Digital Design Principles

Magnitude Comparator

Name: Rajkaran Singh Grewal

Student No: 8882386

Email: rgrewal2386@conestogac.on.ca

Objective:

- Use VHDL and Verilog to implement a simple magnitude comparator.

- Set up a project in Quartus II targeting your FPGA board.

Truth Table:

Table

Description automatically generated

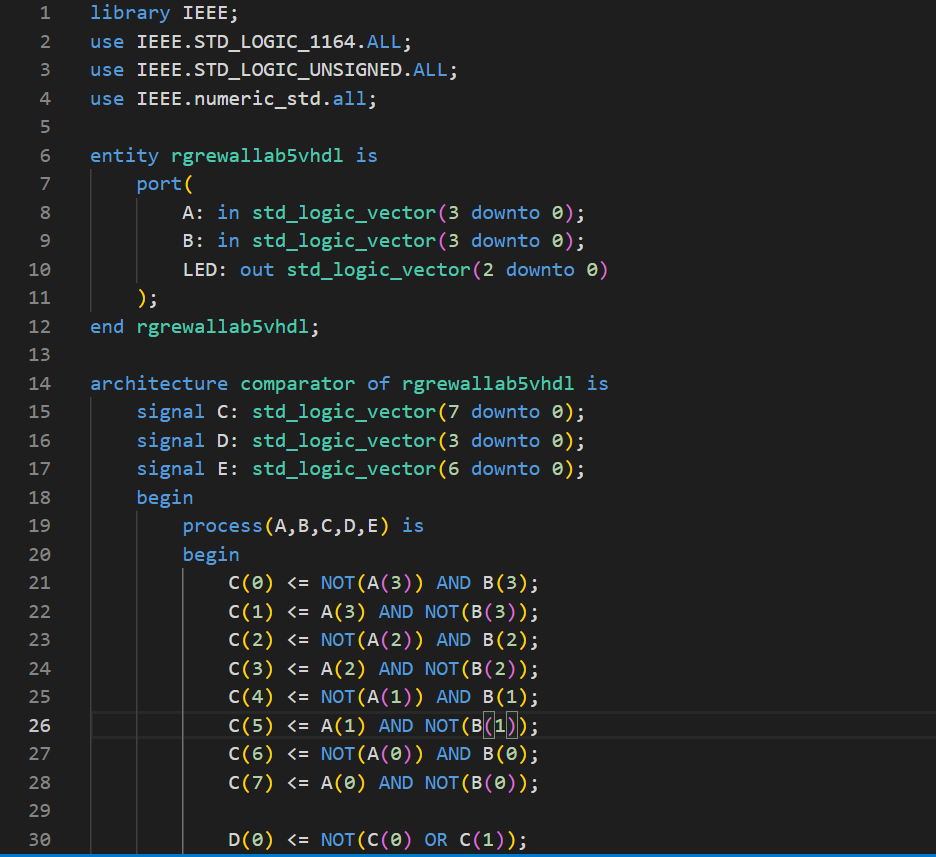
Logic Diagram:

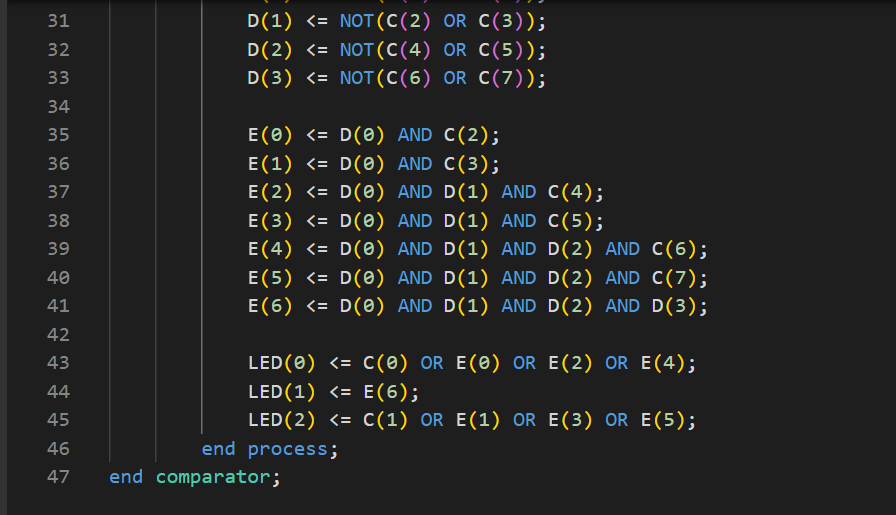
Diagram

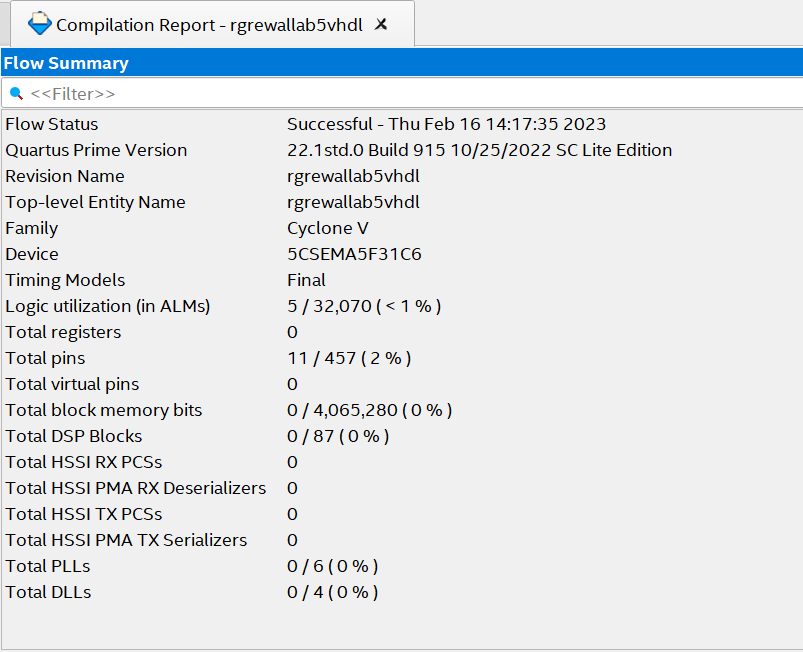
Description automatically generated

VHDL Introduction:

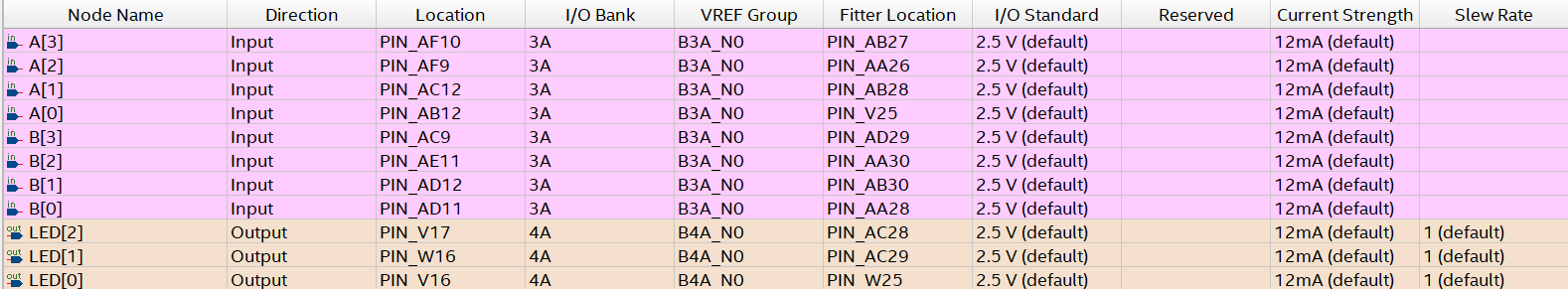
To make the program in VHDL we will need to initialize the library and entity. Within the entity we will state the 8 input switches and the 3 output LEDs. Within the architecture of the program, we will use 3 different array variables. Where the first array variable will have a total storage of 8 bits and will contain the either A’ AND B or A AND B’. The second array variable will store 4 bits where it will contain nor or the ascending order of the previous array. The final array will be 7 bits long and will be AND Gate which merge variable of the second array and some case the first array so that the logic structure will follow the truth table.

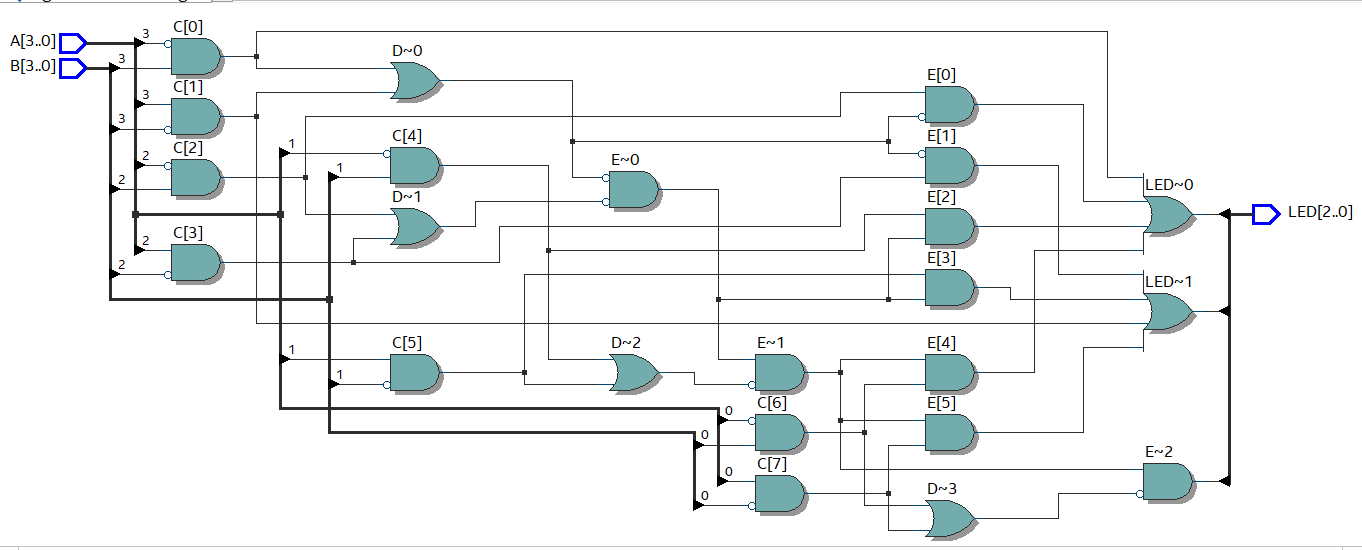
VHDL program screenshot:



VHDL compilation report:

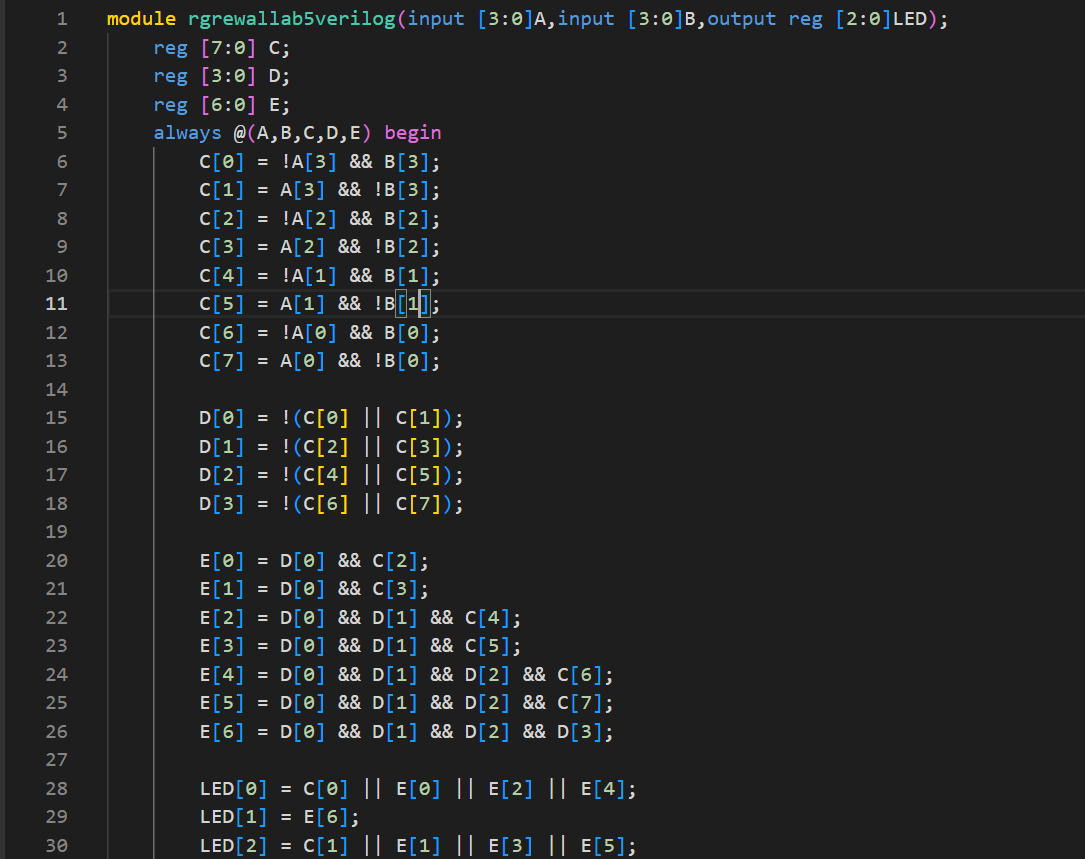
Pin Planner screenshot:



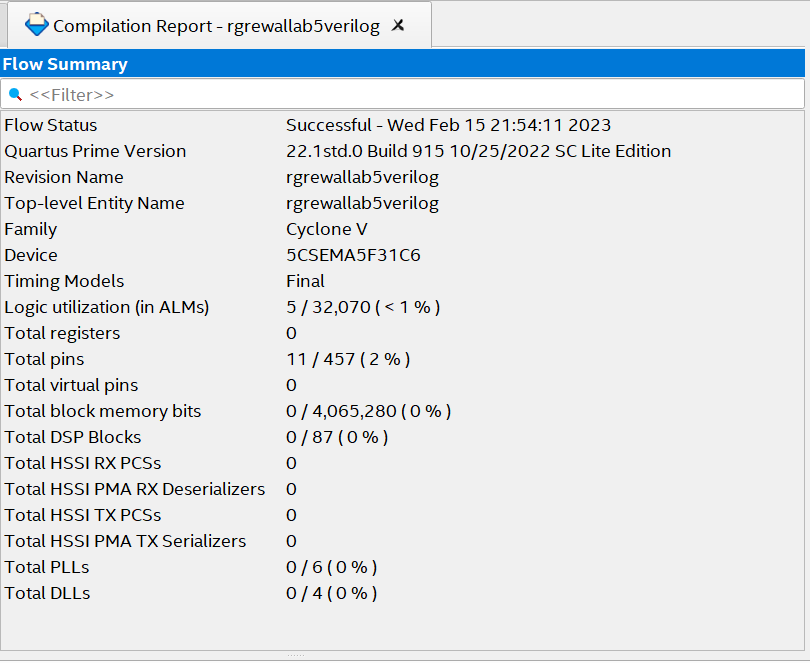
VHDL RTL View:

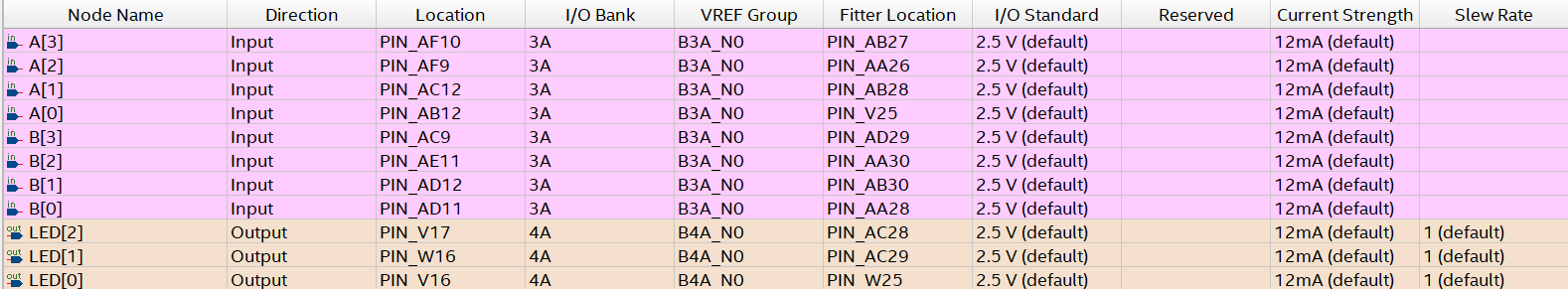
Verilog introduction:

For the program in Verilog we will need to initialize the module with the parameters having the three inputs and eight outputs. Within the module block we will use 3 different array variables. Where the first array variable will have a total storage of 8 bits and will contain the either A’ AND B or A AND B’. The second array variable will store 4 bits where it will contain nor or the ascending order of the previous array. The final array will be 7 bits long and will be AND Gate which merge variable of the second array and some case the first array so that the logic structure will follow the truth table.

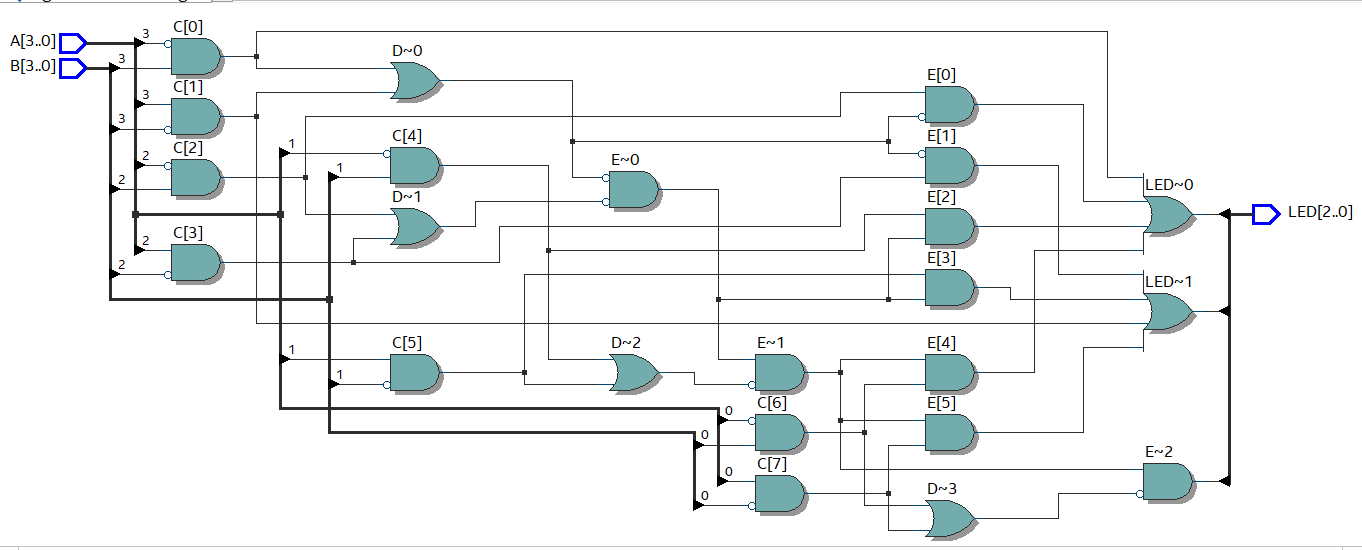
Verilog program screenshot:



Verilog compilation report:

Verilog pin planner screenshot:

Verilog RTL View:



Conclusion:

Thus we can conclude that by both the programming languages we by following the above truth table we can create a 4-bit magnitude comparator.

Reference:

<https://www.101computing.net/binary-comparators-using-logic-gates/>

<https://forums.ni.com/t5/Multisim-and-Ultiboard/4-bit-magnitude-comparator/td-p/3887290>