ceti

CENTRO DE ENSEÑANZA
TÉCNICA INDUSTRIAL

Materia:

Diseño de Circuitos Integrados Digitales CMOS I I

Nombre de la Tarea:

Practica 2 – Adder Substractor

Nombre del estudiante:

Gonzalez Diaz Ariel

Nombre del Profesor:

Jose Roberto Reyes Baron

Fecha:

28 de Agosto del 2023

# Introducción

In this practice we need to create a full Subtractor using our FPGA device and compile with VHDL language, for this is necessary have knowledge about of what is a full Subtractor and Full adder  how we made this configuration:

## Adder – substractor

An "adder-subtractor" is a type of digital circuit designed to perform both addition and subtraction operations on binary numbers within a computer or digital system. This circuit combines the functionalities of an adder and a subtractor, allowing for efficient and flexible arithmetic operations.

Addition Operation:

In the adder mode, the circuit functions like a regular binary adder. It takes two binary numbers, typically represented as binary vectors, and adds them together. The binary inputs are usually called "A" and "B". The adder then produces the sum of these inputs, known as the "sum output." Additionally, the adder might generate a "carry output," which indicates whether a carry occurred during addition.

Subtraction Operation:

In the subtractor mode, the circuit operates as a binary subtractor. It takes two binary inputs, again referred to as "A" and "B," and performs subtraction by finding the difference between these inputs. To achieve this, the subtractor often utilizes the concept of two's complement. It computes the two's complement of the second input ("B") and then adds it to the first input ("A"). The result is the difference between the two inputs. The circuit might also provide an output that indicates whether a borrow occurred during subtraction.

Control Signal:

The key feature of an adder-subtractor is the control signal. This signal determines whether the circuit performs addition or subtraction. When the control signal is set to a specific value, the circuit operates as an adder. When the control signal has a different value, the circuit switches to subtraction mode.

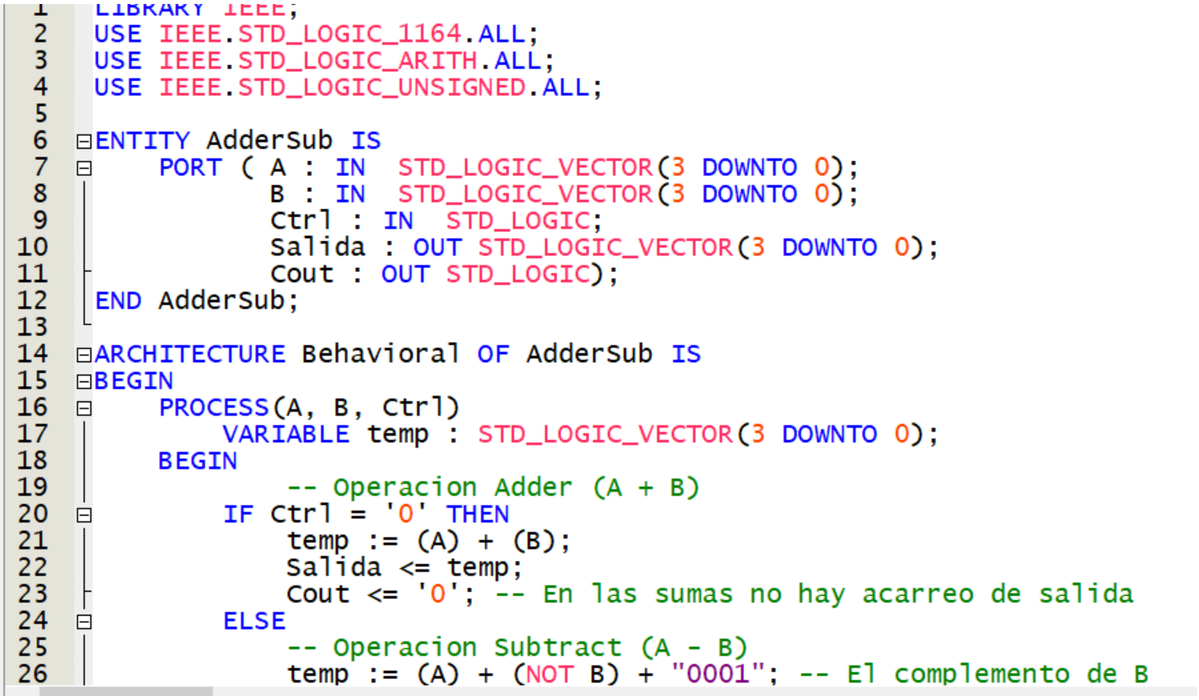
Applications:

Adder-subtractors are essential components in various digital systems, including processors, calculators, and arithmetic logic units (ALUs). These circuits play a vital role in performing arithmetic operations on binary data within computers. The ability to switch between addition and subtraction modes efficiently contributes to the overall versatility and computational capabilities of digital systems.

In summary, an adder-subtractor is a digital circuit that integrates both adder and subtractor functionalities. It operates based on a control signal, allowing it to perform addition or subtraction operations on binary inputs. This circuit is a fundamental building block in digital arithmetic and is widely used in various computing and digital systems.

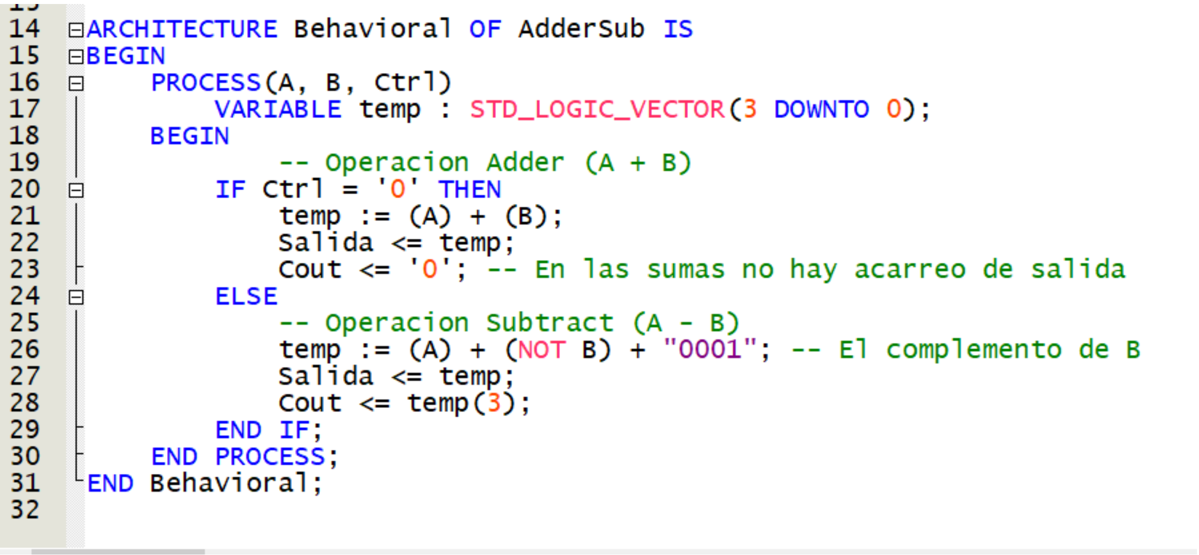
# Process

once we have understood how to create our adder subtractor we can start with the development, in which following the book we have the understanding of how you should compose it, firstly we create an entity which will allow us to create the necessary ports, in this case we need two inputs which will be 4-bit vectors and in addition to this a control bit that will let us control if it is a subtractor or an adder, and we have an output carry bit and one of the output value

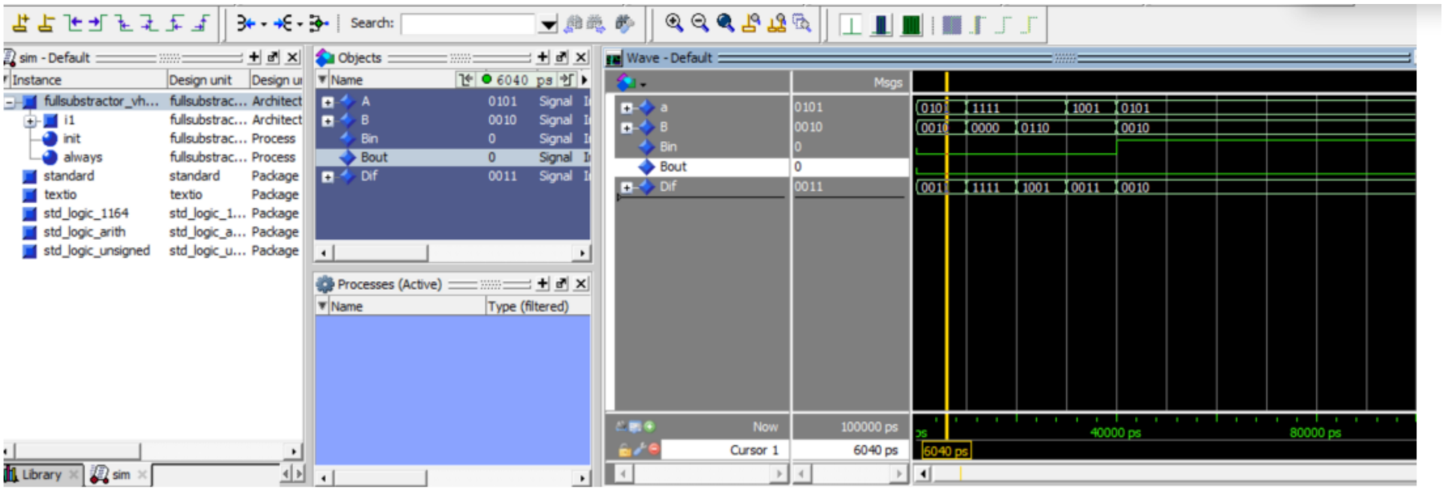


```
LIBRARY IEEE;
USE IEEE.STD_LOGIC_1164.ALL;
USE IEEE.STD_LOGIC_ARITH.ALL;
USE IEEE.STD_LOGIC_UNSIGNED.ALL;

ENTITY AdderSub IS
    PORT ( A : IN  STD_LOGIC_VECTOR(3 DOWNTO 0);
           B : IN  STD_LOGIC_VECTOR(3 DOWNTO 0);
           Ctrl : IN  STD_LOGIC;
           Salida : OUT STD_LOGIC_VECTOR(3 DOWNTO 0);
           Cout : OUT STD_LOGIC);
END AdderSub;

ARCHITECTURE Behavioral OF AdderSub IS
BEGIN
    PROCESS(A, B, Ctrl)
        VARIABLE temp : STD_LOGIC_VECTOR(3 DOWNTO 0);
    BEGIN
            -- Operacion Adder (A + B)
        IF Ctrl = '0' THEN
            temp := (A) + (B);
            Salida <= temp;
            Cout <= '0'; -- En las sumas no hay acarreo de salida
        ELSE
            -- Operacion Subtract (A - B)
            temp := (A) + (NOT B) + "0001"; -- El complemento de B
```

Now we will create the behavioral architecture of which we already know how our circuit is going to work, which is the following



```vhdl
ARCHITECTURE Behavioral OF AdderSub IS
BEGIN
    PROCESS(A, B, Ctrl)
        VARIABLE temp : STD_LOGIC_VECTOR(3 DOWNTO 0);
    BEGIN
            -- Operacion Adder (A + B)
        IF Ctrl = '0' THEN
            temp := (A) + (B);
            Salida <= temp;
            Cout <= '0'; -- En las sumas no hay acarreo de salida
        ELSE
            -- Operacion Subtract (A - B)
            temp := (A) + (NOT B) + "0001"; -- El complemento de B
            Salida <= temp;
            Cout <= temp(3);
        END IF;
    END PROCESS;
END Behavioral;
```



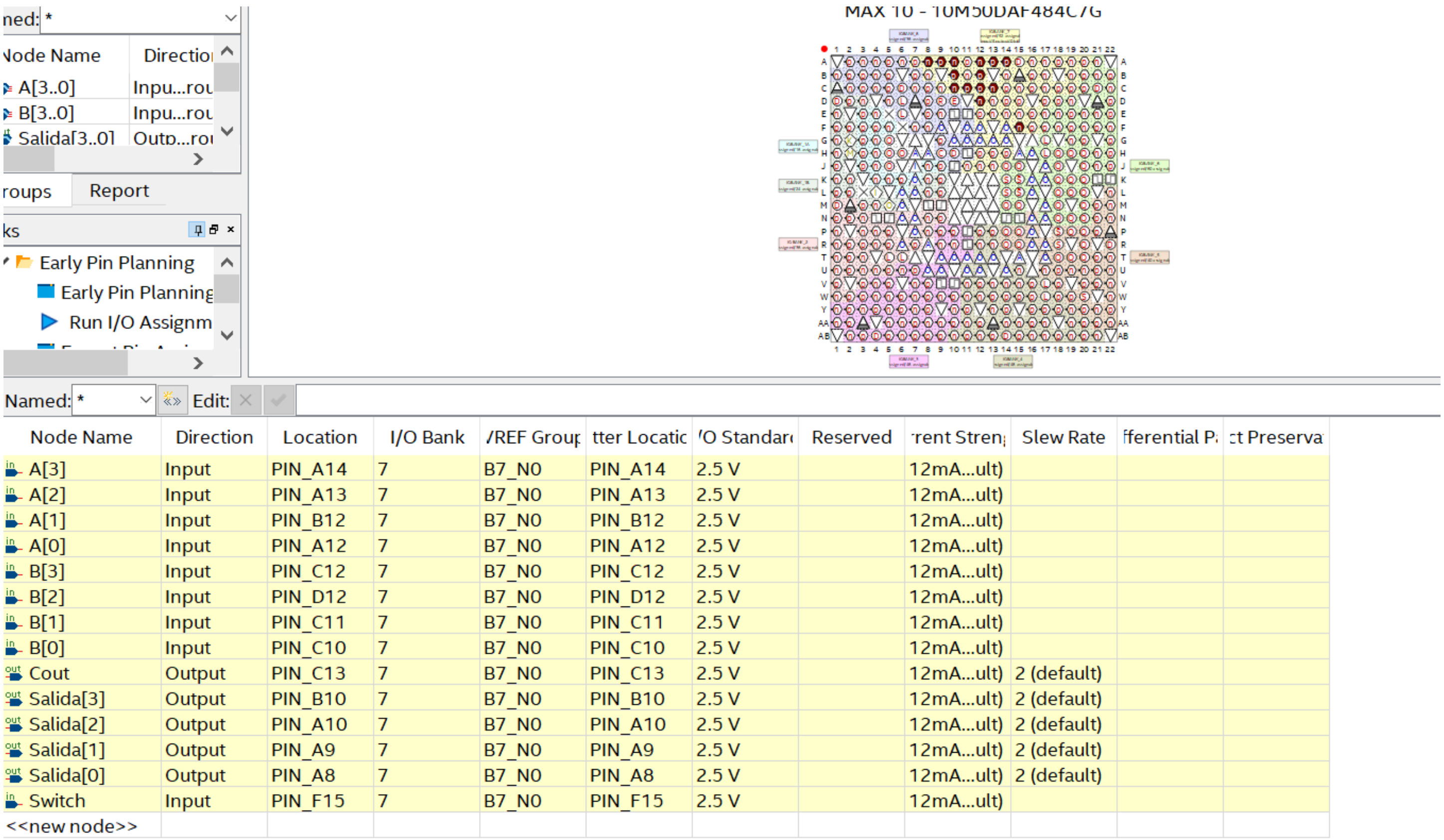
| Instance | Design unit | Design ui |
|---|---|---|
| fullsubstractor_vh... | fullsubstrac... | Architect |
| i1 | fullsubstrac... | Architect |
| init | fullsubstrac... | Process |
| always | fullsubstrac... | Process |
| standard | standard | Package |
| textio | textio | Package |
| std_logic_1164 | std_logic_1... | Package |
| std_logic_arith | std_logic_a... | Package |
| std_logic_unsigned | std_logic_u... | Package |

| Name | 6040 ps | |
|---|---|---|
| A | 0101 | Signal |
| B | 0010 | Signal |
| Bin | 0 | Signal |
| Bout | 0 | Signal |
| Dif | 0011 | Signal |

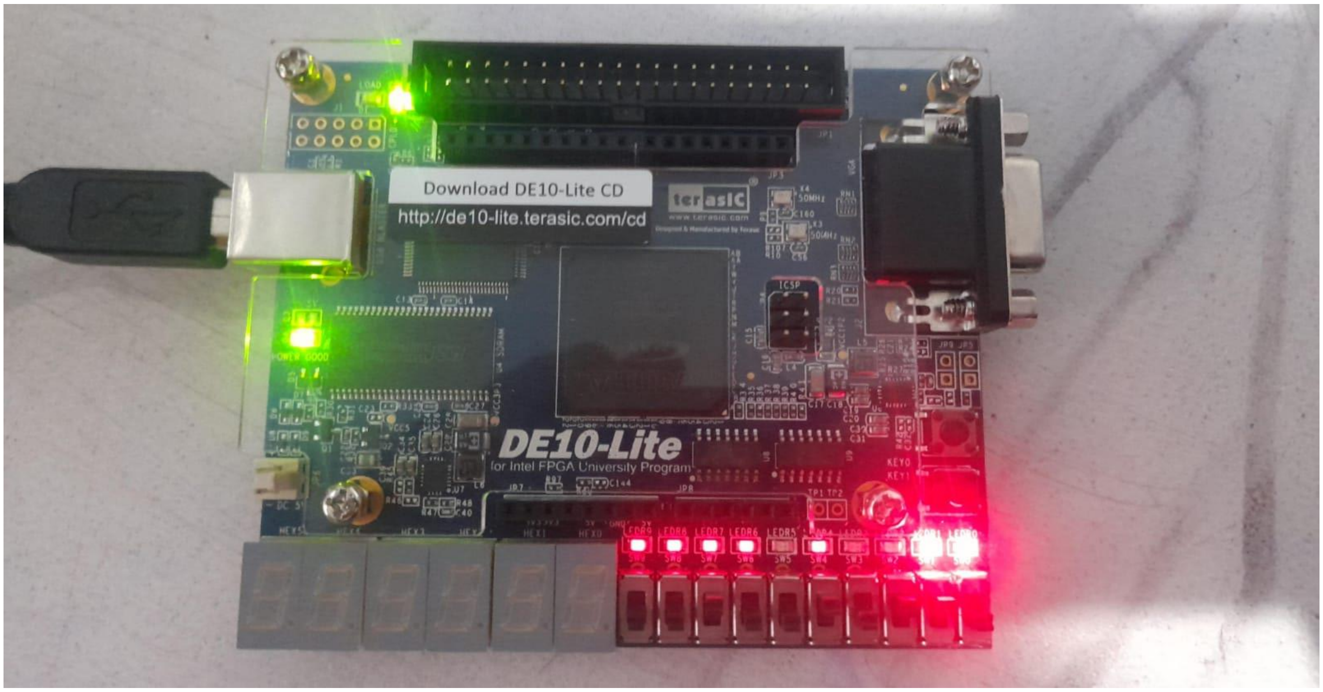
| Name | Type (filtered) |
|---|---|

| | Msgs |
|---|---|
| a | 0101 |
| B | 0010 |
| Bin | 0 |
| Bout | 0 |
| Dif | 0011 |
| Now | 100000 ps |
| Cursor 1 | 6040 ps |

Now  the finally step is assigned our inputs and outputs to the FPGA and recompile our program for the fisical test , we need go to the pin planner with Ctrl + Shift + N



| Node Name | Direction | Location | I/O Bank | /REF Group | tter Locatic | 'O Standar | Reserved | rent Stren | Slew Rate | fferential P | ct Preserva |
|---|---|---|---|---|---|---|---|---|---|---|---|
| A[3] | Input | PIN_A14 | 7 | B7_N0 | PIN_A14 | 2.5 V | | 12mA...ult) | | | |
| A[2] | Input | PIN_A13 | 7 | B7_N0 | PIN_A13 | 2.5 V | | 12mA...ult) | | | |
| A[1] | Input | PIN_B12 | 7 | B7_N0 | PIN_B12 | 2.5 V | | 12mA...ult) | | | |
| A[0] | Input | PIN_A12 | 7 | B7_N0 | PIN_A12 | 2.5 V | | 12mA...ult) | | | |
| B[3] | Input | PIN_C12 | 7 | B7_N0 | PIN_C12 | 2.5 V | | 12mA...ult) | | | |
| B[2] | Input | PIN_D12 | 7 | B7_N0 | PIN_D12 | 2.5 V | | 12mA...ult) | | | |
| B[1] | Input | PIN_C11 | 7 | B7_N0 | PIN_C11 | 2.5 V | | 12mA...ult) | | | |
| B[0] | Input | PIN_C10 | 7 | B7_N0 | PIN_C10 | 2.5 V | | 12mA...ult) | | | |
| Cout | Output | PIN_C13 | 7 | B7_N0 | PIN_C13 | 2.5 V | | 12mA...ult) | 2 (default) | | |
| Salida[3] | Output | PIN_B10 | 7 | B7_N0 | PIN_B10 | 2.5 V | | 12mA...ult) | 2 (default) | | |
| Salida[2] | Output | PIN_A10 | 7 | B7_N0 | PIN_A10 | 2.5 V | | 12mA...ult) | 2 (default) | | |
| Salida[1] | Output | PIN_A9 | 7 | B7_N0 | PIN_A9 | 2.5 V | | 12mA...ult) | 2 (default) | | |
| Salida[0] | Output | PIN_A8 | 7 | B7_N0 | PIN_A8 | 2.5 V | | 12mA...ult) | 2 (default) | | |
| Switch | Input | PIN_F15 | 7 | B7_N0 | PIN_F15 | 2.5 V | | 12mA...ult) | | | |
| <<new node>> | | | | | | | | | | | |


Download DE10-Lite CD
http://de10-lite.terasic.com/cd
terasIC
DE10-Lite
for Intel FPGA University Program

Once that we assignment our inputs and outputs in the FPGA we can programming the FPGA and executes the code.

# Conclusion

In this practice we were able to observe the operation of a complete subtractor of 4 bits and how to obtain the equations to make it work, in addition to understanding in a better way how to use our FPGA and have one of the multiple functions that it has and combinate with full adder.

https://www.electronics-tutorials.ws/combination/binary-subtractor.html

*Full Subtractor in Digital Logic*. (2017, October 10). GeeksforGeeks; GeeksforGeeks. https://www.geeksforgeeks.org/full-subtractor-in-digital-logic/

*Full Subtractor | Definition | Circuit Diagram | Truth Table | Gate Vidyalay*. (n.d.). https://www.gatevidyalay.com/full-subtractor/

*BYJU'S Exam Prep*. (2022). @Gradeupapp. https://byjusexamprep.com/full-subtractor-i

*Virtual Labs*. (2023). Vlabs.ac.in. https://de-iitr.vlabs.ac.in/exp/half-full-subtractor/theory.html