

Digital Design Principles

Seven Segment Display

Name: Rajkaran Singh Grewal

Student No: 8882386

Email: rgrewal2386@conestogac.on.ca

Objective:

- Use VHDL to implement a binary to 7 Segment decoder.
- Introduction to the VHDL Case statement.

Truth Table:

Sw3	Sw2	Sw1	Sw0	LED
0	0	0	0	0
0	0	0	1	1
0	0	1	0	2
0	0	1	1	3
0	1	0	0	4
0	1	0	1	5
0	1	1	0	6
0	1	1	1	7
1	0	0	0	8
1	0	0	1	9
1	0	1	0	A
1	0	1	1	B

1	1	0	0	C
1	1	0	1	D
1	1	1	0	E
1	1	1	1	F

VHDL Introduction:

To make the program in VHDL we will need to initialize the library and entity. Within the entity we will state the 4 input switches and the seven segment display. Within the architecture of the program we will use case and when statement. Where for each different state the four inputs are the according segments of the seven segments display to light up the symbol to show the hexadecimal version of the input. The above truth table is the be followed to know which led is to be set accordingly.

VHDL program screenshot:

```
VHDL > ≡ rgrewallab7vhd1.vhd
1  library IEEE;
2  use IEEE.STD_LOGIC_1164.ALL;
3
4  entity rgrewallab7vhd1 is
5      port(
6          Sw: in std_logic_vector(3 downto 0);
7          led: out std_logic_vector(6 downto 0)
8      );
9  end rgrewallab7vhd1;
10
11 architecture sevenSegmentDisplay of rgrewallab7vhd1 is
12     begin
13         process(Sw) is
14             begin
15                 case Sw is
16                     when "0000" =>
17                         led <= "0000001";
18
19                     when "0001" =>
20                         led <= "1001111";
21
22                     when "0010" =>
23                         led <= "0010010";
24
25                     when "0011" =>
26                         led <= "0000110";
27
28                     when "0100" =>
29                         led <= "1001100";
30
```

```
31      when "0101" =>
32          led <= "0100100";
33
34      when "0110" =>
35          led <= "0100000";
36
37      when "0111" =>
38          led <= "0001111";
39
40      when "1000" =>
41          led <= "0000000";
42
43      when "1001" =>
44          led <= "0001100";
45
46      when "1010" =>
47          led <= "0001000";
48
49      when "1011" =>
50          led <= "1100000";
51
52      when "1100" =>
53          led <= "0110001";
54
55      when "1101" =>
56          led <= "1000010";
57
58      when "1110" =>
59          led <= "0110000";
60
```

```
61         when "1111" =>
62             led <= "0111000";
63
64         when others =>
65             led <= "1111111";
66     end case;
67 end process;
68 end sevenSegmentDisplay;
```

VHDL compilation report:

Compilation Report - rgrewallab7vhdl

Flow Summary

<<Filter>>

Flow Status	Successful - Wed Mar 15 19:06:52 2023
Quartus Prime Version	22.1std.0 Build 915 10/25/2022 SC Lite Edition
Revision Name	rgrewallab7vhdl
Top-level Entity Name	rgrewallab7vhdl
Family	Cyclone V
Device	5CSEMA5F31C6
Timing Models	Final
Logic utilization (in ALMs)	4 / 32,070 (< 1 %)
Total registers	0
Total pins	11 / 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 %)

Pin Planner screenshot:

Pin Planner - C:\Embedded Systems\Semester 1\Digital Design Principles\Labs\Lab 7\VHDL\rgrewallab7vhdl - rgrewallab7vhdl

File Edit View Processing Tools Window Help

Search Intel FPGA

Report

Report not available

Groups Report

Tasks

- Early Pin Planning
 - Early Pin Plan
 - Run I/O Assi
 - Export Pin As
 - Pin Finder...

Top View - Wire Bond
Cyclone V - 5CSEMA5F31C6

Pin Legend

Symbol	Pin Type
●	User I/O
○	User assigned I/O
●	Fitter assigned I...
○	Unbonded pad
○	Reserved pin
○	DEV_OE
○	DIFF_n
○	DIFF_p
○	DIFF_n output
○	DIFF_p output
○	DQ
○	DQS
○	DQSB
○	Hard processor ...
○	CLK_n

Node Name	Direction	Location	I/O Bank	VREF Group	Fitter Location	I/O Standard	Reserved	Current Strength	Slew Rate	Differential Pair	er Analog Se
led[6]	Output	PIN_V25	5B	B5B_NO	PIN_V25	2.5 V		12mA (default)	1 (default)		
led[5]	Output	PIN_AA28	5B	B5B_NO	PIN_AA28	2.5 V		12mA (default)	1 (default)		
led[4]	Output	PIN_Y27	5B	B5B_NO	PIN_Y27	2.5 V		12mA (default)	1 (default)		
led[3]	Output	PIN_AB27	5B	B5B_NO	PIN_AB27	2.5 V		12mA (default)	1 (default)		
led[2]	Output	PIN_AB26	5A	B5A_NO	PIN_AB26	2.5 V		12mA (default)	1 (default)		
led[1]	Output	PIN_AA26	5B	B5B_NO	PIN_AA26	2.5 V		12mA (default)	1 (default)		
led[0]	Output	PIN_AA25	5A	B5A_NO	PIN_AA25	2.5 V		12mA (default)	1 (default)		
Sw[3]	Input	PIN_AF10	3A	B3A_NO	PIN_AF10	2.5 V		12mA (default)			
Sw[2]	Input	PIN_AF9	3A	B3A_NO	PIN_AF9	2.5 V		12mA (default)			
Sw[1]	Input	PIN_AC12	3A	B3A_NO	PIN_AC12	2.5 V		12mA (default)			
Sw[0]	Input	PIN_AB12	3A	B3A_NO	PIN_AB12	2.5 V		12mA (default)			

Filter: Pins: all

0% 00:00:00

Type here to search

4°C 12:07 PM 3/16/2023

Verilog introduction:

For the program in Verilog we will need to initialize the module with the parameters having the four inputs and a seven segment display. Within the module block we use case where the output will assign the value according to the state of the input which matches to the truth table.

Verilog program screenshot:

```
Verilog > rgrewallab7verilog.v
1  module rgrewallab7verilog(input [3:0] Sw,output reg [6:0] led);
2  always @(Sw) begin
3      case(Sw)
4          4'b0000: begin
5              led = 7'b0000001;
6          end
7          4'b0001: begin
8              led = 7'b1001111;
9          end
10         4'b0010: begin
11             led = 7'b0010010;
12         end
13         4'b0011: begin
14             led = 7'b0000110;
15         end
16         4'b0100: begin
17             led = 7'b1001100;
18         end
19         4'b0101: begin
20             led = 7'b0100100;
21         end
22         4'b0110: begin
23             led = 7'b0100000;
24         end
25         4'b0111: begin
26             led = 7'b0001111;
27         end
28         4'b1000: begin
29             led = 7'b0000000;
30         end
30     end
```



```
31      4'b1001: begin
32          |      led = 7'b0001100;
33      end
34      4'b1010: begin
35          |      led = 7'b0001000;
36      end
37      4'b1011: begin
38          |      led = 7'b1100000;
39      end
40      4'b1100: begin
41          |      led = 7'b0110001;
42      end
43      4'b1101: begin
44          |      led = 7'b1000010;
45      end
46      4'b1110: begin
47          |      led = 7'b0110000;
48      end
49      4'b1111: begin
50          |      led = 7'b0111000;
51      end
52      default: begin
53          |      led = 7'b1111111;
54      end
55  endcase
56 end
57 endmodule
```

Verilog compilation report:

The screenshot shows the 'Compilation Report - rgrewallab7verilog' window. The 'Flow Summary' tab is active, displaying the following information:

Flow Status	Successful - Wed Mar 15 18:39:48 2023
Quartus Prime Version	22.1std.0 Build 915 10/25/2022 SC Lite Edition
Revision Name	rgrewallab7verilog
Top-level Entity Name	rgrewallab7verilog
Family	Cyclone V
Device	5CSEMA5F31C6
Timing Models	Final
Logic utilization (in ALMs)	4 / 32,070 (< 1 %)
Total registers	0
Total pins	11 / 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:

The screenshot shows the 'Pin Planner' window for the Cyclone V - 5CSEMA5F31C6 device. The 'Top View - Wire Bond' is displayed, showing the physical layout of the pins. The 'Pin Legend' on the right lists various pin types and their symbols. The 'Tasks' pane on the left shows the 'Early Pin Planning' tasks. The 'Report' pane on the left shows 'Report not available'. The 'Pin Planner' table at the bottom lists the pins and their configurations.

Node Name	Direction	Location	I/O Bank	VREF Group	Fitter Location	I/O Standard	Reserved	Current Strength	Slew Rate	Differential Pair	er Analog Se
Sw[3]	Input	PIN_AF10	3A	B3A_NO	PIN_AF10	2.5 V		12mA (default)			
Sw[2]	Input	PIN_AF9	3A	B3A_NO	PIN_AF9	2.5 V		12mA (default)			
Sw[1]	Input	PIN_AC12	3A	B3A_NO	PIN_AC12	2.5 V		12mA (default)			
Sw[0]	Input	PIN_AB12	3A	B3A_NO	PIN_AB12	2.5 V		12mA (default)			
led[6]	Output	PIN_V25	5B	B5B_NO	PIN_V25	2.5 V		12mA (default)	1 (default)		
led[5]	Output	PIN_AA28	5B	B5B_NO	PIN_AA28	2.5 V		12mA (default)	1 (default)		
led[4]	Output	PIN_Y27	5B	B5B_NO	PIN_Y27	2.5 V		12mA (default)	1 (default)		
led[3]	Output	PIN_AB27	5B	B5B_NO	PIN_AB27	2.5 V		12mA (default)	1 (default)		
led[2]	Output	PIN_AB26	5A	B5A_NO	PIN_AB26	2.5 V		12mA (default)	1 (default)		
led[1]	Output	PIN_AA26	5B	B5B_NO	PIN_AA26	2.5 V		12mA (default)	1 (default)		
led[0]	Output	PIN_AA25	5A	B5A_NO	PIN_AA25	2.5 V		12mA (default)	1 (default)		

Conclusion:

Thus we can conclude that by both the programming languages we will use case statement and the one seven segment display which corresponds to the input will light up. The above truth table will be followed to show a binary to hexacode convertor.