240ns,
26           '1' after 280ns,
27           '0' after 320ns;
28       B <= '0',      '1' after 80ns,
29           '0' after 160ns,
30           '1' after 240ns;
31       A <= '0',      '1' after 160ns;
32 end architecture Prvi_projekat_TB;
```

9. Verifikacija projekta – (*Test bench*)

Definisanje pobude (drugi način)



i željenog odziva



9. Verifikacija projekta – (*Test bench*)

Definisanje pobude i željenog odziva (2)

Na ovom primeru naučili smo kako se jednostavno uz korišćenje ključne reči **after** specificira talasni oblik pobude. Naredbe su zadate konkurentno – u telu arhitekture

Pored toga, pobuda može da se zadaje i u vidu sekvence signala. U tu svrhu definišaćemo proces **pobuda** i razmotriti drugačiji način zadavanja pobude.



9 Verifikacija projekta – (Test bench)

```
1  -----
2  --
3  -- Title       : Test Bench for prvi projekat
4  -- Design      : Strukturni opis
5  -- Author      : Misa
6  -- Company     : Nis
7  --
8  -----
9  --
10 -- File        : $DSN\src\TestBench\prvi projekat TB.vhd
11 -- Generated   : 3/13/2010, 12:43 PM
12 -- From       : $DSN\src\prvi primer.vhd
13 -- By        : Active-HDL Built-in Test Bench Generator ver. 1.2s
14 --
15 -----
16 --
17 -- Description : Automatically generated Test Bench for prvi projekat tb
18 --
19 -----
20 --
21 library ieee;
22 use ieee.std_logic_1164.all;
23
24     -- Add your library and packages declaration here ...
25
26 entity prvi_projekat_tb is
27 end prvi_projekat_tb;
```

9. Verifikacija projekta – (Test bench)

Definisanje pobude i željenog odziva (2)

```
31 architecture TB_ARCHITECTURE of prvi_projekat_tb is
32     -- Component declaration of the tested unit
33     component prvi_projekat
34     port(
35         a : in std logic;
36         b : in std logic;
37         c : in std logic;
38         y : out std logic );
39     end component;
40
41     -- Stimulus signals - signals mapped to the input and inout ports of
42     tested entity
43     signal a : std logic;
44     signal b : std logic;
45     signal c : std logic;
46     -- Observed signals - signals mapped to the output ports of tested
47     entity
48     signal y : std logic;
49     -- Add your code here ...
50     signal ulaz: std_logic_vector (2 downto 0); -- deklarisanje novog
51     signala
52     constant PropDelay: time := 40 ns; -- deklarisanje konstante
53 begin
```

9. Verifikacija projekta – (*Test bench*)

Definisanje pobude i željenog odziva (2)

```
50 begin
51
52     -- Unit Under Test port map
53     UUT : prvi_projekat
54         port map (
55             a => ulaz(2), -- a=MSB od ulaz
56             b => ulaz(1),
57             c => ulaz(0), -- c=LSB od ulaz
58             y => y
59         );
60
61     -- Add your stimulus here ...
62     pobuda: process
63     begin
64         ulaz <= "000"; wait for PropDelay;
65         ulaz <= "001"; wait for PropDelay;
66         ulaz <= "010"; wait for PropDelay;
67         ulaz <= "011"; wait for PropDelay;
68         ulaz <= "100"; wait for PropDelay;
69         ulaz <= "101"; wait for PropDelay;
70         ulaz <= "110"; wait for PropDelay;
71         ulaz <= "111"; wait for PropDelay;
72     end process pobuda;
73 end TB_ARCHITECTURE;
```

9. Verifikacija projekta – (*Test bench*)

Definisanje pobude i željenog odziva (2)

```
73 end TB_ARCHITECTURE;  
74  
75 configuration TESTBENCH_FOR_prvi_projekat of prvi_projekat_tb is  
76     for TB_ARCHITECTURE  
77         for UUT : prvi_projekat  
78             use entity work.prvi_projekat (strukturni_opis);  
79         end for;  
80     end for;  
81 end TESTBENCH_FOR_prvi_projekat;
```

```
73 end TB_ARCHITECTURE;  
74  
75 configuration TESTBENCH_FOR_prvi_projekat of prvi_projekat_tb is  
76     for TB_ARCHITECTURE  
77         for UUT : prvi_projekat  
78             use entity work.prvi_projekat (protok_podataka);  
79         end for;  
80     end for;  
81 end TESTBENCH_FOR_prvi_projekat;
```