

Εισαγωγή στη Γλώσσα VHDL

- Παράδειγμα and3
- Entity και Architecture
- Entity
- Entity - Παραδείγματα
- Architecture
- VHDL simulation παραδείγματος and3
- Παράδειγμα NAND
- VHDL simulation παραδείγματος nand
- Boolean Operators
- Παράδειγμα XOR με NAND
- VHDL simulation παραδείγματος XOR με NAND
- Παράδειγμα συνάρτησης Boole
- VHDL simulation παραδείγματος συνάρτησης Boole
- Ασκήσεις - Προβλήματα

Παράδειγμα and3

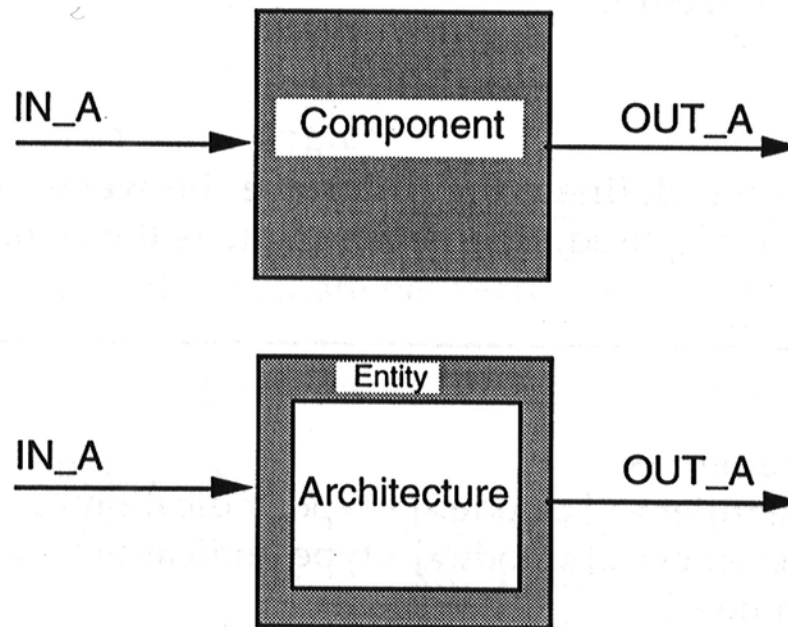
```
-- File: c:\my designs\AND3\SRC\and3.VHD
-- created by Design Wizard: 01/24/01 19:23:03
--
library IEEE;
use IEEE.std_logic_1164.all;

entity and3 is
    port ( a, b, c :    in STD_LOGIC;    --inputs
          d      :    out STD_LOGIC);    --outputs
end and3;

architecture data_flow of and3 is
begin
    d <= a and b and c;
end data_flow;
```

Entity και Architecture

- Ένα component στην VHDL αποτελείται από δύο μέρη
 - Entity : Περιέχει τον ορισμό των εισόδων - εξόδων
 - Architecture : Περιγραφή του component



Entity και Architecture

```
Entity VHDL_COMPONENT is  
  port( A : in std_logic;  
        B : out std_logic);  
end VHDL_COMPONENT;
```

Entity
declaration

```
Architecture VHDL_CODE of VHDL_COMPONENT is  
begin  
  B <= not A;  
  ....  
end VHDL_CODE;
```

Architecture
declaration

Code

Entity

- Ορισμός των σημάτων σε **in**, **out**, **inout**, **buffer**.

Syntax:

```
entity <identifier_name> is  
port ( [signal] <identifier> : [<mode>] <type_indication> ;  
       [signal] <identifier> : [<mode>] <type_indication> ) ;  
end [ <identifier_name> ] ;
```

Syntax (VHDL-93):

```
entity <identifier_name> is  
port ( [signal] <identifier> : [<mode>] <type_indication> ;  
       [signal] <identifier> : [<mode>] <type_indication> ) ;  
end entity [ <identifier_name> ] ;
```

Entity - Παραδείγματα

Entity and3 **is**

port(**signal** a, b, c : **in** std_logic ;
 signal d : **out** std_logic);

end and3 ;

Entity and3 **is**

port(a, b, c : **in** std_logic ;
 d : **out** std_logic);

end and3 ;

Architecture

- Στο architecture ορίζεται το κυρίως σώμα για ένα component entity

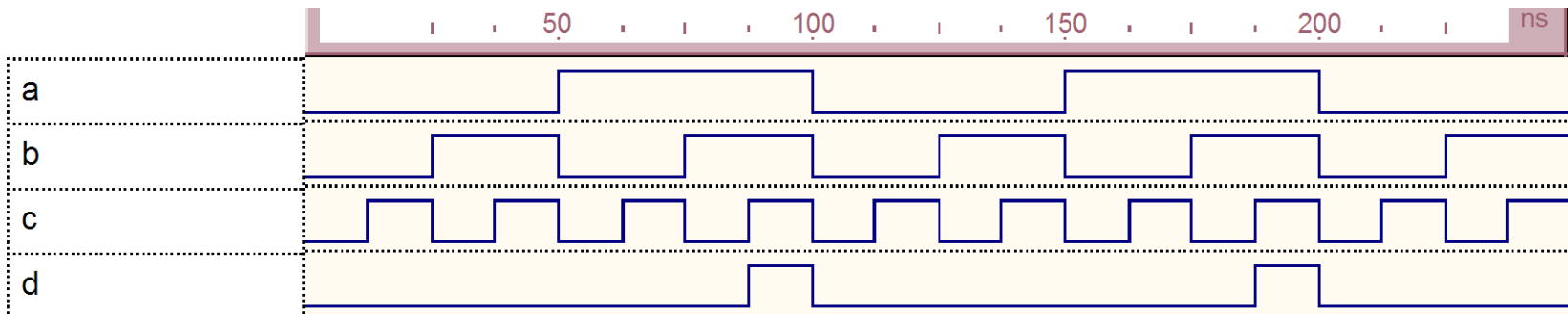
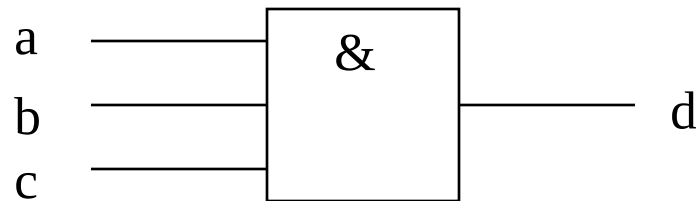
Syntax:

```
architecture <architecture_name> of <entity_identifier> is  
[<architecture_declarative_part>]  
begin  
    <architecture_statement_part>  
end [ <architecture_name> ] ;
```

Syntax (VHDL-93):

```
architecture <architecture_name> of <entity_identifier> is  
[<architecture_declarative_part>]  
begin  
    <architecture_statement_part>  
end [architecture] [ <architecture_name> ] ;
```

VHDL simulation παραδείγματος and3



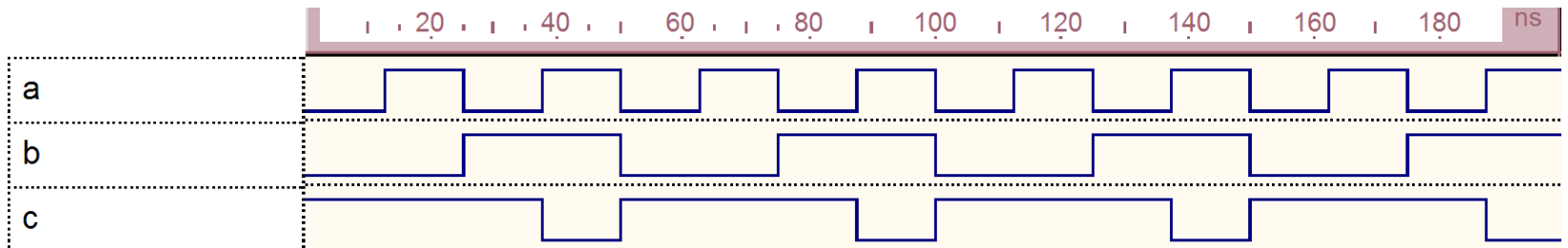
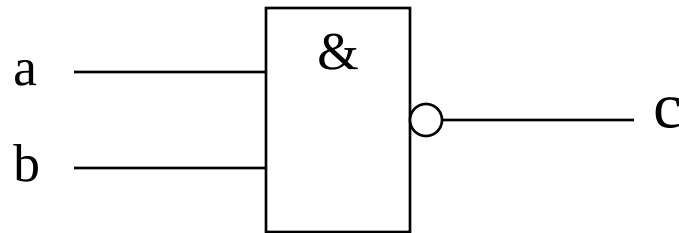
Παράδειγμα NAND

```
-- File: c:\my designs\nand_vhdl\SRC\nand_comp.VHD
-- created by Design Wizard: 01/25/01 12:12:58
--
library IEEE;
use IEEE.std_logic_1164.all;

entity nand_comp is
    port (a, b : in STD_LOGIC;      -- inputs
          c   : out STD_LOGIC);      -- output
end nand_comp ;

architecture nand_behv of nand_comp is
    signal int: STD_LOGIC              -- internal signal declaration
begin
    int<= a and b;
    c <= not int;
end nand_behv ;
```

VHDL simulation παραδείγματος nand



Boolean Operators

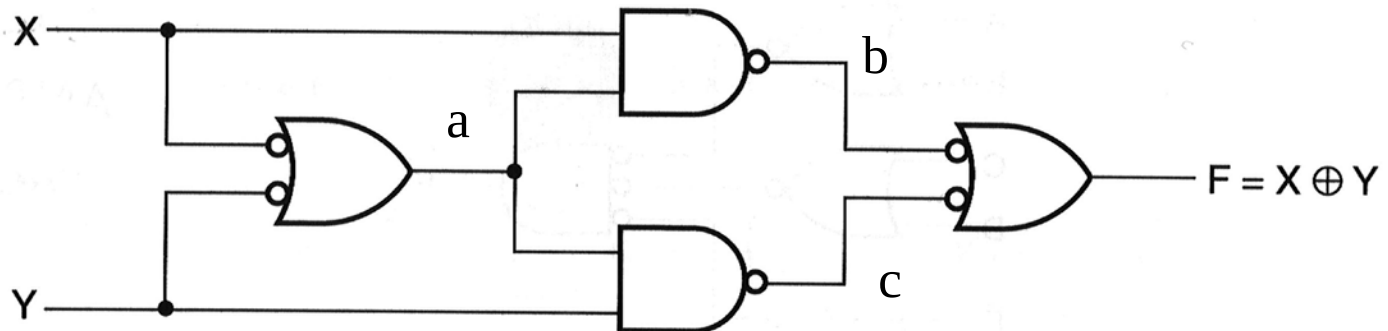
- NOT
- AND
- NAND
- OR
- NOR
- XOR
- XNOR -- VHDL-93

- Απλοποίηση παραδείγματος
nand

```
architecture nand_behv of nand_comp is  
begin  
    c <= a nand b;  
end nand_behv ;
```

Παράδειγμα XOR με NAND

X	Y	F
0	0	0
0	1	1
1	0	1
1	1	0



Παράδειγμα XOR με NAND

-- File: c:\my designs\xor_with_nand\SRC\xor_with_nand.VHD

-- created by Design Wizard: 01/25/01 12:59:08

library IEEE;

use IEEE.std_logic_1164.all;

entity xor_with_nand **is**

port (X, Y : **in** STD_LOGIC;

 F : **out** STD_LOGIC);

end xor_with_nand;

architecture xor_comp **of** xor_with_nand **is**

signal a, b, c: STD_LOGIC;

begin

 a <= (**not** X) or (**not** Y);

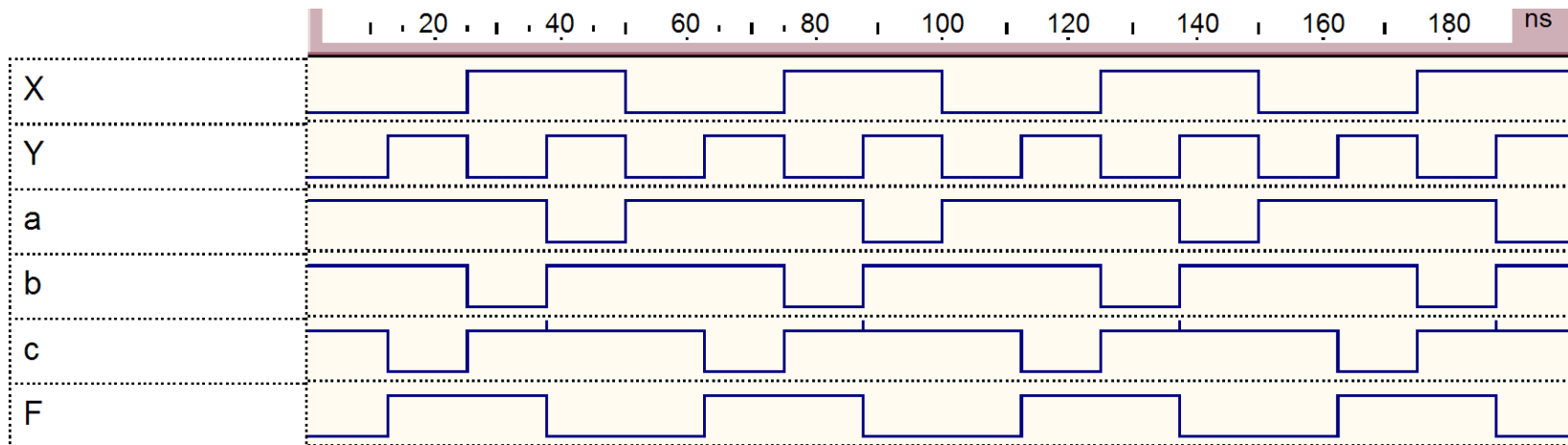
 b <= X **nand** a;

 c <= Y **nand** a;

 F <= (**not** b) or (**not** c);

end xor_comp;

VHDL simulation παραδείγματος XOR με NAND



Παράδειγμα συνάρτησης Boole

$$F(W,X,Y,Z) = \Sigma m(0,1,2,4,5,6,8,9,12,13,14)$$

m_0	m_1	m_3	m_2
m_4	m_5	m_7	m_6
m_{12}	m_{13}	m_{15}	m_{14}
m_8	m_9	m_{11}	m_{10}

WX \ YZ	Y			
	00	01	11	10
00				
01				
11				
10				

Z

X

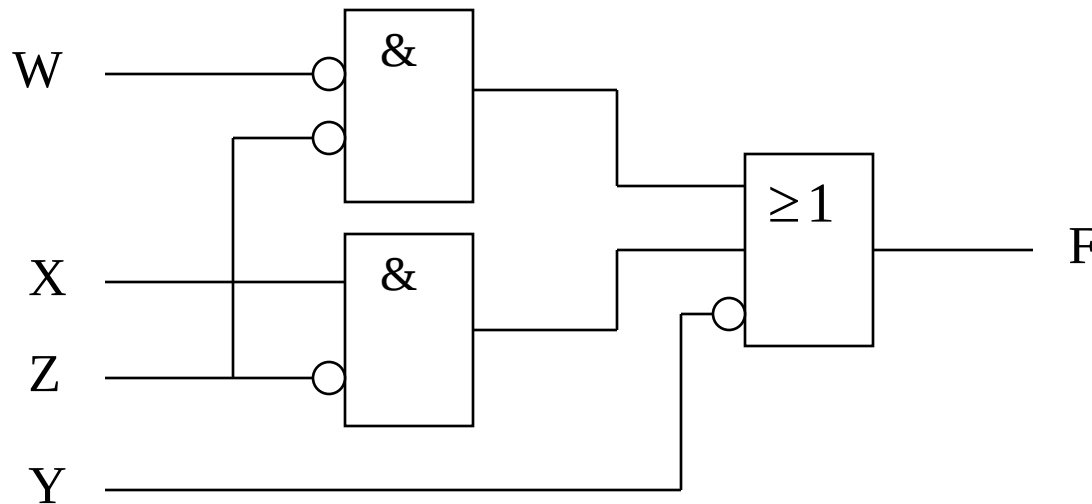
WX \ YZ	Y			
	00	01	11	10
00	1	1		1
01	1	1		1
11	1	1		1
10	1	1		

Z

X

Παράδειγμα συνάρτησης Boole

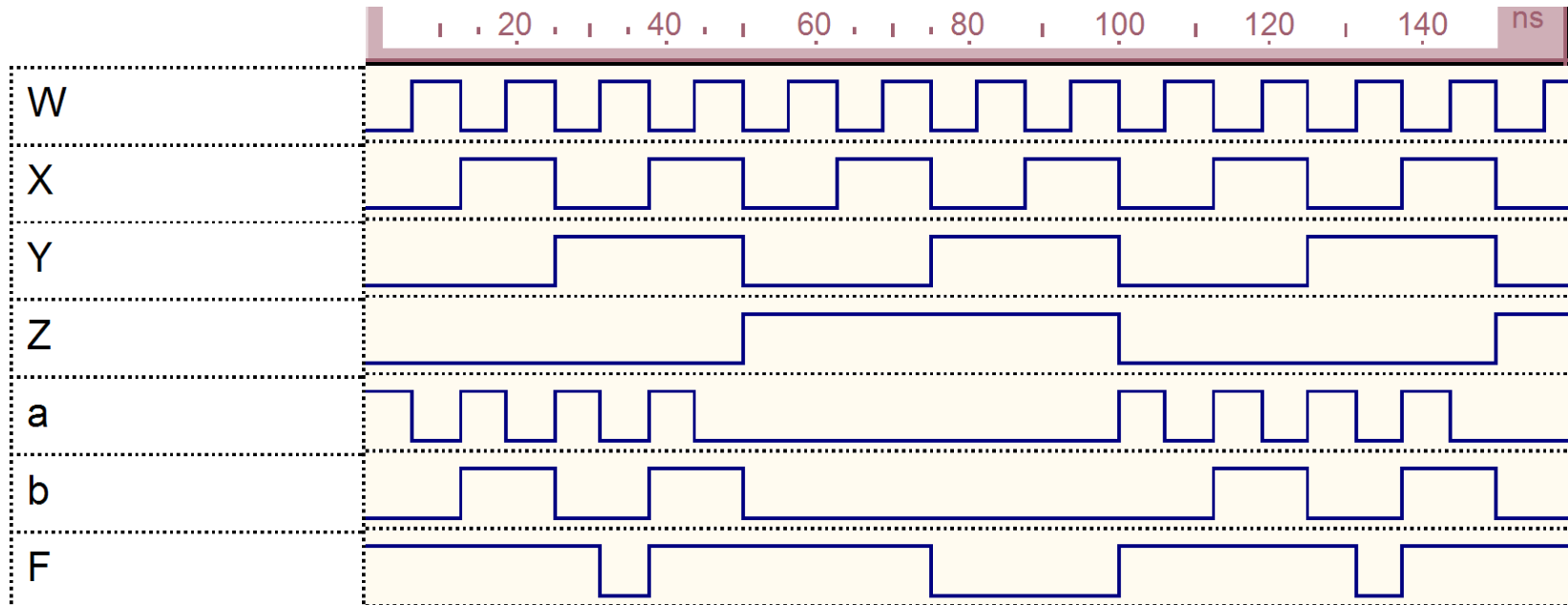
$$F = \overline{Y} + \overline{W}\overline{Z} + X\overline{Z}$$



Παράδειγμα συνάρτησης Boole

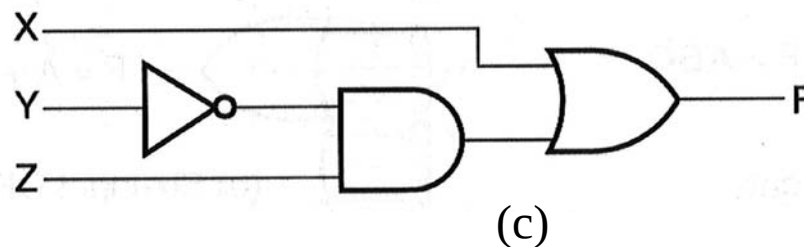
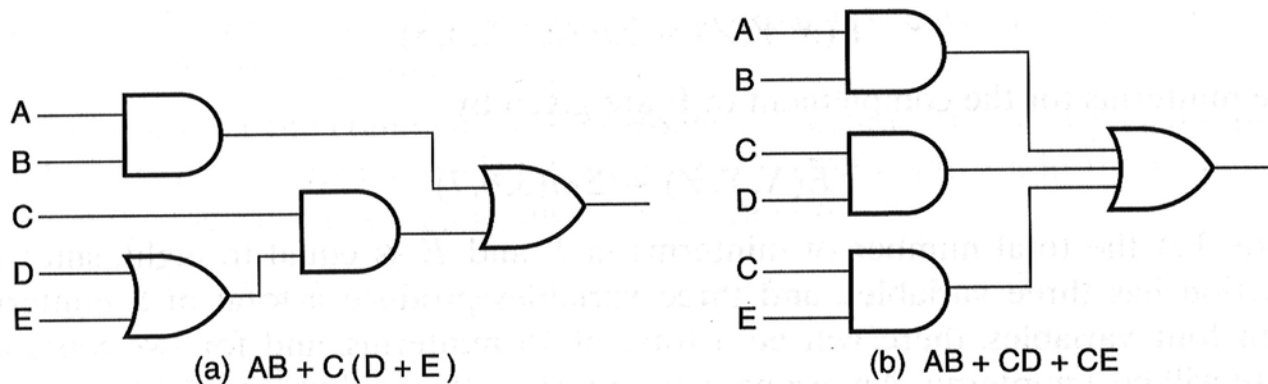
```
--  
-- File: c:\my designs\f_boole\SRC\f_boole.VHD  
-- created by Design Wizard: 01/25/01 18:29:15  
--  
library IEEE;  
use IEEE.std_logic_1164.all;  
  
entity f_boole is  
    port ( W, X, Y, Z: in STD_LOGIC;  
          F: out STD_LOGIC);  
end f_boole;  
  
architecture f_boole_comp of f_boole is  
signal a, b, c: STD_LOGIC;  
begin  
    a <= (not W) and (not Z);  
    b <= X and (not Z);  
    F <= (not Y) or a or b;  
end f_boole_comp;
```

VHDL simulation παραδείγματος συνάρτησης Boole



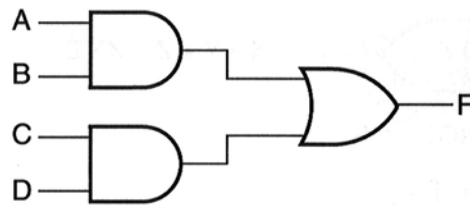
Ασκήσεις - Προβλήματα

1. Για τα παρακάτω κυκλώματα υπολογίστε τον Πίνακα Αληθείας και στη συνέχεια αναπτύξτε τον κώδικα σε VHDL. Συγκρίνετε τα αποτελέσματα του Πίνακα Αληθείας με τα αποτελέσματα της εξομοίωσης (simulation).

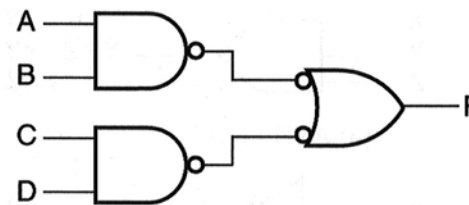


Ασκήσεις - Προβλήματα

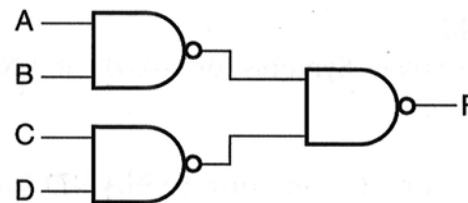
2. Για τα παρακάτω κυκλώματα υπολογίστε τον Πίνακα Αληθείας και στη συνέχεια αναπτύξτε για μόνο ένα από αυτά τον κώδικα σε VHDL. Συγκρίνετε τα αποτελέσματα του Πίνακα Αληθείας με τα αποτελέσματα της εξομοίωσης (simulation).



(a)



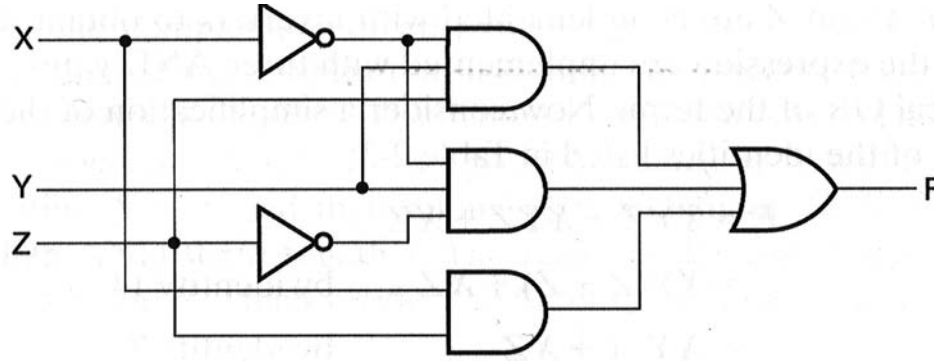
(b)



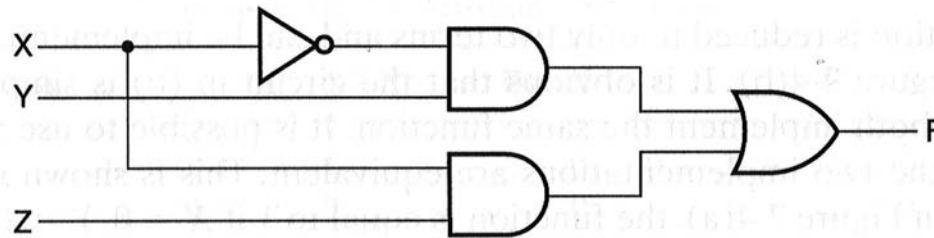
(c)

Ασκήσεις – Προβλήματα

3. Για τα παρακάτω κυκλώματα υπολογίστε τον Πίνακα Αληθείας και στη συνέχεια αναπτύξτε τον κώδικα σε VHDL. Συγκρίνετε τα αποτελέσματα του Πίνακα Αληθείας με τα αποτελέσματα της εξομοίωσης (simulation).



(a) $F = \bar{X}YZ + \bar{X}Y\bar{Z} + XZ$



(b) $F = \bar{X}Y + XZ$

Ασκήσεις – Προβλήματα

4. Να απλουστεύσετε τις ακόλουθες συναρτήσεις Boole χρησιμοποιώντας Πίνακες Karnaugh. Ακολουθώντας να συντάξετε τον Πίνακα Αληθείας, να σχεδιάσετε το απλουστευμένο κύκλωμα με πύλες και να αναπτύξετε τον κώδικα σε VHDL. Συγκρίνετε τα αποτελέσματα του Πίνακα Αληθείας με τα αποτελέσματα της εξομοίωσης (simulation).

$$F = \overline{A}\overline{B}\overline{C} + \overline{B}\overline{C}\overline{D} + \overline{A}\overline{B}\overline{C} + \overline{A}\overline{B}\overline{C}\overline{D}$$

$$F = \overline{A}\overline{C} + \overline{B}D + \overline{A}CD + ABCD$$

$$F(A, B, C) = \Sigma(3,4,6,7)$$

$$F(A, B, C) = \Sigma(0,2,4,5,6)$$

$$F(A, B, C, D) = \Sigma(1,5,9,12,13,15)$$

$$F(A, B, C, D) = \Sigma(1,3,9,11,12,13,14,15)$$

$$F(A, B, C, D) = \Sigma(0,2,4,5,6,7,8,10,13,15)$$