

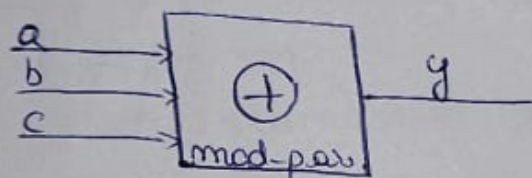
Să se realizeze un circuit digital care efectuează det. parității a unui număr binar pe 3 biți. D.p.d.v. binar, un nr. e impar când nr. de biți de 1 este impar.

Obs) A nu se confunda cu paritatea numerelor zecimale.

001 - impar

011 - par (binar) -  $3_{10}$  - impar d.p.d.v. zecimal

100 - impar



| a | b | c | y |
|---|---|---|---|
| 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 |
| 0 | 1 | 0 | 1 |
| 0 | 1 | 1 | 0 |
| 1 | 0 | 0 | 1 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 0 |
| 1 | 1 | 1 | 1 |

0 - par

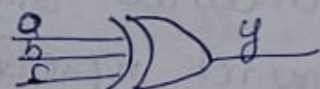
1 - impar

$$y = \bar{a}\bar{b}c + \bar{a}b\bar{c} + a\bar{b}\bar{c} + abc$$

$$\Rightarrow y = \bar{a}(\bar{b}c + b\bar{c}) + a(\bar{b}\bar{c} + bc)$$

$$\Rightarrow y = \bar{a}(b \oplus c) + a(\overline{b \oplus c})$$

$$\Rightarrow y = a \oplus b \oplus c$$



Prin descriere hardware folosim limbajul VHDL  
avem următorul cod:



```

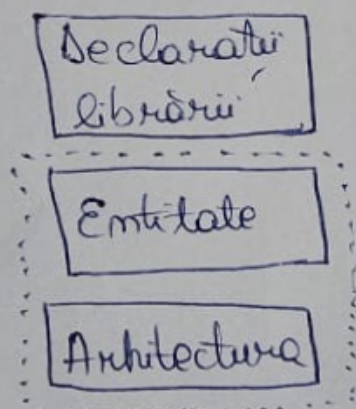
LIBRARY IEEE;
USE IEEE.std_logic_1164.all;
ENTITY mod_par IS
PORT (a, b, c: IN std_logic;
      y: OUT std_logic);
END mod_par;
ARCHITECTURE ARCHITECTURE docu_ex OF mod_par IS
BEGIN
    y <= a XOR b XOR c;
END docu_ex;

```

VHDL

~~EXERCITIU~~  
~~VHDL~~

## UNITĂȚILE FUNDAMENTALE ALE LIMBAJULUI



În limbajul VHDL se vor găsi întotdeauna urm. unități fundamentale:

- declararea librărilor care conține o listă de librări pt a fi utilizate în proiectul curent (IEEE, STD, WORK)
- entitatea în care se specifică circuitul digital sub forma unei black-box și sunt reprezentate toate intrările și ieșirile modului digital.
- arhitectura e zona în care se descrie componența circuitului digital.



## Declanarea bibliotecilor:

LIBRARY nume-librărie;

USE nume-librărie . nume-pachet . parte-dim-pachet;

LIBRARY = această clauză permite introducerea în cadrul proiectului a unei librării

USE = e utilizată pt. specificarea pachetelor și părților acestora din cadrul unei librării

LIBRARY IEEE;

USE IEEE . std-logic-1164.all;

USE IEEE . std-arith.all;

USE IEEE . std-signed.all;

LIBRARY STD;

USE STD . standard.all;

Clauza „all” specifică faptul că este inclus tot pachetul în cadrul proiectului.

Cele mai utilizate pachete din librăria IEEE:

- standard-logic-1164 = este utilizat pt. reprezentarea tuturor nivelelor logice.

Specifică 8 nivele logice și un nivel unlogic. ('0', '1', 'z',

'H', 'W')

high impedance

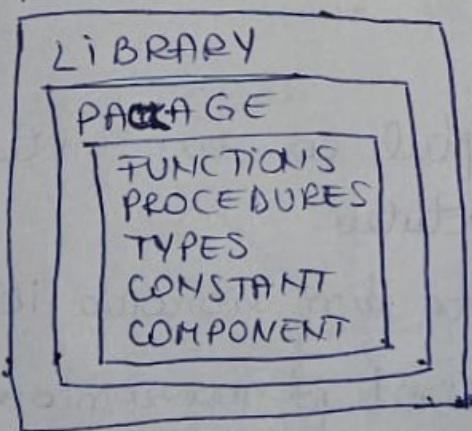
- std-logic-arith = conține operații aritmetice și de comparație ce se pot aplica asupra operatorilor cu semn sau fără semn.



- std - logic - signed
- std - logic - unsigned

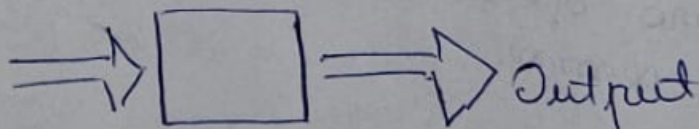
= sunt pachete care realizează operații cu vectori de semnale (std - logic - vector). Pachetele standard din bibliotecă "std" conțin funcții de tip "i/o text" și sunt incluse automat înău a și necesară specificarea explicită a acestuia la crearea unui program.

În cadrul unei biblioteci sunt create pachete (PACKAGE) care pot conține funcții, proceduri, componente, constante, tipuri.



### DECLARAREA ENTITĂȚII (ENTITY)

Entitatea reprezintă o abstractizare a unui sistem complet la nivelul de porturi intrare / ieșire. Elementele dintr-o entitate pot fi vizibile și celorlalte entități asociate acesteia.





La o entitate pot fi asociate mai multe arhitecturi prin clauza CONFIGURATION, dar în procesul de sinteză la un moment dat entității îi este asociată o singură arhitectură.

ENTITY nume\_entitate;

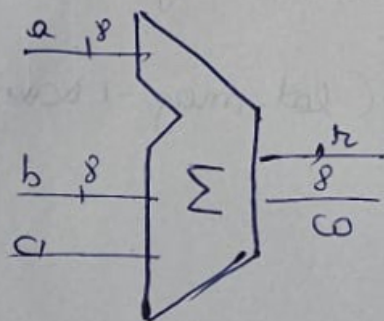
GENERIC ~~etc~~ (lista\_parametrilor\_generici);

PORT (lista\_porturi\_intrare\_iesire);

END nume\_entitate;

În cadrul entității se poate declara o listă de parametri generici utilizată în cadrul arhitecturii asociate acesteia.

Listă de porturi reprezintă semnalele de intrare - ieșire la modulul digital în cadrul arhitecturii.



ENTITY sum\_86 is

PORT(a,b: IN std\_logic\_vector(7 DOWNTO 0);

c1: IN std\_logic;

r: OUT std\_logic\_vector(7 DOWNTO 0);

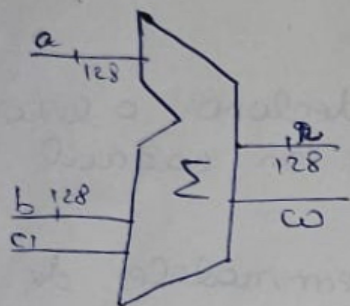
co: OUT std\_logic);

END sum\_86;

std\_logic\_vector declară o magistrală de semnale.

liste parametrilor generici realizarea  
transmiterea unor caracteristici entitatilor  
arhitecturilor asociate acestora.

Acesti parametri repr. inform. statice sub  
forma unei constante ce nu pot fi modificate  
in urma procesului de sinteza.



latime\_mag := 128

ENTITY sum\_M IS

GENERIC (lat\_mag := 128);

PORT (a, b : IN std\_logic\_vector (lat\_mag - 1 DOWNTO 0),  
ci : IN std\_logic.

s : OUT std\_logic\_vector (lat\_mag - 1 DOWNTO 0),  
co : OUT std\_logic);

END sum\_M;