جامعة نينوى
كلية هندسة الإلكترونيات

Circuit Design with VHDL 10

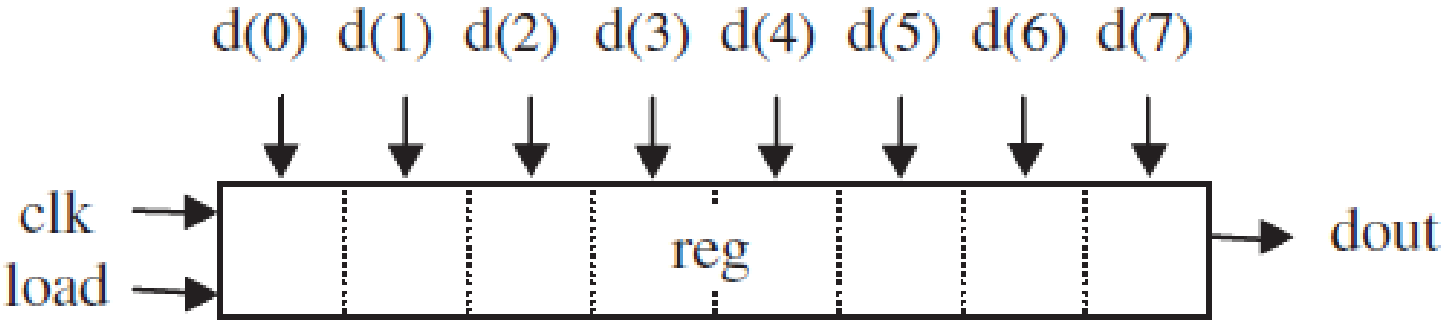
Textbook: Volnei A. Pedroni

Submitted By: Hussein Aideen

- Signed and Unsigned Comparators

```
1  ---- Signed Comparator: ---------------
2  LIBRARY ieee;
3  USE ieee.std_logic_1164.all;
4  USE ieee.std_logic_arith.all;  -- necessary!
5  ----------------------------------------
6  ENTITY comparator IS
7     GENERIC (n: INTEGER := 7);
8     PORT (a, b: IN SIGNED (n DOWNTO 0);
9           x1, x2, x3: OUT STD_LOGIC);
10 END comparator;
11 ----------------------------------------
12 ARCHITECTURE signed OF comparator IS
13 BEGIN
14    x1 <= '1' WHEN a > b ELSE '0';
15    x2 <= '1' WHEN a = b ELSE '0';
16    x3 <= '1' WHEN a < b ELSE '0';
17 END signed;
18 ----------------------------------------
```

- Parallel-to-Serial Converter
- circuit operation:
- when load = 1 the data must stored in the register.
- when load= 0 the registerd data shifted out to dout each time clk goes from 0 to 1.



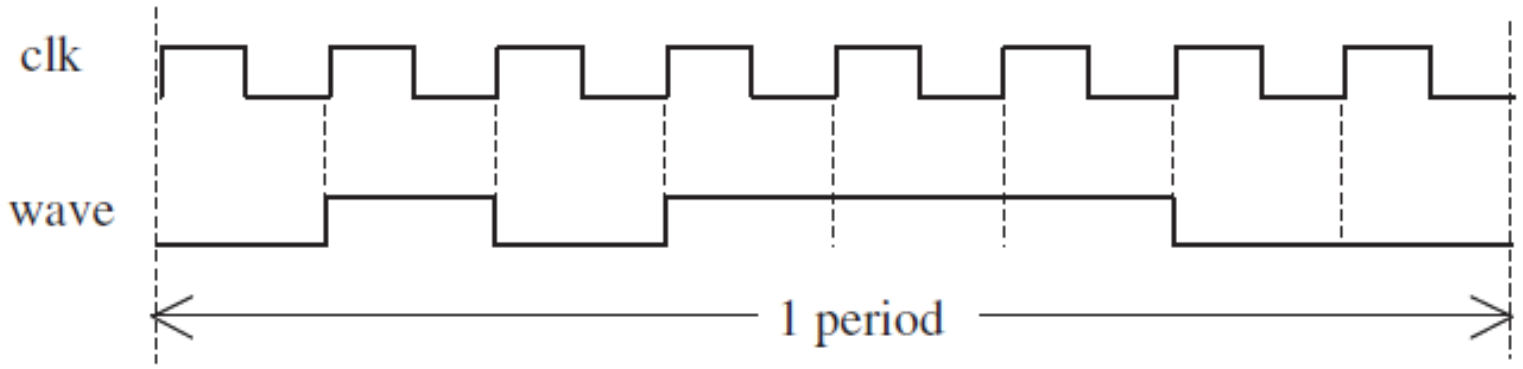
- Parallel-to-Serial Converter

```
1     ----------------------------------------
2     LIBRARY ieee;
3     USE ieee.std_logic_1164.all;
4     ----------------------------------------
5     ENTITY serial_converter IS
6        PORT ( d: IN STD_LOGIC_VECTOR (7 DOWNTO 0);
7               clk, load: IN STD_LOGIC;
8               dout: OUT STD_LOGIC);
9     END serial_converter;
10 ----------------------------------------
```

- Parallel-to-Serial Converter

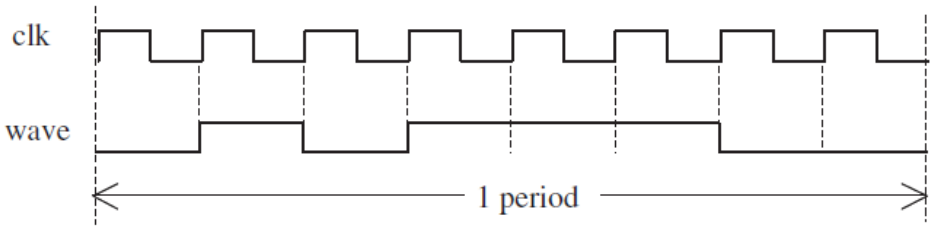
```
11 ARCHITECTURE serial_converter OF serial_converter IS
12     SIGNAL reg: STD_LOGIC_VECTOR (7 DOWNTO 0);
13 BEGIN
14     PROCESS (clk)
15     BEGIN
16         IF (clk'EVENT AND clk='1') THEN
17             IF (load='1') THEN  reg <= d;
18             ELSE reg <= reg(6 DOWNTO 0) & '0';
19             END IF;
20         END IF;
21     END PROCESS;
22     dout <= reg(7);
23 END serial_converter;
24 ---------------------------------------------
```

- Signal Generators



VHDL> Examples

- Signal Generators




```
1  ---------------------------------------
2  LIBRARY ieee;
3  USE ieee.std_logic_1164.all;
4  ---------------------------------------
5  ENTITY signal_gen1 IS
6     PORT (clk: IN BIT;
7           wave: OUT BIT);
8  END signal_gen1;
9  ---------------------------------------
```

VHDL> Examples

- Signal Generators

```
10 ARCHITECTURE arch1 OF signal_gen1 IS
11 BEGIN
12    PROCESS
13       VARIABLE count: INTEGER RANGE 0 TO 7;
14    BEGIN
15       WAIT UNTIL (clk'EVENT AND clk='1');
16       CASE count IS
17          WHEN 0 => wave <= '0';
18          WHEN 1 => wave <= '1';
19          WHEN 2 => wave <= '0';
20          WHEN 3 => wave <= '1';
21          WHEN 4 => wave <= '1';
22          WHEN 5 => wave <= '1';
23          WHEN 6 => wave <= '0':
24           WHEN 7 => wave <= '0';
25       END CASE;
26       count := count + 1;
27    END PROCESS;
28 END arch1;
29 ------------------------------------
```