المحور الثاني : lect.3 & lect 4

1. The core building block in a modern FPGA from Xilinx is called a ....
   a) Logic cell (LC) b) Logic element (LE)
   c) Logic block (LB) d) Logic gate

2. In Xilinx, a logic cell (LC) comprises a ....
   a) 2-input LUT and multiplexer b) 4-input LUT, a multiplexer and a register
   c) Gates only d) Registers only

3. The equivalent core building block in an FPGA from Altera is called a.........
   a) Logic cell (LC) b) Logic element (LE)
   c) Logic block (LB) d) Logic gate

4. A Xilinx slice contains.........
   a) Two logic cells b) Six logic cells
   c) Four logic cells d) Three logic cells

5. A Configurable logic block (CLB) in Xilinx contains............
   a) Four slices b) Six slices
   c) Ten slices d) Eight slices

6. The LUTs within a single Configurable logic block (CLB) to be configured together to implement a shift register containing up to ....
   a) 64 bits b) 256 bits
   c) 128 bits d) None of the above

7. FPGAs include relatively large chunks of embedded RAM called ………….
   a) e-RAM b) block RAM
   c) A and B d) None of the above

8. Embedded RAM blocks can be used for a variety of purposes such as implementing …………
   a) Sigle- or dual- port RAM b) First-in first-out (FIFO) functions
   c) State machines d) All of the above

9. The main clock signal branches again and again in ……….
   a) Clock tree b) Clock manager
   c) Clock pin d) Clock signal

10. ....generates a number of daughter clocks.
    a) Clock tree b) Clock manager
    c) Clock pin d) Clock signal

11. Clock managers may also support ………………
    a) Phase shift b) Jitter removal
    c) Frequency synthesis d) All of the above

12. The operation when the clock manager is used to generate daughter clocks with frequencies that are derived by multiplying or dividing the original signal is called a...............
    a) Phase shifting b) Jitter removal
    c) Frequency synthesis d) None of the above

13. The operation when the clock manager is used to generate daughter clocks that are phase shifted with respect to each other is called a..............
    a) Phase shifting b) Jitter removal
    c) Frequency synthesis d) None of the above

14. ………….. contains the information that will be uploaded into the FPGA in order to program it to perform a specific function
    a) Block file b) Erasing file
    c) Injection file d) Configuration file

15. Bits that are used to define the state of programmable logic elements directly are called …………
    a) Block data b) Configuration data
    c) Injection data d) Erasing data

16. Instructions that tell the device what to do with the configuration data are called ……………
    a) Block commands b) Injection commands
    c) Configuration commands d) Erasing commands

17. ....declaration contains a list of all libraries to be used in the design. For example: ieee, std, work, etc.
    a) library b) architecture
    c) entity d) package

18. Entity section in VHDL program, specifies ....
    a) How circuit behaves b) I/O pins of the circuits

c) List of libraries d) All of the above

19. ....section in VHDL program, contains the VHDL code proper, which describes how the circuit should behave.
   a) library b) architecture
   c) entity d) package

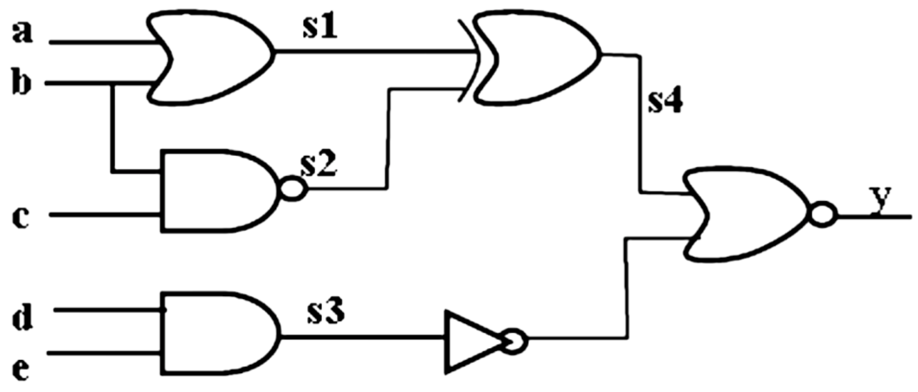
20. the ieee library contains several packages, including …………….
   a) Std_logic_1164 b) Std_logic_signed
   c) Std_logic_arith d) All of the above

Q2: draw the equivalent logic circuit for the VHDL code shown below

1).

```
entity log_cct is
port ( a,b,c,d,e,:in bit;_y : out bit);
end log_cct;
architecture fin_logic of log_cct is
signal s1,s2,s3,s4: bit;
begin
s1 <= a or b;
s2 <= b nand c;
s3 <= d and e;
s4 <= s1 xor s2;
y <= s4 nor (not s3);
end fin_logic;
```

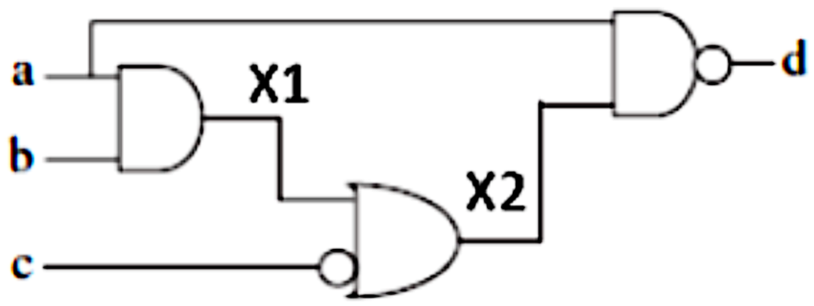
Sol:



2).

```
entity exam is
port ( a,b,c: IN bit; d: OUT bit);
end exam;
ARCHITECTURE behavior OF exam IS
signal x1,x2:bit;
begin
x1<= a and b;
x2<= x1 or not c;
d<= a nand x2;
end behavior;
```

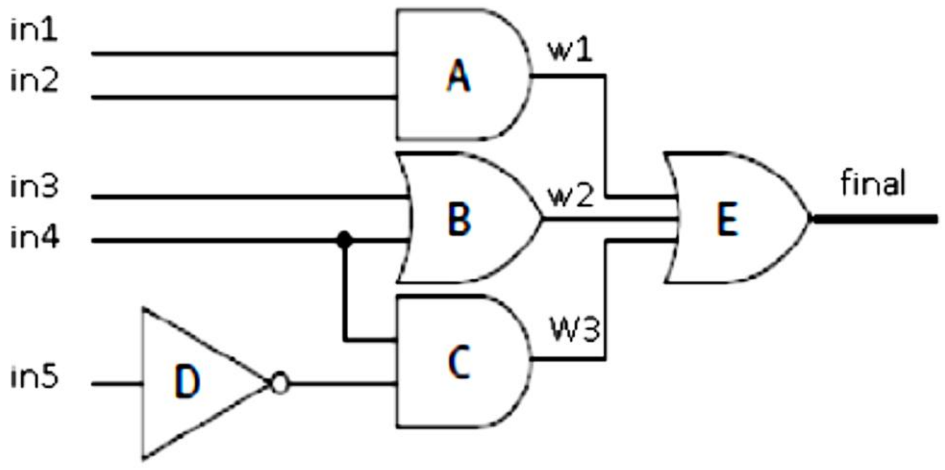
Sol:



3).

```
entity mix_logic is
port (in1,in2,in3,in4,in5 : in bit;
                 final : out bit);
end mix_logic;
architecture sec_logic of mix_logic is
signal w1,w2,w3 : bit;
begin
w1 <= in1 and in2;
w2 <= in3 or in4;
w3 <= in4 and (not in5);
final <= w1 or w2 or w3;
end  sec_logic;
```

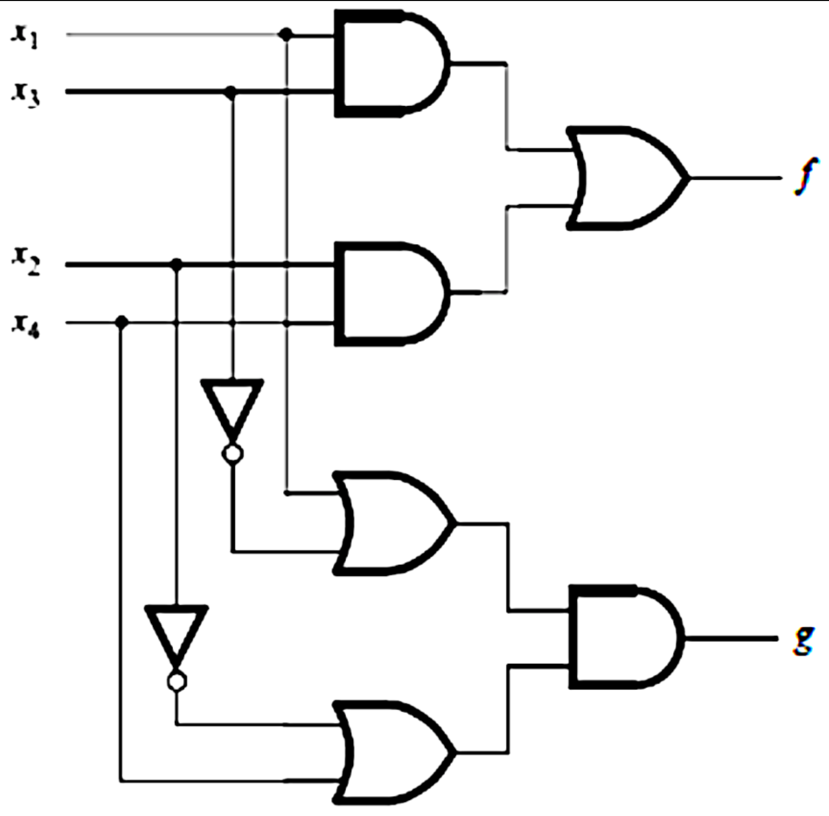
Sol:



4).

```
entity cct1 is
port (x1,x2,x3,x4 : in bit;
       f,g : out bit);
end cct1;
architecture cct2 of cct1 is
signal sig1,sig2,sig3 , sig4: bit;
begin
sig1 <= x1 and x3;
sig2 <= x2 and x4 ;
sig3 <= x1 or (not x3);
sig4 <= x4 or ( not x2);
f <= sig1 or sig2;
g <= sig3 and sig4;
end cct2;
```

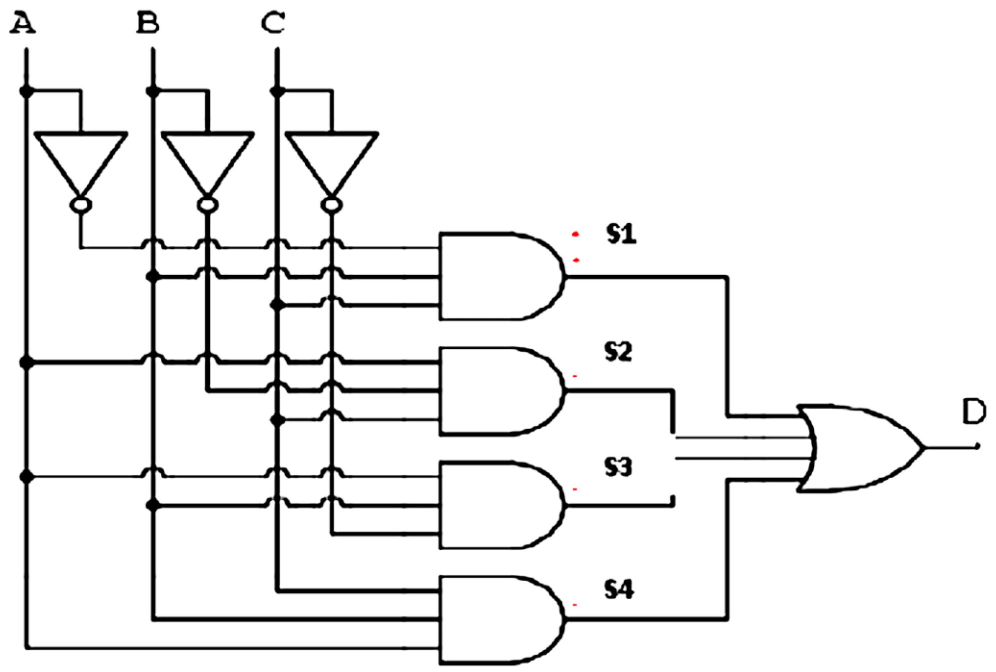
Sol:



5).

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
entity test1_s is
Port ( a,b,c : in STD_LOGIC;
d : out STD_LOGIC);
end test1_s;
architecture Behavioral of test1_s is
signal s1,s2,s3,s4:std_logic;
begin
s1<= (not a and b and c);
s2<= (not b and a and c) ;
s3<= (not c and a and b);
s4<= (a and b and c);
d<=( s1 or s2 or s3 or s4);
end Behavioral;
```

Sol:


A
B
C
S1
S2
S3
S4
D