INSTITUTO TECNOLOGICO DE ESTUDIOS SUPERIORES DE MONTERREY

LABORATORIO DE SISTEMAS DIGITALES AVANZADOS

PROF. EMILIO CABALLERO

HECTOR JAVIER PEQUEÑO CHAIREZ

A01246364

SUBTRACTOR 4 BITS.

LINKS GITHUB:

Subtractor.vhdl:
https://github.com/HectorPequeno06/HectorPequeno06/blob/master/Subtractor4bits/Subtractor.vhdl

SubtractorTb.vhdl:
https://github.com/HectorPequeno06/HectorPequeno06/blob/master/Subtractor4bits/SubtractorTb.vhdl

Subtractor 4 bits Tb:

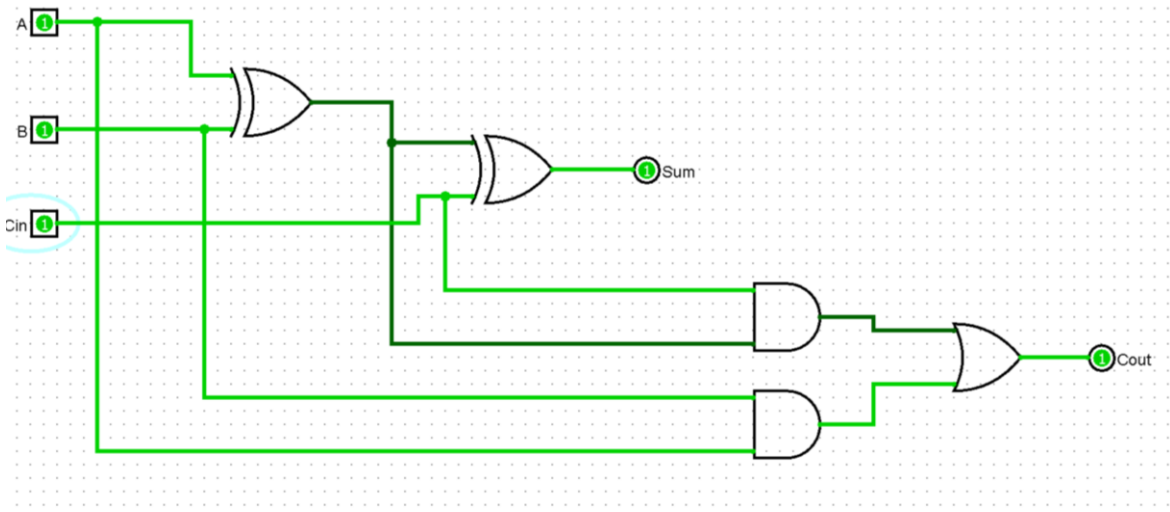
https://github.com/HectorPequeno06/HectorPequeno06/blob/master/Subtractor4bits/Subtractor4BitsTb.vhdl

Subtractor 4 bits vhdl:

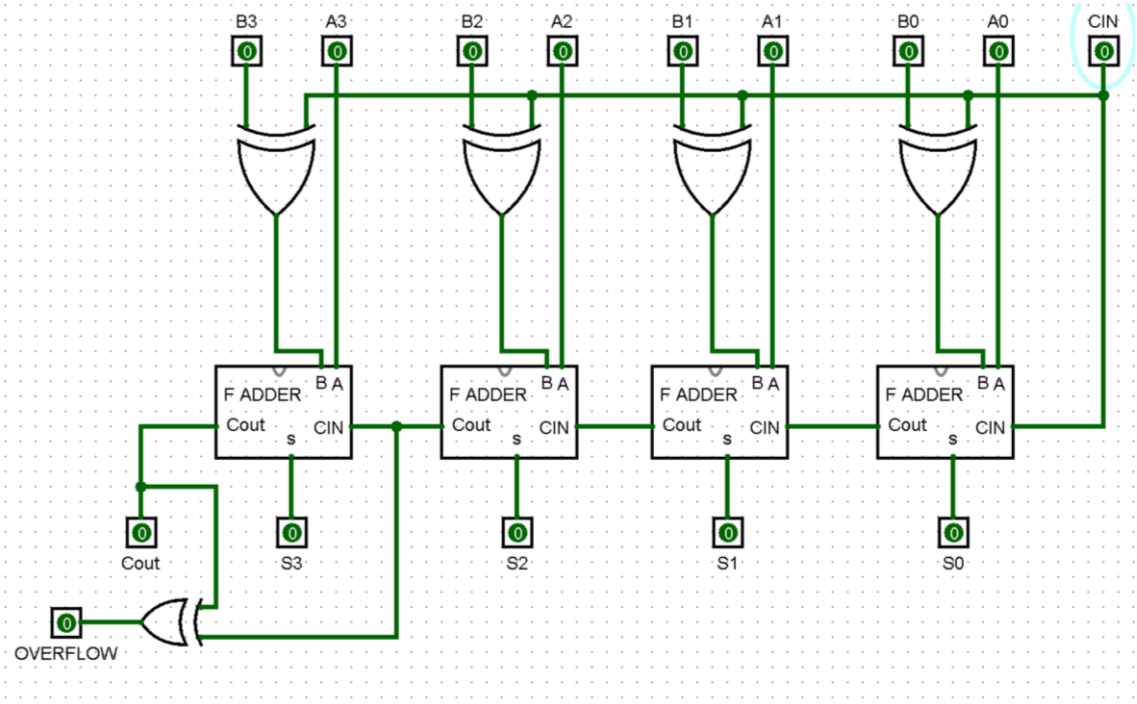
https://github.com/HectorPequeno06/HectorPequeno06/blob/master/Subtractor4bits/Subtractor4bits.vhdl

DISEÑOS LOGISIM.

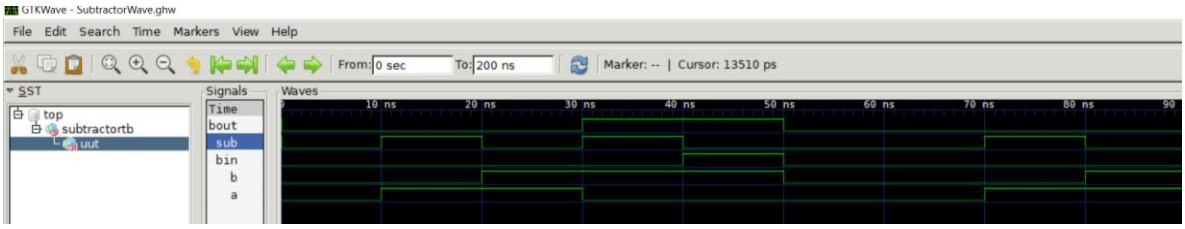
FULL ADDER.



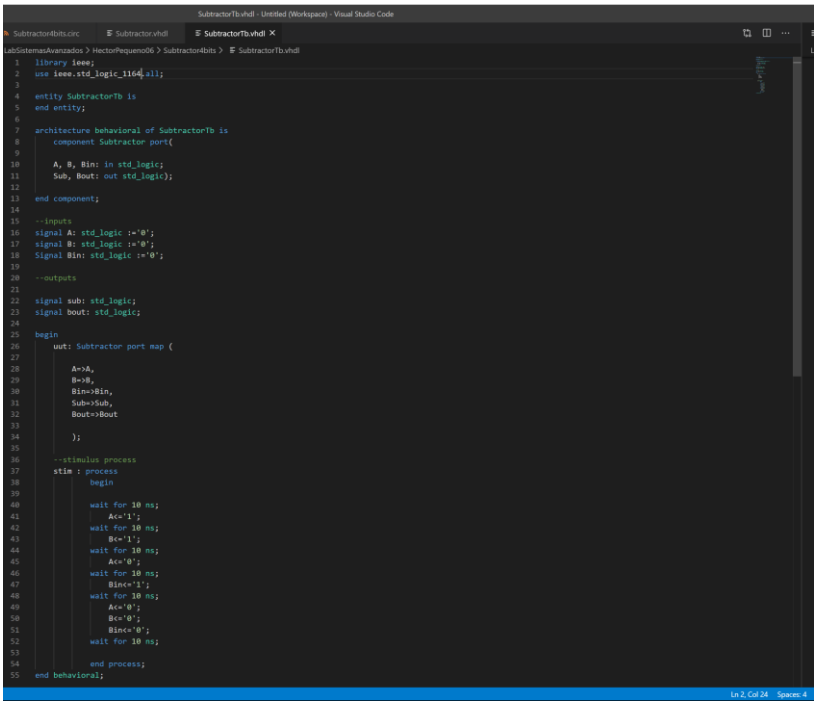
SUBTRACTOR 4 BITS



Simulación Subtractor VHDL

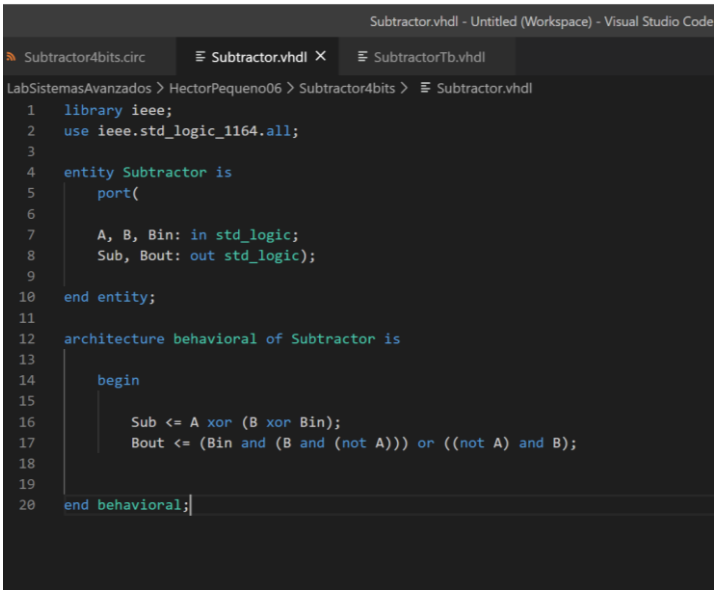


Código TB:



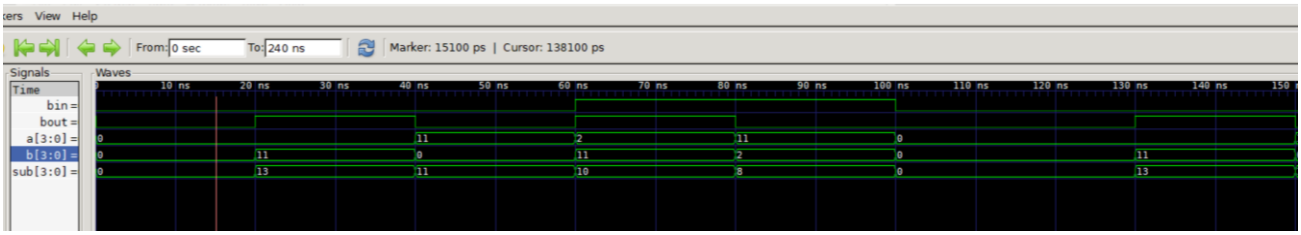
```vhdl
library ieee;
use ieee.std_logic_1164.all;

entity SubtractorTb is
end entity;

architecture behavioral of SubtractorTb is
    component Subtractor port(

    A, B, Bin: in std_logic;
    Sub, Bout: out std_logic);

end component;

--inputs
signal A: std_logic :='0';
signal B: std_logic :='0';
Signal Bin: std_logic :='0';

--outputs

signal sub: std_logic;
signal bout: std_logic;

begin
    uut: Subtractor port map (

        A=>A,
        B=>B,
        Bin=>Bin,
        Sub=>Sub,
        Bout=>Bout

        );

    --stimulus process
    stim : process
            begin

            wait for 10 ns;
                A<='1';
            wait for 10 ns;
                B<='1';
            wait for 10 ns;
                A<='0';
            wait for 10 ns;
                Bin<='1';
            wait for 10 ns;
                A<='0';
                B<='0';
                Bin<='0';
            wait for 10 ns;

            end process;
end behavioral;
```

Código Fuente Subtractor:



```vhdl
library ieee;
use ieee.std_logic_1164.all;

entity Subtractor is
    port(

    A, B, Bin: in std_logic;
    Sub, Bout: out std_logic);

end entity;

architecture behavioral of Subtractor is

    begin

        Sub <= A xor (B xor Bin);
        Bout <= (Bin and (B and (not A))) or ((not A) and B);


end behavioral;
```

Simulación Subtractor4Bits GTK wave:



Código TestBench:

```
    end entity;

architecture behavioral of Subtractor4BitsTb is
component Subtractor4bits port(

    A: in std_logic_vector(3 downto 0);
    B: in std_logic_vector(3 downto 0);
    Bin: in std_logic;
    Bout: out std_logic;
    Sub: out std_logic_vector(3 downto 0)


);
end component;

--inputs
SIGNAL A : std_logic_vector(3 downto 0) := "0000";
SIGNAL B : std_logic_vector(3 downto 0) := "0000";
SIGNAL Bin : std_logic := '0';
--OUTPUTS
SIGNAL Bout : std_logic;
SIGNAL Sub : std_logic_vector(3 downto 0);

begin
    uut: Subtractor4bits Port map(
        A => A,
        B => B,
        Bin => Bin,
        Bout => Bout,
        Sub => Sub
    );

--stimulus process
stim: process
        begin
            wait for 20 ns;
            A<="0000";
            B<="1011";
            wait for 20 ns;
            A<="1011";
            B<="0000";
            wait for 20 ns;
            A<="0010";
            B<="1011";
            Bin<='1';
            wait for 20 ns;
            A<="1011";
            B<="0010";
            Bin<='1';
            wait for 20 ns;
            A<="0000";
            B<="0000";
            Bin<='0';
            wait for 10 ns;
```

Código Fuente:

```
library ieee;
use ieee.std_logic_1164.all;

entity Subtractor4bits is
    port(
        A: in std_logic_vector(3 downto 0);
        B: in std_logic_vector(3 downto 0);
        Bin: in std_logic;
        Bout: out std_logic;
        Sub: out std_logic_vector(3 downto 0)
        );

end Subtractor4bits;

architecture behavioral of Subtractor4bits is
    component Subtractor
    port(
        A, B, Bin: in std_logic;
        Bout, Sub : out std_logic
    );
end component;
signal Bouts: std_logic_vector(2 downto 0);
begin
    --concurrent statements

    SUB0: Subtractor port map(A(0), B(0), Bin ,Bouts(0) ,Sub(0));
    SUB1: Subtractor port map(A(1), B(1), Bouts(0), Bouts(1), Sub(1));
    SUB2: Subtractor port map(A(2), B(2), Bouts(1), Bouts(2), sub(2));
    SUB3: Subtractor port map(A(3), B(3), Bouts(2), bout, Sub(3));


end behavioral;
```