COL215 HW Assignment 2 - 4-Digit 7-Segment Display

Submission By :

Aaditya Sharma **Namit Jain**

2023CS10420 **2023CS10483**

8th September, 2024

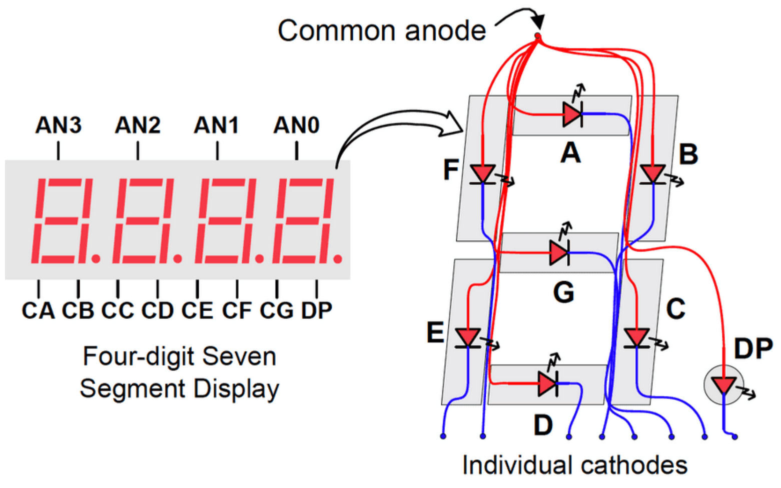


Figure 18. Common anode circuit node.

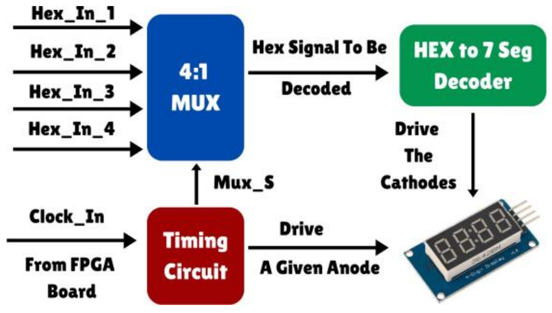
Problem Statement

Design a combinational circuit that converts a 4-bit hexadecimal or decimal input to a 7-bit output suitable for driving a seven-segment display on a Basys 3 FPGA board.

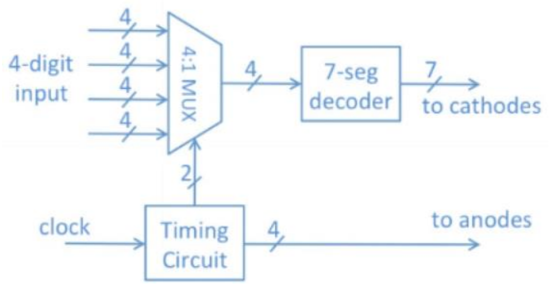
Expand the design to accommodate four 4-bit inputs, allowing for the simultaneous display of a 4-digit number on four seven-segment displays.

Overview of Assignment :

The assignment requires us to design three modules with the following inheritance logic :



Overview of Design of Assignment



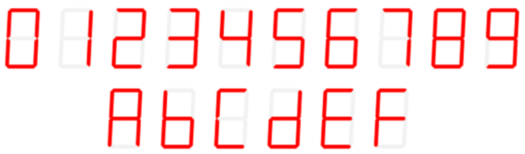
Circuit Diagram of Assignment

A wrapper VHDL file, named 7SegDisplay, will encapsulate the entire design, managing the input signals and outputting the necessary control signals for the seven-segment displays.
A 4-to-1 multiplexer will receive the four hexadecimal input signals and select the appropriate one based on the timing circuit's control signals.
A timing circuit will determine the anode to be activated and supply the corresponding selection bits to the multiplexer.
A decoder module will convert the 7-bit output from the multiplexer into a format suitable for driving the cathodes of the seven-segment displays using minimized combinational logic.

Design Decisions of Overall Project

Hexadecimal Digits

As we are implementing display of 4 hexadecimal digits, we require 16 input bits from the board to represent the inputs to the MUX. We use the following symbols for representation on 7 Segment display :

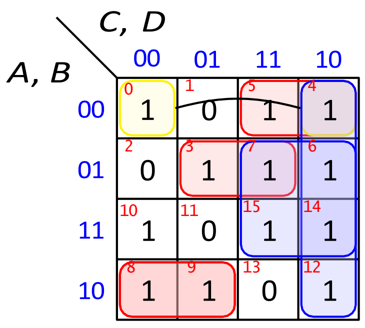


Truth Table - For BCD to 7 Segment Display

Digits & Input Bits					7 Segment Display							Min-Terms: $f(A, B, C, D)$
Digit	A	B	C	D	a	b	c	d	e	f	g	Min-Term Index
0	0	0	0	0	1	1	1	1	1	1	0	m_0
1	0	0	0	1	0	1	1	0	0	0	0	m_1
2	0	0	1	0	1	1	0	1	1	0	1	m_2
3	0	0	1	1	1	1	1	1	0	0	1	m_3
4	0	1	0	0	0	1	1	0	0	1	1	m_4
5	0	1	0	1	1	0	1	1	0	1	1	m_5
6	0	1	1	0	1	0	1	1	1	1	1	m_6
7	0	1	1	1	1	1	1	0	0	0	0	m_7
8	1	0	0	0	1	1	1	1	1	1	1	m_8
9	1	0	0	1	1	1	1	0	0	1	1	m_9
A	1	0	1	0	1	1	1	0	1	1	1	m_{10}
b	1	0	1	1	0	0	1	1	1	1	1	m_{11}
C	1	1	0	0	1	0	0	1	1	1	0	m_{12}
d	1	1	0	1	0	1	1	1	1	0	1	m_{13}
E	1	1	1	0	1	0	0	1	1	1	1	m_{14}
F	1	1	1	1	1	0	0	0	1	1	1	m_{15}

Minimizing the Combinational Logic using K-Maps

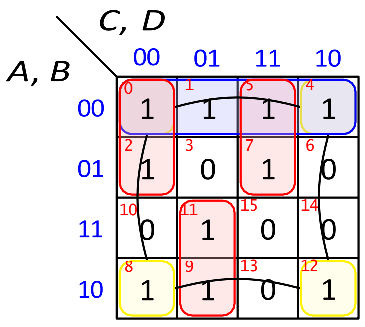
Segment a



The reduced formula for segment a using the mentioned K-Map reduction is as following :

f(A,B,C,D) = A’B’D’ + A’BD + AC’D’ + AB’C’ + A’C + BC + CD’

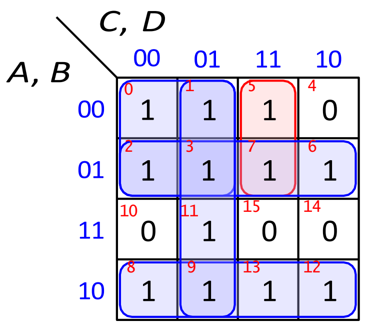
Segment b



The reduced formula for segment b using the mentioned K-Map reduction is as following :

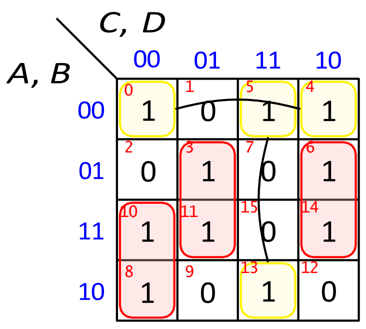
f(A,B,C,D) = A’C’D’ + A’CD + AC’D + B’D’ + A’B’

Segment c



The reduced formula for segment c using the mentioned K-Map reduction is as following :
f(A,B,C,D) = A’CD + A’C’ + A’B + C’D + AB’

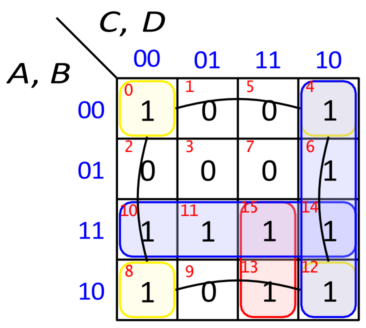
Segment d



The reduced formula for segment d using the mentioned K-Map reduction is as following :

f(A,B,C,D) = A’B’D’ + B’CD + BC’D + BCD’ + A’C’D’

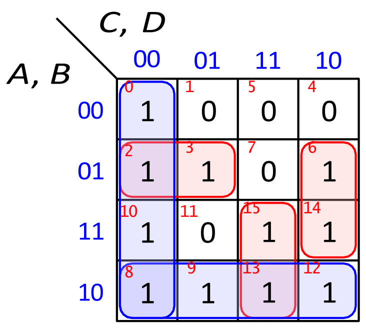
Segment e



The reduced formula for segment e using the mentioned K-Map reduction is as following :

f(A,B,C,D) = B’C’D’ + AB + CD’ + AC

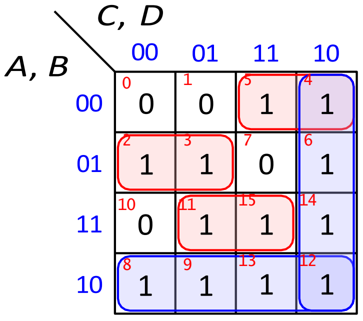
Segment f



The reduced formula for segment f using the mentioned K-Map reduction is as following :

f(A,B,C,D) = A’BC’ + BCD’ + AB’ + AC + C’D’

Segment g



A, B \ C, D	00	01	11	10
00	0	0	1	1
01	1	1	0	1
11	0	1	1	1
10	1	1	1	1

The reduced formula for segment g using the mentioned K-Map reduction is as following :

f(A,B,C,D) = A’B’C + A’BC’ + CD’ + AB’ + AD

Design Decisions for Modules

Hexadecimal-To-7-Segment-Display

Decoder is a combinational module which accepts the 4 input bits from the **mux4x1_4bit** to be displayed on a seven segment display. Then the decoder implements the reduced combinational logic and sets the respective states of cathodes to drive the output on the seven segment display.

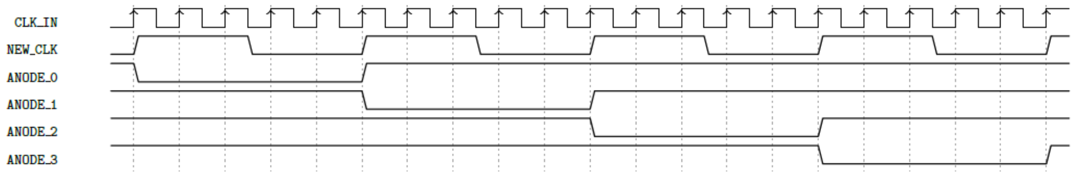
4:1 MUX

The board is used to generate 4 hexadecimal inputs, one for each seven segment display. Since the cathodes of all the 4 displays are common, the timer circuit provides select bits : *dec_in* which also corresponds to the anode, i.e. one of the four 7 Segment displays which are being driven. The implemented MUX Process chooses the hexadecimal input to be displayed using case statement, which is passed onto the decoder module to be decoded into 7 segment bits.

Timing Circuit

Timing circuit is the crux of this project. Since the cathodes of all the four 7 Segment displays are common, we need a way to cycle through the different anodes and a way to keep track which hexadecimal input is to displayed on the given anode (which is what the 4:1 MUX module implements).

This module also accepts **clk_in** from the Basys3 board, an inbuilt 100 MHz clock. For our practical purposes using such a high frequency clock is not viable since it may cause flickering issues, care is taken to implement a **new_clk** which uses a **counter** and **N** to increase the time period from 10 *ns* to 1ms, a frequency 1000 *Hz* as given below.



Clock and Anode Selection Logic to be implemented by Timing Block

Since human eye vision is persistent at these refresh rates, the **MUX_PROC** process changes the state of **mux_select_counter** cyclically according to the

rising edge of **new clk** and assigns it to **mux select** which goes to MUX module. This part handles the cyclical change of anodes which allow us to display all 4 digits perceived at once.

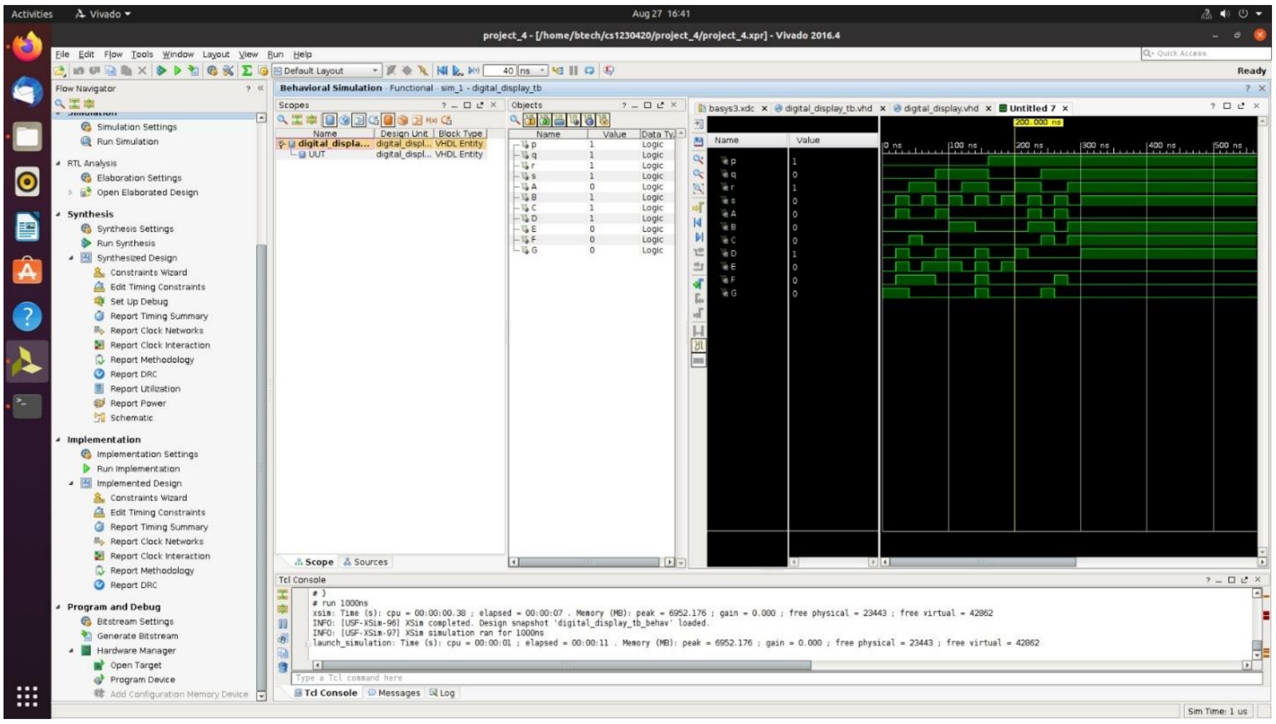
Finally the **ANODE select** process chooses the anode states for choosing which anode the digit is to be displayed upon.

Note that whenever the board detects reset, the **counter** and the **new clk** is restored to a known default. Also all the displays/anodes are switched off to show the circuit is in a reset state. (We used N = 50000). When there are 50000 rising edges of click_in, new_clk will invert its value, which means it will take 100k rising edges for one complete cycle of new_clk.

7Seg Display - The Final Wrapper

This instantiates all the other modules as entities and implements the necessary signals for input from the board to the modules and the output back to board.

Simulation Snapshots



project_4 - [/home/btech/cs1230420/project_4/project_4.xpr] - Vivado 2016.4

Objects

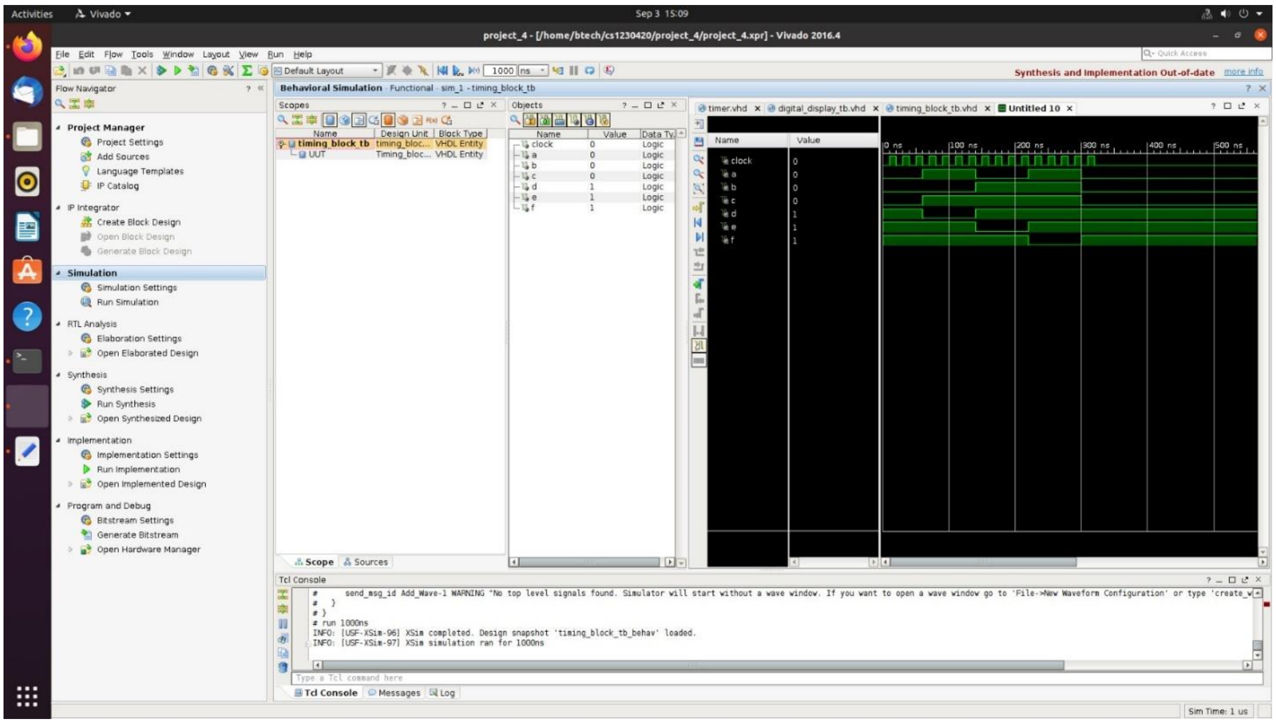
Name	Value	Data Ty.
p	1	Logic
q	1	Logic
r	1	Logic
s	1	Logic
A	0	Logic
B	1	Logic
C	1	Logic
D	1	Logic
E	0	Logic
F	0	Logic
G	0	Logic

Name	Value
p	1
q	0
r	1
s	0
A	0
B	0
C	0
D	1
E	0
F	0
G	0

Tcl Console

```
# }
# run 1000ns
xsim: Time (s): cpu = 00:00:00.38 ; elapsed = 00:00:07 . Memory (MB): peak = 6952.176 ; gain = 0.000 ; free physical = 23443 ; free virtual = 42862
INFO: [USF-XSim-96] XSim completed. Design snapshot 'digital_display_tb_behav' loaded.
INFO: [USF-XSim-97] XSim simulation ran for 1000ns
launch_simulation: Time (s): cpu = 00:00:01 ; elapsed = 00:00:11 . Memory (MB): peak = 6952.176 ; gain = 0.000 ; free physical = 23443 ; free virtual = 42862
```

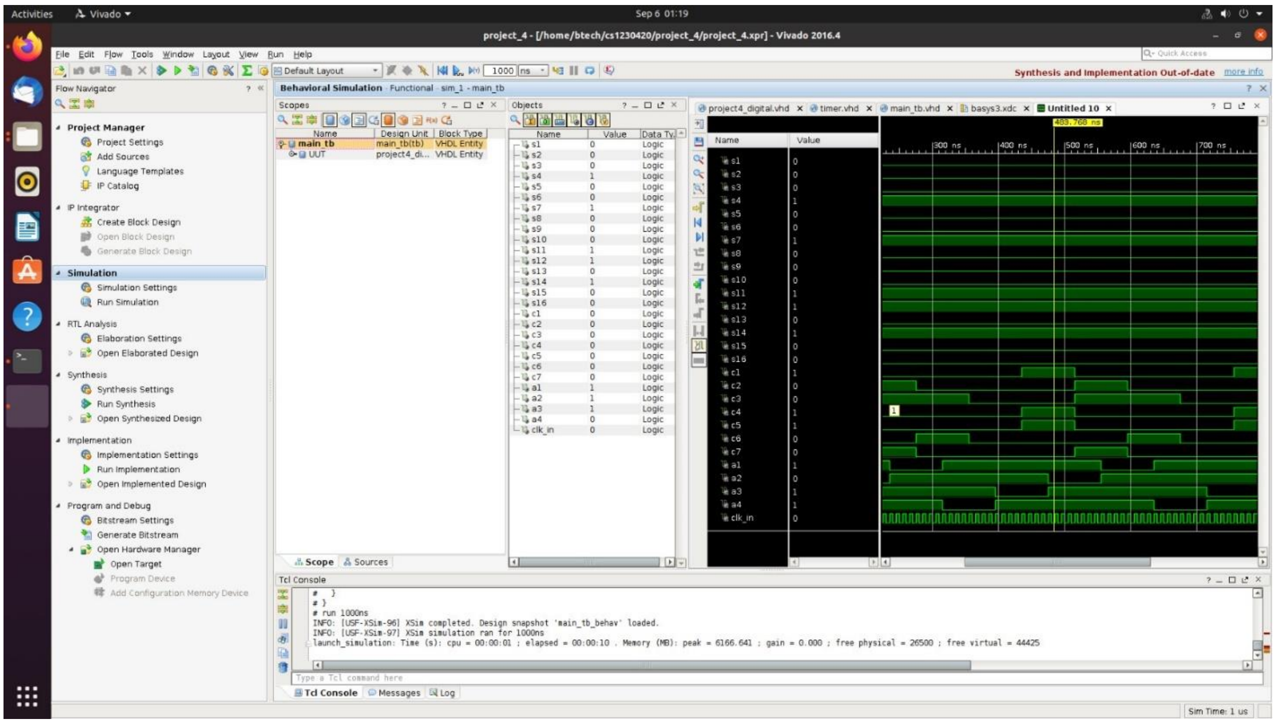
7 Segment Display Simulation



```
project_4 - [/home/btech/cs1230420/project_4/project_4.xpr] - Vivado 2016.4
Behavioral Simulation - Functional - sim_1 - timing_block_tb
```

```
#      send_msg_id Add_Wave-1 WARNING "No top level signals found. Simulator will start without a wave window. If you want to open a wave window go to 'File->New Waveform Configuration' or type 'create_w
#   }
# }
# run 1000ns
INFO: [USF-XSim-96] XSim completed. Design snapshot 'timing_block_tb_behav' loaded.
INFO: [USF-XSim-97] XSim simulation ran for 1000ns
```

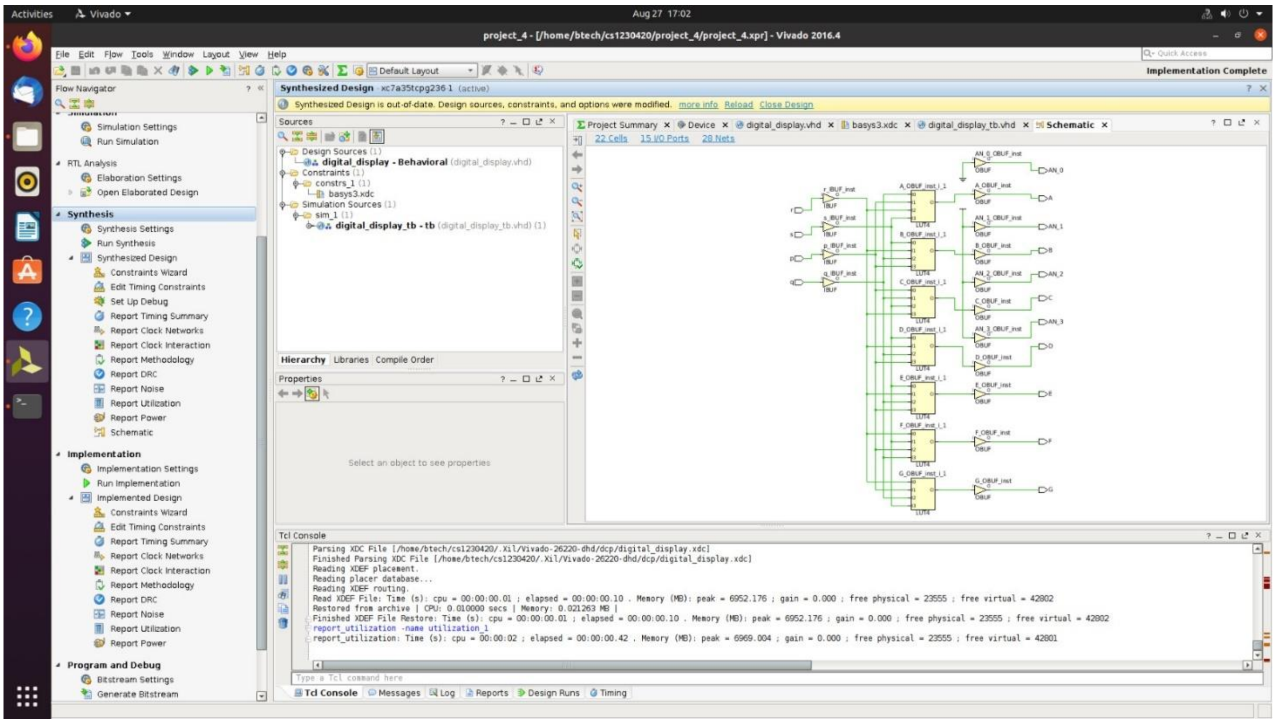
Timing Block Simulation



```
project_4 - [/home/btech/cs1230420/project_4/project_4.xpr] - Vivado 2016.4
Behavioral Simulation - Functional - sim_1 - main_tb
```

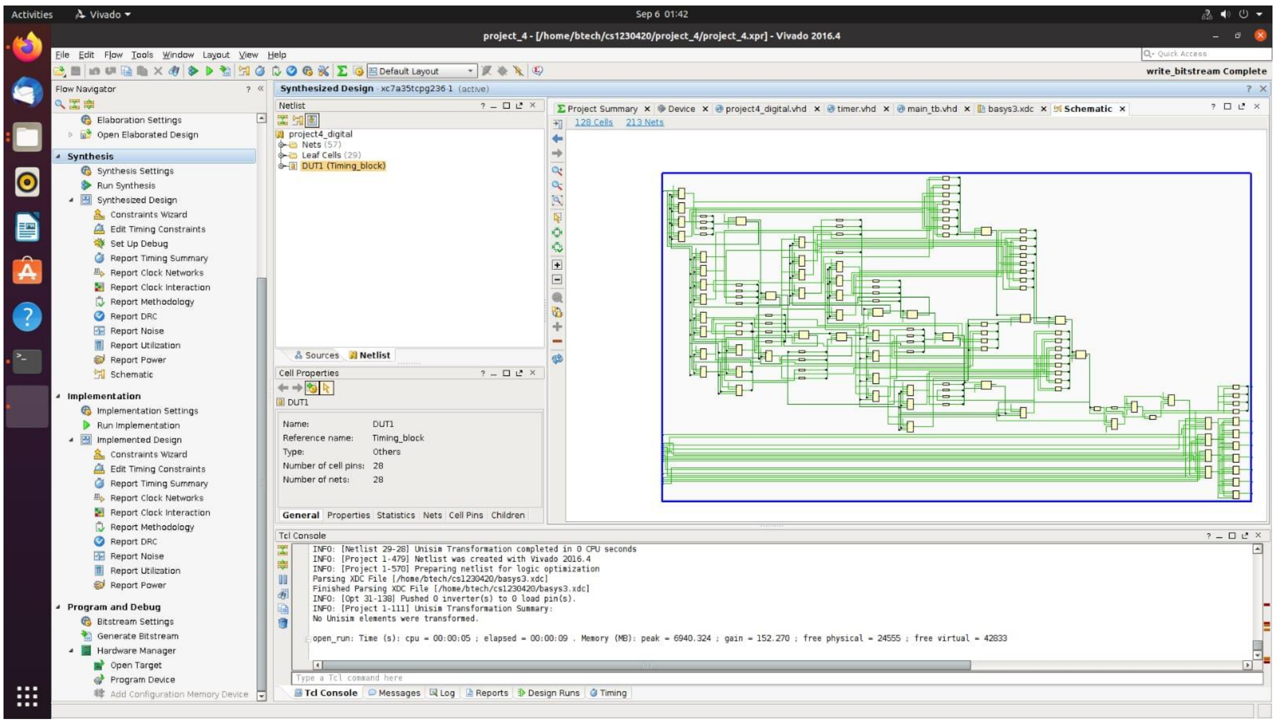
```
#   }
# }
# run 1000ns
INFO: [USF-XSim-96] XSim completed. Design snapshot 'main_tb_behav' loaded.
INFO: [USF-XSim-97] XSim simulation ran for 1000ns
launch_simulation: Time (s): cpu = 00:00:01 ; elapsed = 00:00:10 . Memory (MB): peak = 6166.641 ; gain = 0.000 ; free physical = 26500 ; free virtual = 44425
```

Main Code Simulation



```
Parsing XDC File [/home/btech/cs1230420/.Xil/Vivado-26220-dhd/dcp/digital_display.xdc]
Finished Parsing XDC File [/home/btech/cs1230420/.Xil/Vivado-26220-dhd/dcp/digital_display.xdc]
Reading XDEF placement.
Reading placer database...
Reading XDEF routing.
Read XDEF File: Time (s): cpu = 00:00:00.01 ; elapsed = 00:00:00.10 . Memory (MB): peak = 6952.176 ; gain = 0.000 ; free physical = 23555 ; free virtual = 42802
Restored from archive | CPU: 0.010000 secs | Memory: 0.021263 MB |
Finished XDEF File Restore: Time (s): cpu = 00:00:00.01 ; elapsed = 00:00:00.10 . Memory (MB): peak = 6952.176 ; gain = 0.000 ; free physical = 23555 ; free virtual = 42802
report_utilization -name utilization_1
report_utilization: Time (s): cpu = 00:00:02 ; elapsed = 00:00:00.42 . Memory (MB): peak = 6969.004 ; gain = 0.000 ; free physical = 23555 ; free virtual = 42801
```

Seven Segment Display Schematic

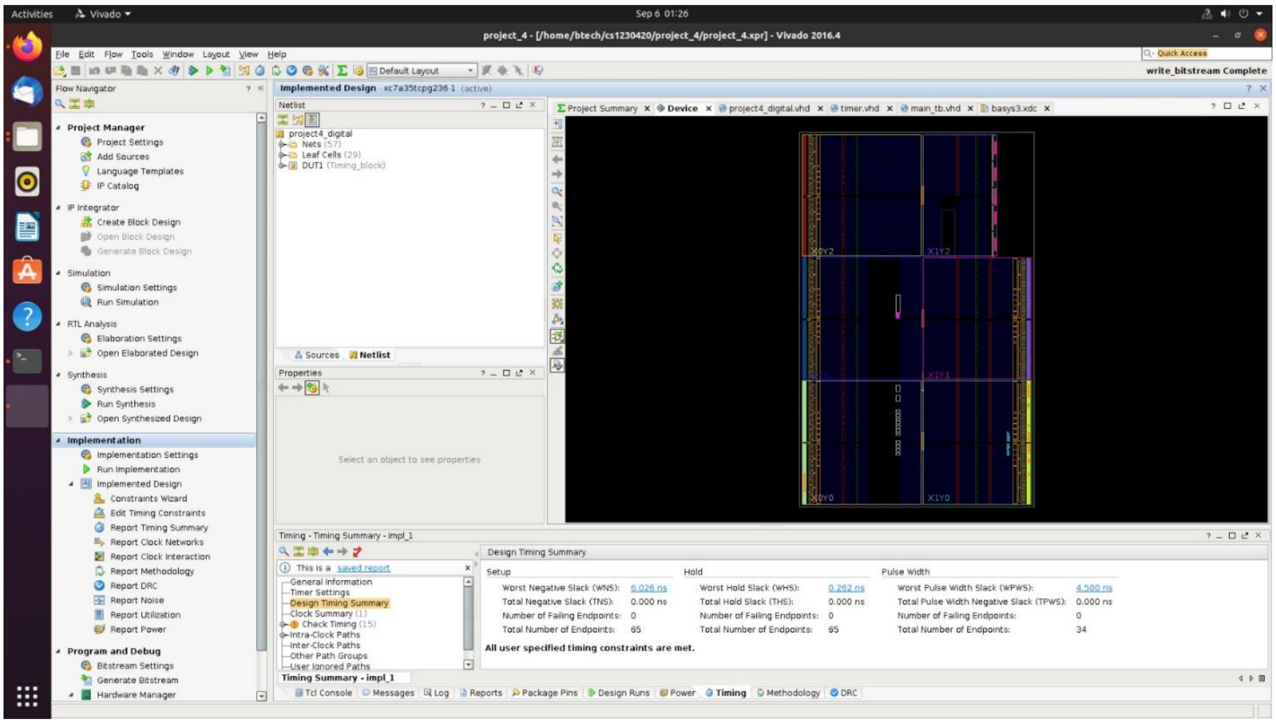


Name: DUT1
Reference name: Timing_block
Type: Others
Number of cell pins: 28
Number of nets: 28

```
INFO: [Netlist 29-28] Unisim Transformation completed in 0 CPU seconds
INFO: [Project 1-479] Netlist was created with Vivado 2016.4
INFO: [Project 1-570] Preparing netlist for logic optimization
Parsing XDC File [/home/btech/cs1230420/basys3.xdc]
Finished Parsing XDC File [/home/btech/cs1230420/basys3.xdc]
INFO: [Opt 31-138] Pushed 0 inverter(s) to 0 load pin(s).
INFO: [Project 1-111] Unisim Transformation Summary:
No Unisim elements were transformed.

open_run: Time (s): cpu = 00:00:05 ; elapsed = 00:00:09 . Memory (MB): peak = 6940.324 ; gain = 152.270 ; free physical = 24555 ; free virtual = 42833
```

Main Code Schematic


project_4 - [/home/btech/cs1230420/project_4/project_4.xpr] - Vivado 2016.4

write_bitstream Complete

Implemented Design - xc7a35tcpg236-1 (active)

Netlist: project4_digital — Nets (57), Leaf Cells (29), DUT1 (Timing_block)

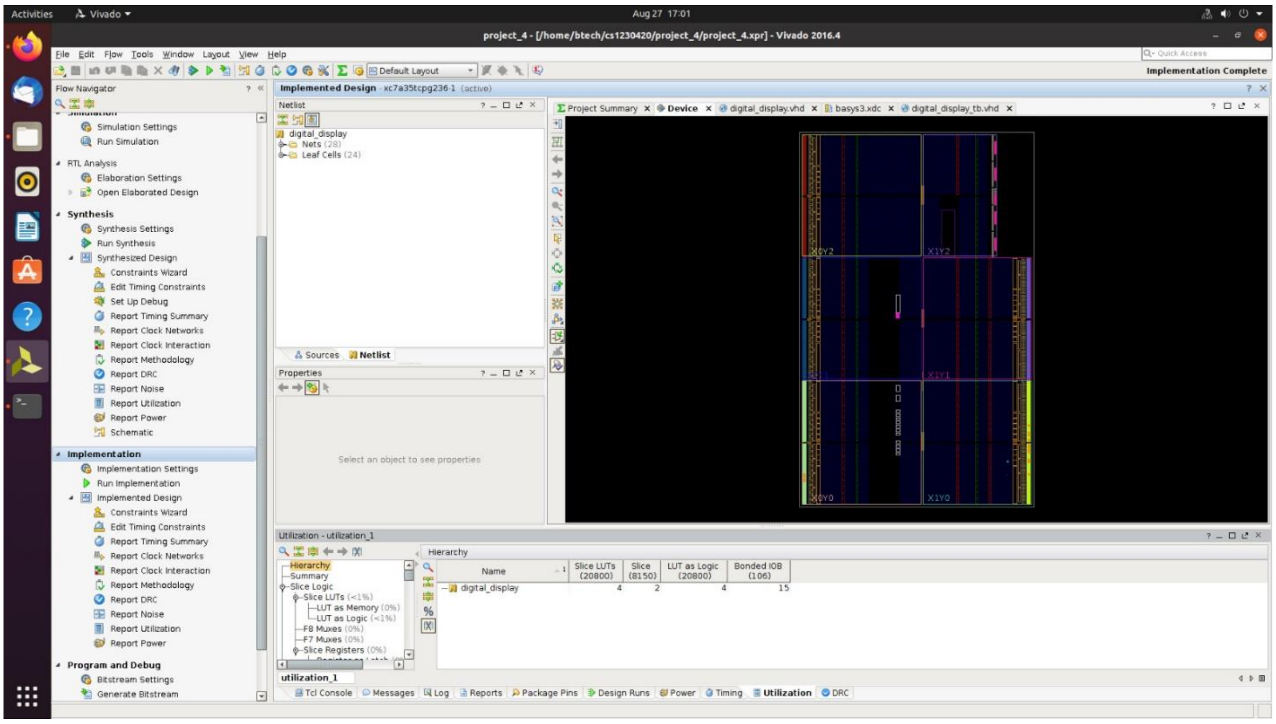
Timing - Timing Summary - impl_1

Design Timing Summary

Setup		Hold		Pulse Width	
Worst Negative Slack (WNS):	6.026 ns	Worst Hold Slack (WHS):	0.262 ns	Worst Pulse Width Slack (WPWS):	4.500 ns
Total Negative Slack (TNS):	0.000 ns	Total Hold Slack (THS):	0.000 ns	Total Pulse Width Negative Slack (TPWS):	0.000 ns
Number of Failing Endpoints:	0	Number of Failing Endpoints:	0	Number of Failing Endpoints:	0
Total Number of Endpoints:	65	Total Number of Endpoints:	65	Total Number of Endpoints:	34

All user specified timing constraints are met.

Main Code Implementation


project_4 - [/home/btech/cs1230420/project_4/project_4.xpr] - Vivado 2016.4

Implementation Complete

Implemented Design - xc7a35tcpg236-1 (active)

Netlist: digital_display — Nets (28), Leaf Cells (24)

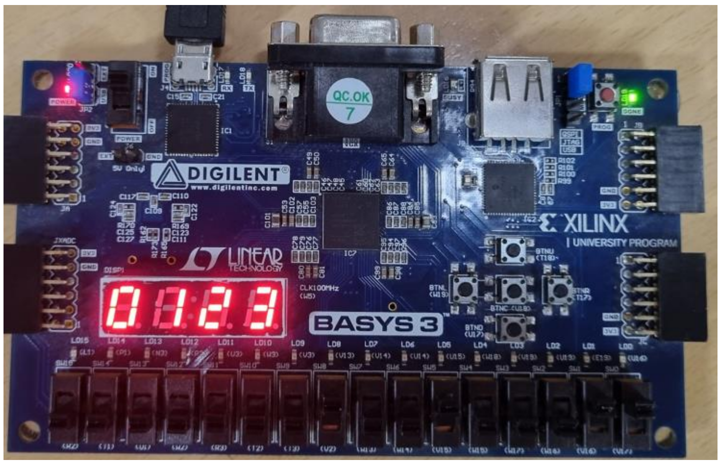
Utilization - utilization_1

Hierarchy

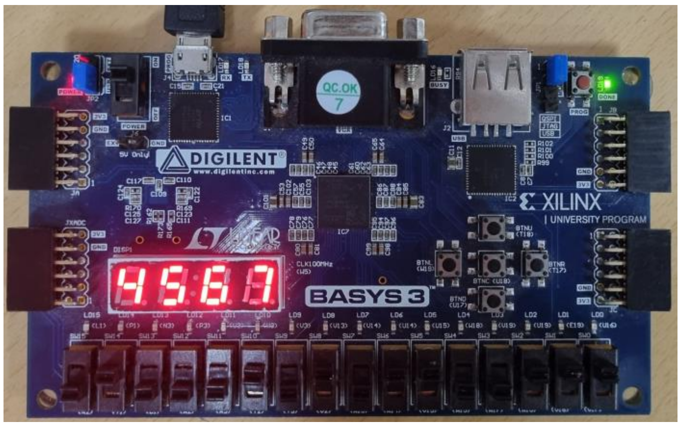
Name	Slice LUTs (20800)	Slice (8150)	LUT as Logic (20800)	Bonded IOB (106)
digital_display	4	2	4	15

Slice Logic: Slice LUTs (<1%), LUT as Memory (0%), LUT as Logic (<1%), F8 Muxes (0%), F7 Muxes (0%), Slice Registers (0%)

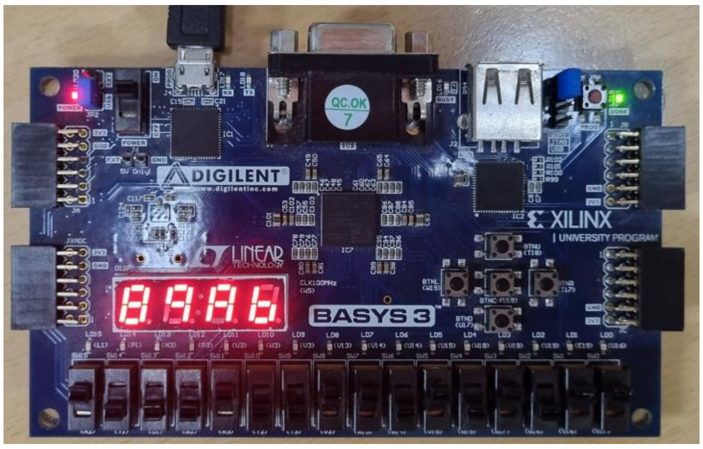
Basys 3 Board Output:



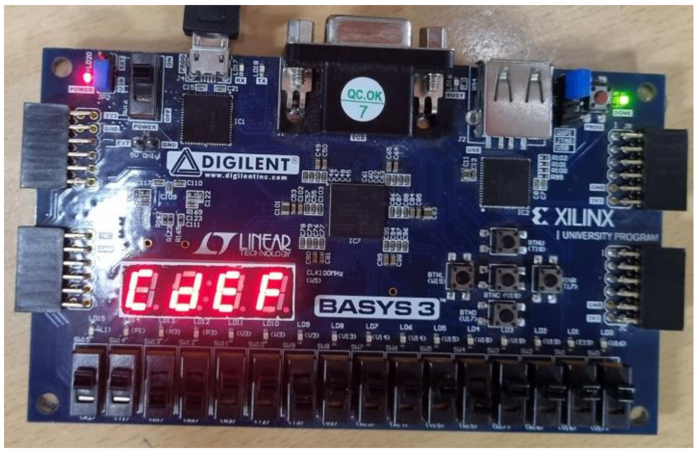
Digits 0-3



Digits 4 – 7

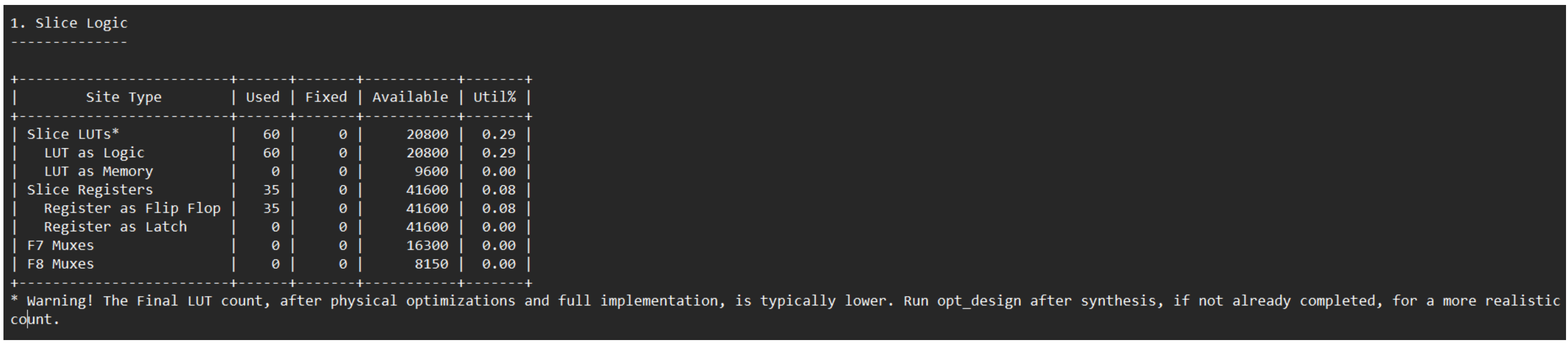


Digits $8 - b$



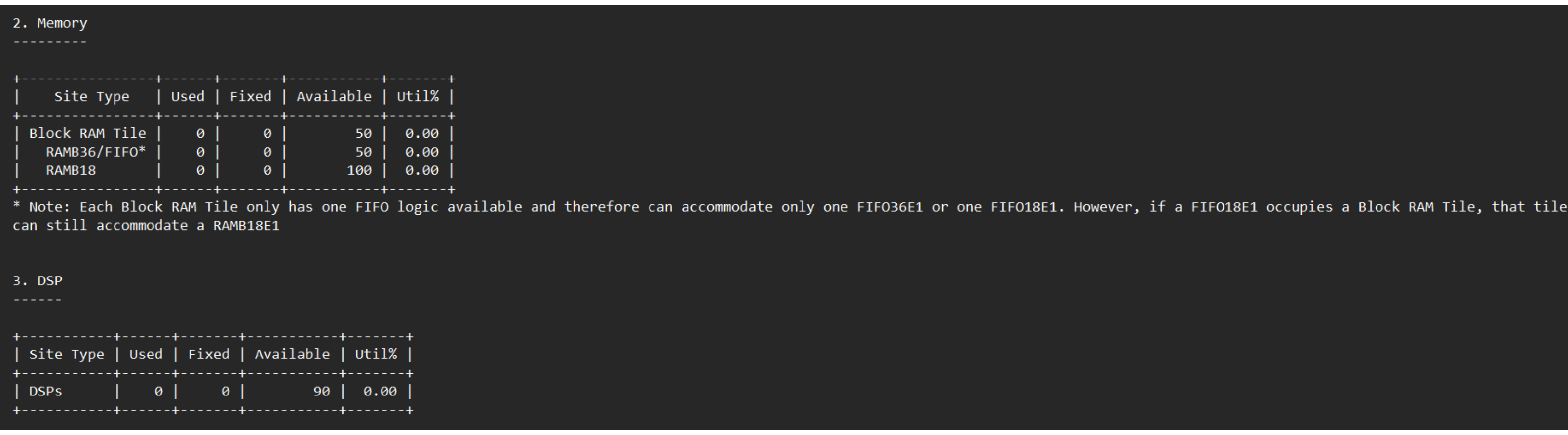
Digits $C - F$

Resource Utilisation:-



```
1. Slice Logic
--------------

+-------------------------+------+-------+-----------+-------+
|        Site Type        | Used | Fixed | Available | Util% |
+-------------------------+------+-------+-----------+-------+
| Slice LUTs*             |   60 |     0 |     20800 |  0.29 |
|   LUT as Logic          |   60 |     0 |     20800 |  0.29 |
|   LUT as Memory         |    0 |     0 |      9600 |  0.00 |
| Slice Registers         |   35 |     0 |     41600 |  0.08 |
|   Register as Flip Flop |   35 |     0 |     41600 |  0.08 |
|   Register as Latch     |    0 |     0 |     41600 |  0.00 |
| F7 Muxes                |    0 |     0 |     16300 |  0.00 |
| F8 Muxes                |    0 |     0 |      8150 |  0.00 |
+-------------------------+------+-------+-----------+-------+
* Warning! The Final LUT count, after physical optimizations and full implementation, is typically lower. Run opt_design after synthesis, if not already completed, for a more realistic count.
```



```
2. Memory
---------

+----------------+------+-------+-----------+-------+
|    Site Type   | Used | Fixed | Available | Util% |
+----------------+------+-------+-----------+-------+
| Block RAM Tile |    0 |     0 |        50 |  0.00 |
|   RAMB36/FIFO* |    0 |     0 |        50 |  0.00 |
|   RAMB18       |    0 |     0 |       100 |  0.00 |
+----------------+------+-------+-----------+-------+
* Note: Each Block RAM Tile only has one FIFO logic available and therefore can accommodate only one FIFO36E1 or one FIFO18E1. However, if a FIFO18E1 occupies a Block RAM Tile, that tile
can still accommodate a RAMB18E1


3. DSP
------

+-----------+------+-------+-----------+-------+
| Site Type | Used | Fixed | Available | Util% |
+-----------+------+-------+-----------+-------+
| DSPs      |    0 |     0 |        90 |  0.00 |
+-----------+------+-------+-----------+-------+
```