Digital Design Principles

Parking Control System

Name: Rajkaran Singh Grewal
Student No: 8882386
Email: rgrewal2386@conestogac.on.ca

Objective:

- Implementing a solution.
- Implementing your design in Quartus II, using VHDL file.
- Connecting your FPGA.

Truth Table:

Sw0	Sw1	Sw2	Sw3	Sw4	Sw5	LEDR	LEDG
0	0	0	0	0	0	0	1
0	0	0	0	0	1	0	1
0	0	0	0	1	0	0	1
0	0	0	0	1	1	0	1
0	0	0	1	0	0	0	1
0	0	0	1	0	1	0	1
0	0	0	1	1	0	0	1
0	0	0	1	1	1	0	1
0	0	1	0	0	0	0	1
0	0	1	0	0	1	0	1
0	0	1	0	1	0	0	1

0	0	1	0	1	1	0	1
0	0	1	1	0	0	0	1
0	0	1	1	0	1	0	1
0	0	1	1	1	0	0	1
0	0	1	1	1	1	0	1
0	1	0	0	0	0	0	1
0	1	0	0	0	1	0	1
0	1	0	0	1	0	0	1
0	1	0	0	1	1	0	1
0	1	0	1	0	0	0	1
0	1	0	1	0	1	0	1
0	1	0	1	1	0	0	1
0	1	0	1	1	1	0	1
0	1	1	0	0	0	0	1
0	1	1	0	0	1	0	1
0	1	1	0	1	0	0	1
0	1	1	0	1	1	0	1
0	1	1	1	0	0	0	1
0	1	1	1	0	1	0	1
0	1	1	1	1	0	0	1

0	1	1	1	1	1	0	1
1	0	0	0	0	0	0	1
1	0	0	0	0	1	0	1
1	0	0	0	1	0	0	1
1	0	0	0	1	1	0	1
1	0	0	1	0	0	0	1
1	0	0	1	0	1	0	1
1	0	0	1	1	0	0	1
1	0	0	1	1	1	0	1
1	0	1	0	0	0	0	1
1	0	1	0	0	1	0	1
1	0	1	0	1	0	0	1
1	0	1	0	1	1	0	1
1	0	1	1	0	0	0	1
1	0	1	1	0	1	0	1
1	0	1	1	1	0	0	1
1	0	1	1	1	1	0	1
1	1	0	0	0	0	0	1
1	1	0	0	0	1	0	1
1	1	0	0	1	0	0	1

1	1	0	0	1	1	0	1
1	1	0	1	0	0	0	1
1	1	0	1	0	1	0	1
1	1	0	1	1	0	0	1
1	1	0	1	1	1	0	1
1	1	1	0	0	0	0	1
1	1	1	0	0	1	0	1
1	1	1	0	1	0	0	1
1	1	1	0	1	1	0	1
1	1	1	1	0	0	0	1
1	1	1	1	0	1	0	1
1	1	1	1	1	0	0	1
1	1	1	1	1	1	1	0

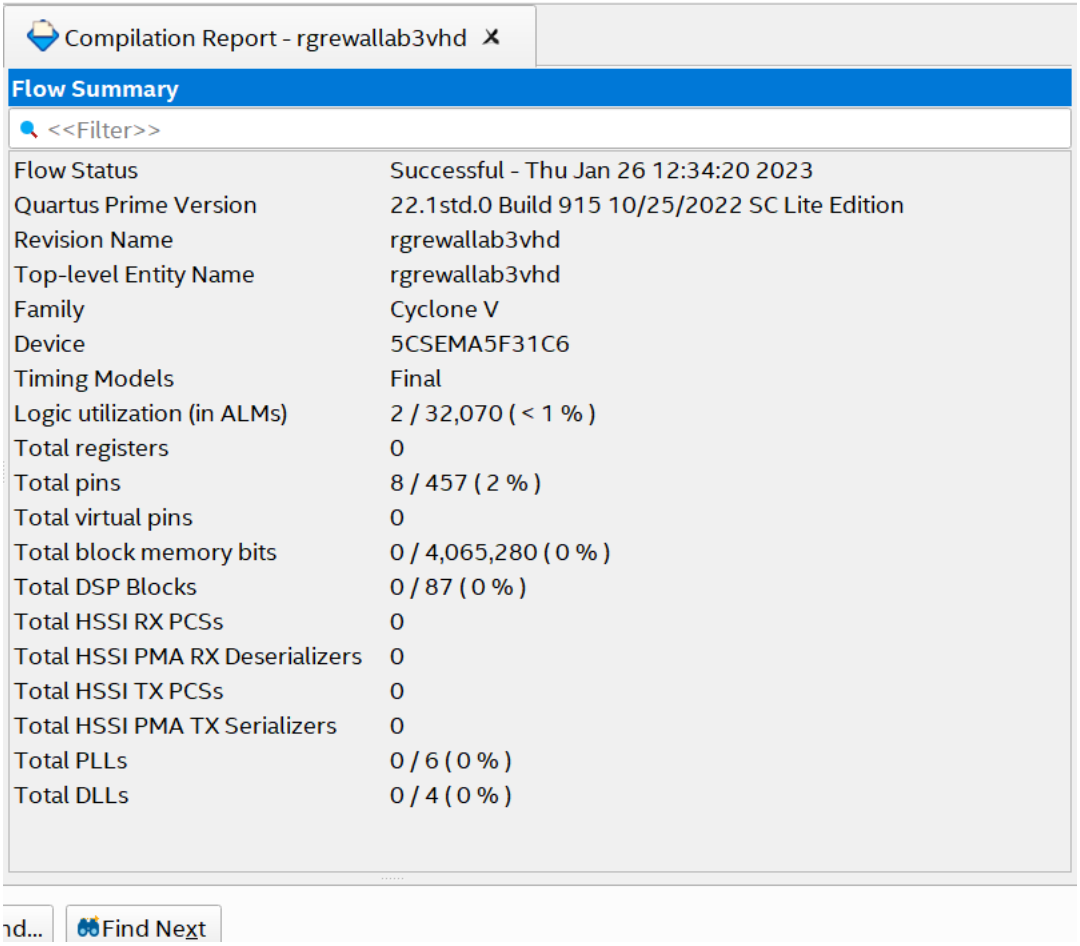
VHDL Introduction:

To make the program in VHDL we will need to initialize the library and entity. Within the entity we will state the 6 input switches and the two output LEDs. Within the architecture of the program we will state the one output will be assign with AND gate. And the other output will be NAND gate, or to not the first output.

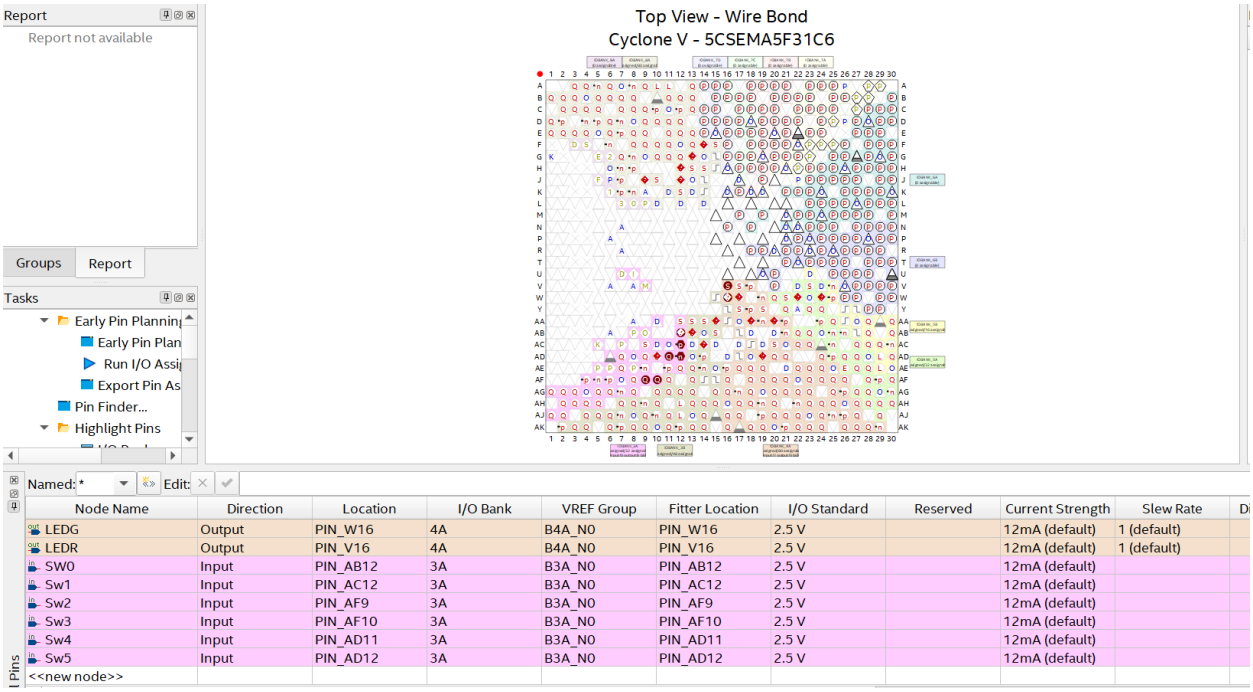
VHDL program screenshot:

```
1  -- library initialize
2  library IEEE;
3  use IEEE.STD_LOGIC_1164.ALL;
4  -- end library initialize
5  -- entity initialize
6  entity rgrewallab3vhd is
7     port(SW0: in std_logic;
8          Sw1: in std_logic;
9          Sw2: in std_logic;
10         Sw3: in std_logic;
11         Sw4: in std_logic;
12         Sw5: in std_logic;
13         LEDR: out std_logic;
14         LEDG: out std_logic);
15 end rgrewallab3vhd;
16 -- end entity initialize
17 -- architecture initialize
18 architecture parkinglot of rgrewallab3vhd is
19    begin
20       -- and gate as LEDR will turn on when all switch is on.
21       LEDR <= Sw0 and Sw1 and Sw2 and Sw3 and Sw4 and Sw5;
22       -- nand gate as LEDG will turn on when there is atleast one switch is off
23       LEDG <= not (Sw0 and Sw1 and Sw2 and Sw3 and Sw4 and Sw5);
24 end parkinglot;
25 -- end architecture initialize
```

VHDL compilation report:



Pin Planner screenshot:



Node Name	Direction	Location	I/O Bank	VREF Group	Fitter Location	I/O Standard	Reserved	Current Strength	Slew Rate	D
LEDG	Output	PIN_W16	4A	B4A_N0	PIN_W16	2.5 V		12mA (default)	1 (default)	
LEDR	Output	PIN_V16	4A	B4A_N0	PIN_V16	2.5 V		12mA (default)	1 (default)	
SW0	Input	PIN_AB12	3A	B3A_N0	PIN_AB12	2.5 V		12mA (default)		
Sw1	Input	PIN_AC12	3A	B3A_N0	PIN_AC12	2.5 V		12mA (default)		
Sw2	Input	PIN_AF9	3A	B3A_N0	PIN_AF9	2.5 V		12mA (default)		
Sw3	Input	PIN_AF10	3A	B3A_N0	PIN_AF10	2.5 V		12mA (default)		
Sw4	Input	PIN_AD11	3A	B3A_N0	PIN_AD11	2.5 V		12mA (default)		
Sw5	Input	PIN_AD12	3A	B3A_N0	PIN_AD12	2.5 V		12mA (default)		
<<new node>>										

Verilog introduction:

For the program in Verilog we will need to initialize the module with the parameters having the six inputs and two outputs. Within the module block we will assign both LEDs where one LED is and of all the inputs. And the other is not of and of all the inputs. Afterwards we end the module.

Verilog program screenshot:

```
1  // initialize module
2  module rgrewallab3verilog(input Sw0,
3                            input Sw1,
4                            input Sw2,
5                            input Sw3,
6                            input Sw4,
7                            input Sw5,
8                            output LEDR,
9                            output LEDG);
10
11 // initalize and gate LEDR which will turn on when all switch is turned on
12 assign LEDR = Sw0 & Sw1 & Sw2 & Sw3 & Sw4 & Sw5;
13 // initialize nand gate LEDG which will turn on when atleast one switch is turned
14 // off
15 assign LEDG = !(Sw0 & Sw1 & Sw2 & Sw3 & Sw4 & Sw5);
16 endmodule
17 // end initilize module
```

Verilog compilation report:

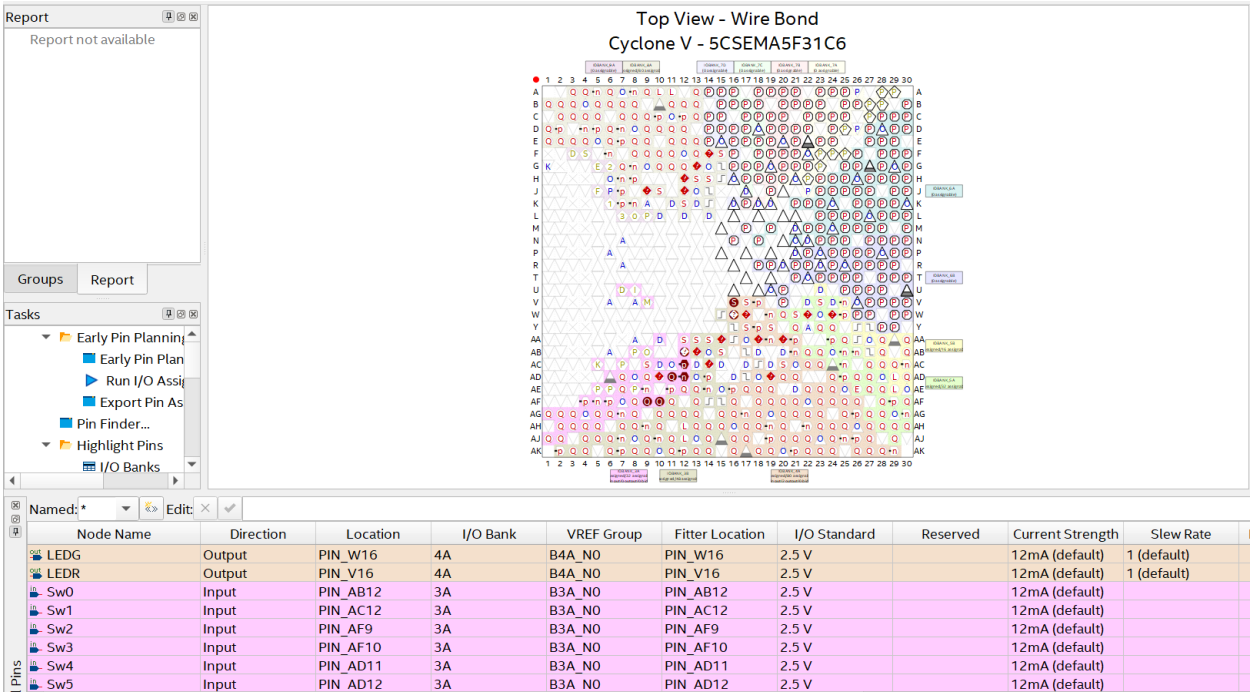
Compilation Report - rgrewallab3verilog

Flow Summary

<<Filter>>

Flow Status	Successful - Thu Jan 26 12:44:38 2023
Quartus Prime Version	22.1std.0 Build 915 10/25/2022 SC Lite Edition
Revision Name	rgrewallab3verilog
Top-level Entity Name	rgrewallab3verilog
Family	Cyclone V
Device	5CSEMA5F31C6
Timing Models	Final
Logic utilization (in ALMs)	2 / 32,070 (< 1 %)
Total registers	0
Total pins	8 / 457 (2 %)
Total virtual pins	0
Total block memory bits	0 / 4,065,280 (0 %)
Total DSP Blocks	0 / 87 (0 %)
Total HSSI RX PCSs	0
Total HSSI PMA RX Deserializers	0
Total HSSI TX PCSs	0
Total HSSI PMA TX Serializers	0
Total PLLs	0 / 6 (0 %)
Total DLLs	0 / 4 (0 %)

Verilog pin planner screenshot:



Node Name	Direction	Location	I/O Bank	VREF Group	Fitter Location	I/O Standard	Reserved	Current Strength	Slew Rate
LEDG	Output	PIN_W16	4A	B4A_N0	PIN_W16	2.5 V		12mA (default)	1 (default)
LEDR	Output	PIN_V16	4A	B4A_N0	PIN_V16	2.5 V		12mA (default)	1 (default)
Sw0	Input	PIN_AB12	3A	B3A_N0	PIN_AB12	2.5 V		12mA (default)	
Sw1	Input	PIN_AC12	3A	B3A_N0	PIN_AC12	2.5 V		12mA (default)	
Sw2	Input	PIN_AF9	3A	B3A_N0	PIN_AF9	2.5 V		12mA (default)	
Sw3	Input	PIN_AF10	3A	B3A_N0	PIN_AF10	2.5 V		12mA (default)	
Sw4	Input	PIN_AD11	3A	B3A_N0	PIN_AD11	2.5 V		12mA (default)	
Sw5	Input	PIN_AD12	3A	B3A_N0	PIN_AD12	2.5 V		12mA (default)	