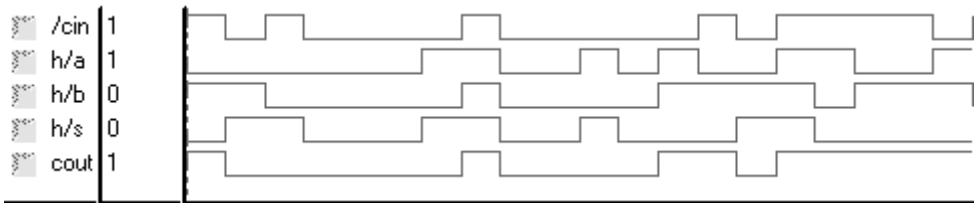
UNIVERSIDAD NACIONAL MAYOR DE SAN MARCOS
FACULTAD DE INGENIERIA ELECTRONICA
ESCUELA DE ELECTRONICA

1.-Diseñe un ckto full_ad der utilizando cualquier estilo de l VHDL.

FA.vhd :

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;

entity FA is
    Port ( Cin : in std_logic;
           A : in std_logic;
           B : in std_logic;
           S : out std_logic;
           Cout : out std_logic);
end FA;

architecture Behavioral of FA is
begin
S<=A Xor B Xor Cin;
Cout<=(A and B) or (A and Cin) or(B and Cin);
end Behavioral;
```



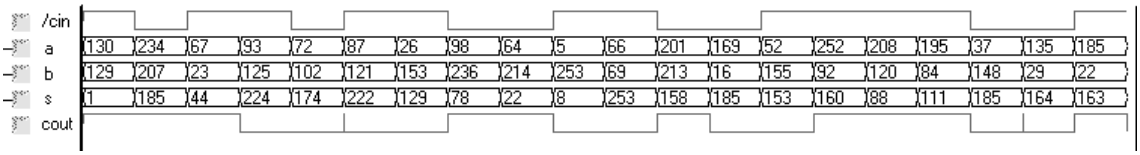
2.- Desñe un sumador-restador de 8 bits utilizando sumadores de 2 bits(full-adder) utilice el estilo estructural.

Suma_resta.vhd :

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;

entity sum_rest is
    Port ( Cin : in std_logic;
           A : in std_logic_vector(7 downto 0);
           B : in std_logic_vector(7 downto 0);
           S : out std_logic_vector(7 downto 0);
           Cout : out std_logic);
end sum_rest;

architecture Estructural of sum_rest is
   component fa
    Port ( Cin : in std_logic;
           A : in std_logic;
           B : in std_logic;
           S : out std_logic;
           Cout : out std_logic);
   end component;
signal C,P:std_logic_vector(7 downto 0);
begin
P(0)<=Cin Xor B(0);
P(1)<=Cin Xor B(1);
P(2)<=Cin Xor B(2);
```

Elver Yoel Ocmin Grandez
yoelocmin@hotmail.com
http://proyectos-fie-tk

```
P(3)<=Cin Xor B(3);
P(4)<=Cin Xor B(4);
P(5)<=Cin Xor B(5);
P(6)<=Cin Xor B(6);
P(7)<=Cin Xor B(7);
S_R00:fa port MAp(Cin,A(0),P(0),S(0),C(0));
suma_resta: For i in 0 to 6 Generate
             S_R:fa port map(C(i),A(i+1),P(i+1),S(i+1),C(i+1));
end generate;
      Cout<=C(7);
end Estructural;
```

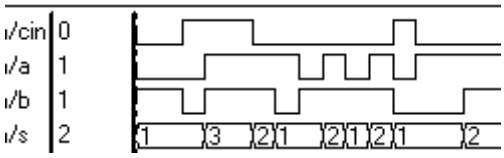


3.-

Usa Suma,Suma2,Suma3,Suma4,Incr5:

Suma.vhd:

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;

entity suma is
    Port ( Cin : in std_logic;
           A : in std_logic;
           B : in std_logic;
           S : out std_logic_vector(1 downto 0));
end suma;

architecture Behavioral of suma is
   component fa
    Port ( Cin : in std_logic;
           A : in std_logic;
           B : in std_logic;
           S : out std_logic;
           Cout : out std_logic);
   end component;
begin
sum_es:fa port map(Cin,A,B,S(0),S(1));
end Behavioral;
```

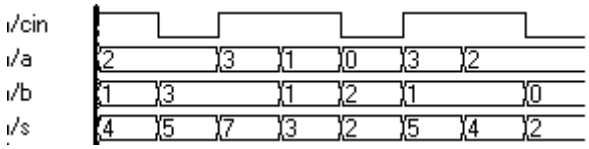


sumador2.vhd:

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;

entity sumador2 is
    Port ( Cin : in std_logic;
           A : in std_logic_vector(1 downto 0);
           B : in std_logic_vector(1 downto 0);
           S : out std_logic_vector(2 downto 0)
                 );
```

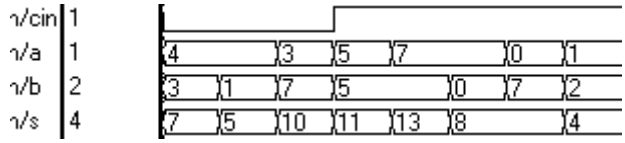
Elver Yoel Ocmin Grandez
yoelocmin@hotmail.com
http://proyectos-fie-tk

```
end sumador2;

architecture estructural of sumador2 is
   component fa
    Port ( Cin : in std_logic;
           A : in std_logic;
           B : in std_logic;
           S : out std_logic;
           Cout : out std_logic);
   end component;
signal C:std_logic_vector(1 downto 0):="00";
begin
suma1:fa port MAp(Cin,A(0),B(0),S(0),C(0));
suma2:fa port MAp(C(0),A(1),B(1),S(1),S(2));
end estructural;
```



sumador3.vhd:

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;

entity sumador3 is
    Port ( Cin : in std_logic;
           A : in std_logic_vector(2 downto 0);
           B : in std_logic_vector(2 downto 0);
           S : out std_logic_vector(3 downto 0)
                 );
end sumador3;

architecture estructural of sumador3 is
   component fa
    Port ( Cin : in std_logic;
           A : in std_logic;
           B : in std_logic;
           S : out std_logic;
           Cout : out std_logic);
   end component;
signal C:std_logic_vector(2 downto 0):="000";
begin
pri: fa port MAp(Cin,A(0),B(0),S(0),C(0));
suma: For i in 0 to 1 Generate
            Sum:fa port map(C(i),A(i+1),B(i+1),S(i+1),C(i+1));
end generate;
S(3)<=C(2);
end estructural;
```



sumador4.vhd:

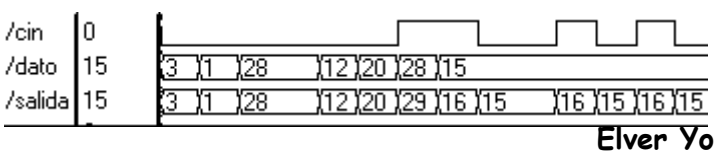
```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;
```

Elver Yoel Ocmin Grandez
yoelocmin@hotmail.com
http://proyectos-fie-tk

```
entity sumador4 is
    Port ( Cin : in std_logic;
           A : in std_logic_vector(3 downto 0);
           B : in std_logic_vector(3 downto 0);
           S : out std_logic_vector(4 downto 0)
                        );
end sumador4;

architecture estructural of sumador4 is
   component fa
    Port ( Cin : in std_logic;
           A : in std_logic;
           B : in std_logic;
           S : out std_logic;
           Cout : out std_logic);
   end component;
signal C:std_logic_vector(3 downto 0):="0000";
begin
pri: fa port MAp(Cin,A(0),B(0),S(0),C(0));
suma: For i in 0 to 2 Generate
             Sum:fa port map(C(i),A(i+1),B(i+1),S(i+1),C(i+1));
end generate;
S(4)<=C(3);
end estructural;
```

incr5.vhd:

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;

entity INCR5 is
    Port (Cin:in std_logic;
       Dato:in std_logic_vector(4 downto 0);
       Salida:out std_logic_vector(5 downto 0)
       );
end INCR5;

architecture Estructural of INCR5 is
   component fa
    Port ( Cin : in std_logic;
           A : in std_logic;
           B : in std_logic;
           S : out std_logic;
           Cout : out std_logic);
   end component;
signal C:std_logic_vector(4 downto 0):="00000";
begin
pri: fa port MAp(Cin,dAto(0),'0',Salida(0),C(0));
suma: For i in 0 to 3 Generate
             Sum:fa port map(C(i),dAto(i+1),'0',Salida(i+1),C(i+1));
end generate;
Salida(5)<=C(4);
end Estructural;
```



Elver Yoel Ocmin Grandez
yoelocmin@hotmail.com
http://proyectos-fie-tk

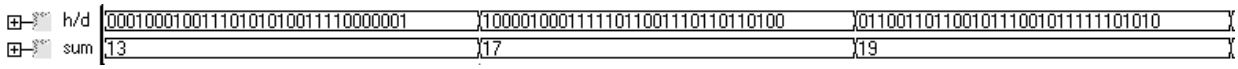
Y el programa final es:

PREG3.VHD :

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;

entity Preg3 is
    Port ( D : in std_logic_vector(31 downto 0);
           sum : out std_logic_vector(5 downto 0));
end Preg3;

architecture estructural of Preg3 is
   component suma
    Port ( Cin : in std_logic;
           A : in std_logic;
           B : in std_logic;
           S : out std_logic_vector(1 downto 0)
             );
   end component;
   component sumador2
    Port ( Cin : in std_logic;
           A : in std_logic_vector(1 downto 0);
           B : in std_logic_vector(1 downto 0);
           S : out std_logic_vector(2 downto 0)
                    );
   end component;
   component sumador3
    Port ( Cin : in std_logic;
           A : in std_logic_vector(2 downto 0);
           B : in std_logic_vector(2 downto 0);
           S : out std_logic_vector(3 downto 0)
                   );
   end component;
   component sumador4
    Port ( Cin : in std_logic;
           A : in std_logic_vector(3 downto 0);
           B : in std_logic_vector(3 downto 0);
           S : out std_logic_vector(4 downto 0)
                   );
   end component;
   component incr5
    Port (Cin:in std_logic;
        Dato:in std_logic_vector(4 downto 0);
        Salida:out std_logic_vector(5 downto 0)
        );
   end component;
signal S0,s1,s2,s3,s4,s5,s6,s7:std_logic_vector(1 downto 0):="00";
signal s8,s9,s10,s11:std_logic_vector(2 downto 0):="000";
signal s12,s13:std_logic_vector(3 downto 0):="0000";
signal s14:std_logic_vector(4 downto 0):="00000";
signal s15:std_logic_vector(5 downto 0):="000000";
begin
D2y0: suma port MAp(D(2),D(1),D(0),S0);
D5y3: suma port MAp(D(5),D(4),D(3),S1);
D8y6: suma port MAp(D(8),D(7),D(6),S2);
D11y9: suma port MAp(D(11),D(10),D(9),S3);
D14y12: suma port MAp(D(14),D(13),D(12),S4);
D17y15: suma port MAp(D(17),D(16),D(15),S5);
D20y18: suma port MAp(D(20),D(19),D(18),S6);
D23y21: suma port MAp(D(23),D(22),D(21),S7);
```

Elver Yoel Ocmin Grandez
yoelocmin@hotmail.com
http://proyectos-fie-tk

```
DS01y24: sumador2 port MAp(D(24),S0,S1,s8);
DS23y25: sumador2 port MAp(D(25),S2,S3,s9);
DS45y26: sumador2 port MAp(D(26),S4,S5,s10);
DS57y27: sumador2 port MAp(D(27),S6,S7,s11);

DS8y9y28: sumador3 port MAp(D(28),S8,S9,s12);
DS10y11y29: sumador3 port MAp(D(29),S10,S11,s13);

DS12y13y30: sumador4 port MAp(D(30),S12,S13,s14);

DS14y31: incr5 port MAp(D(31),S14,S15);
sum<=S15;
end estructural;
```



4:

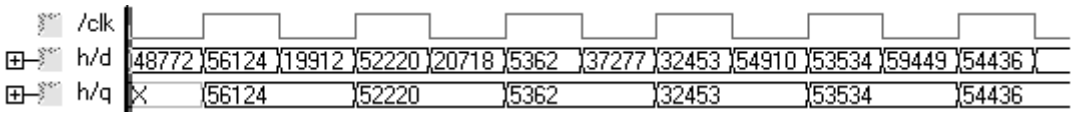
Usa a **FDD.vhd** :

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;

entity FFD is
    Port ( CLK : in std_logic;
           D : in std_logic;
           Q : out std_logic);
end FFD;

architecture Behavioral of FFD is
begin
process (Clk)
begin
IF CLK='1' THEN Q<=d; END IF;
end process;
end Behavioral;
```

y **PREG4.VHD** :

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;

entity preg4 is
    Port ( clk : in std_logic;
           D : in std_logic_vector(15 downto 0);
           Q : OUT std_logic_vector(15 downto 0));
end preg4;

architecture Despl16bits of preg4 is
component FFD
    Port ( CLK : in std_logic;
           D : in std_logic;
           Q : out std_logic);
end component;
begin
registro16: for i in 0 to 15 generate
par_par:FFD Port Map(Clk,D(i),Q(i));
end generate registro16;
end Despl16bits;
```

Elver Yoel Ocmin Grandez
yoelocmin@hotmail.com
http://proyectos-fie-tk



5.- Crear una entidad y arquitectura donde el usuario tenga la capacidad de generar un registro de "n" bits.

A nuestro archivo lo llamaremos **registro.vhd** :

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;

entity registro is
generic(numero:positive);
   Port ( clk : in std_logic;
          D : in std_logic_vector((numero-1) downto 0);
          Q : out std_logic_vector((numero-1) downto 0));
end registro;

architecture Behavioral of registro is
component FFD
   Port ( CLK : in std_logic;
          D : in std_logic;
          Q : out std_logic);
end component;
begin
registros: for i in 0 to numero-1 generate
par_par:FFD Port Map(Clk,D(i),Q(i));
end generate registros;
end Behavioral;
```

y este funciona con el archivo anterior **FFD.vhd**

Y para probar su funionamiento, crearemos un registro de 10 bits.

Y lo llamaremos **registro10.vhd**

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;

entity registro10 is
    Port ( clk : in std_logic;
           D : in std_logic_vector(10 downto 0);
           Q : out std_logic_vector(10 downto 0));
end registro10;

architecture estructural of registro10 is
component registro
generic(numero:positive);
    Port ( clk : in std_logic;
           D : in std_logic_vector((numero-1) downto 0);
           Q : out std_logic_vector((numero-1) downto 0));
```

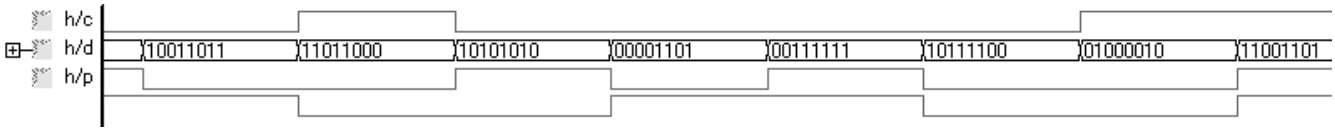
Elver Yoel Ocmin Grandez
yoelocmin@hotmail.com
http://proyectos-fie-tk

```
end component;
begin
regis:registro generic map(10) Port MAp(clk,D,Q);
end estructural;
```

Su simulacion es:

6:

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;

entity preg6 is
    Port ( C : in bit;
           D : in std_logic_vector(7 Downto 0);
           P : out std_logic);
end preg6;

architecture Estructural of preg6 is

begin
process(D)
variable num_bits : integer;
variable paridad: integer;
begin
 num_bits:=0;
 for i in 0 to 7 loop
 if D(i) ='1' then
    num_bits := num_bits +1;
 end if;
 end loop;
 paridad:=(num_bits mod 2);
 case C is
 when '1' =>if paridad=0 then P<='0'; else p<='1'; end if;
 when '0' =>if paridad=0 then P<='1'; else p<='0'; end if;
 end case;
 end process;
end Estructural;
```



Elver Yoel Ocmin Grandez
yoelocmin@hotmail.com
http://proyectos-fie-tk