Cole Maxwell (25017014): ECE 658

Lab 3: 4-Bit Accumulator

November 2020

Part A: Custom Accumulator Design using Bitsliced Layout

**Step A1: Schematic**

**A traffic light

Description automatically generated**

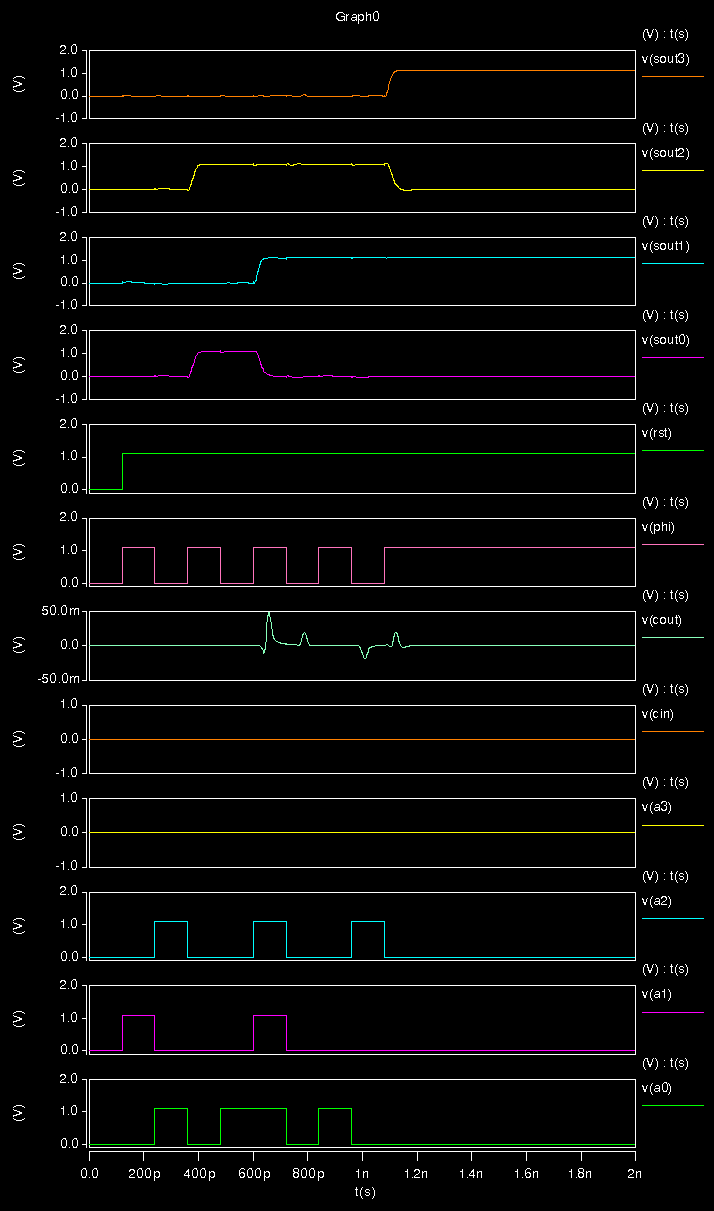
**Step A2: Validate schematic using HSPICE .VEC**

Table:

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Input** | | | | | | | **Current state** | | | |
| **A0** | **A1** | **A2** | **A3** | **CIN** | **RST** | **PHI** | **FF0** | **FF1** | **FF2** | **FF3** |
| **0** | **0** | **0** | **0** | **0** | **0** | **0** | **1** | **1** | **1** | **1** |
| **0** | **1** | **0** | **0** | **0** | **1** | **1** | **0** | **1** | **0** | **0** |
| **1** | **0** | **1** | **0** | **0** | **1** | **0** | **1** | **0** | **1** | **0** |
| **0** | **0** | **0** | **0** | **0** | **1** | **1** | **0** | **0** | **0** | **0** |
| **1** | **0** | **0** | **0** | **0** | **1** | **0** | **1** | **0** | **0** | **0** |
| **1** | **1** | **1** | **0** | **0** | **1** | **1** | **1** | **1** | **1** | **0** |
| **0** | **0** | **0** | **0** | **0** | **1** | **0** | **0** | **0** | **0** | **0** |
| **1** | **0** | **0** | **0** | **0** | **1** | **1** | **1** | **0** | **0** | **0** |
| **0** | **0** | **1** | **0** | **0** | **1** | **0** | **0** | **0** | **1** | **0** |
| **0** | **0** | **0** | **0** | **0** | **1** | **1** | **0** | **0** | **0** | **0** |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | **OUTPUT** |  |  | **Next state** | | | |
| **SOUT0** | **SOUT1** | **SOUT2** | **SOUT3** | **COUT** | **FF0** | **FF1** | **FF2** | **FF3** |
| **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** |
| **0** | **0** | **0** | **0** | **0** | **1** | **0** | **1** | **0** |
| **1** | **0** | **1** | **0** | **0** | **1** | **0** | **1** | **0** |
| **1** | **0** | **1** | **0** | **0** | **0** | **1** | **1** | **0** |
| **0** | **1** | **1** | **0** | **1** | **0** | **1** | **1** | **0** |
| **0** | **1** | **1** | **0** | **1** | **0** | **1** | **1** | **0** |
| **0** | **1** | **1** | **0** | **0** | **0** | **1** | **1** | **0** |
| **0** | **1** | **1** | **0** | **0** | **0** | **1** | **0** | **1** |
| **0** | **1** | **0** | **1** | **1** | **0** | **1** | **0** | **1** |
| **0** | **1** | **0** | **1** | **0** | **0** | **1** | **0** | **0** |

Waveforms:



COUT goes high for b’0001 + b’0111

SOUT3 & 2 change for b’0010 + b’0010 (7)

RST occurs

SOUT0 & 2 go High for b’0101 (5)

SOUT0 & 1 change for b’0111 (7)

**Step A3: Design a layout for your accumulator**

Layout:

A picture containing clock, table

Description automatically generated

Description of changes:

The changes required were 1) adding labels for the input/output pins at the 4-bit level, 2) vertically aligning the left and right edges of VDD, GND, CIN, COUT, RST, and PHI so there was no unnecessary overlap in metals, and 3) modifying the rulers.

**Step A4: Testing**

DRC:

Graphical user interface, text, application, email

Description automatically generated

LVS:

Graphical user interface, application, table

Description automatically generated

Waveforms:

Graphical user interface

Description automatically generated

**Step A5: Determine Maximum Clock Frequency**

SUBMIT: A description of the critical path, and discussion of why you believe it is the critical path.

Critical path:

Pictured below, the critical path is the longest path from the first flip-flop to the last flip-flip, where the carry bit is activated in each bitslice.

Diagram

Description automatically generated

The test sequence that exercises the critical path generates a carry at each cycle, and can be achieved by pushing all 1 values after reset:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **A0** | **A1** | **A2** | **A3** | **RST** | **CIN** |
| **0** | **0** | **0** | **0** | **0** | **0** |
| **1** | **1** | **1** | **1** | **1** | **1** |
| **1** | **1** | **1** | **1** | **1** | **1** |
| **1** | **1** | **1** | **1** | **1** | **1** |
| **1** | **1** | **1** | **1** | **1** | **1** |
| **1** | **1** | **1** | **1** | **1** | **1** |

Correct operation at maximum clock frequency:

Graphical user interface

Description automatically generated

Frequency = **16.393** GHz

Incorrect operation 20% above maximum clock frequency:

Graphical user interface

Description automatically generated

Incorrect value for SOUT3, should go back low

Frequency = **16.666** GHz

Part B: Standard Cell Accumulator Design (Synthesis and P&R)

**Step B1: Synthesis using Synopsys Design Compiler**

*SUBMIT: What is the statement “set modname” used for and how did you change it?*

“set modname” is used to reference the top-level, to match the module name in the Verilog file which is used in files such as Constraints.tcl. This improves maintainability of the code since it only needs to be changed in one place. I set modname to accumulator since the Verilog file is accumulator.v.

*SUBMIT: Why do you think the specification of clock name is necessary?*

Designs may have multiple clocks, so specification avoids conflicts and ambiguity.

*SUBMIT: If you want to achieve optimal area, how would you specify area constraint?*

Excerpt from Constraints.tcl:

#---------------------------------------------------------

# Now set the GOALS for the compile

# In most cases you want minimum area, so set the

# goal for maximum area to be 0

#---------------------------------------------------------

 set\_max\_area 0

*SUBMIT: The synthesis tool needs a target clock frequency to perform performance optimization and timing analysis. How do you specify the frequency in your constraints?*

#---------------------------------------------------------

# Specify a CLK\_PER ns clock period with 50% duty cycle

# and a skew of 50ps

#---------------------------------------------------------

 set CLK\_PER  20.0

*SUBMIT: Does your circuit have timing violations?*

No, it appears that data arrives 1.9374 seconds before the required time, so timing is satisfied.

*SUBMIT: What is the estimated area of your circuit? How many flip-flops does your circuit use?*

Estimated total area: 39.9000

# of flip-flips: 4

  DFF\_X1 SOUT\_reg\_0\_ ( .D(n43), .CK(PHI), .Q(SOUT[0]) );

  DFF\_X1 SOUT\_reg\_1\_ ( .D(n42), .CK(PHI), .Q(SOUT[1]) );

  DFF\_X1 SOUT\_reg\_2\_ ( .D(n41), .CK(PHI), .Q(SOUT[2]) );

  DFF\_X1 SOUT\_reg\_3\_ ( .D(n40), .CK(PHI), .Q(SOUT[3]) );

*SUBMIT: Take a look at the netlist file which will be used for physical implementation. Explain what is RTL synthesis based on your understanding so far. Is it technology dependent?*

RLT synthesis is the process of expanding and transforming a more abstract Verilog module or several modules that describe a circuit into a specific netlist to be implemented.

*SUBMIT: What is the maximum frequency based on synthesis?*

Maximum frequency: 55.56 GHz

CLK\_PER: 1.8

*SUBMIT: What is the lower frequency that produced a different implementation?*

Lower frequency: 50.00 GHz

CLK\_PER: 2.0

SUBMIT: Compare the area and the cells used in these two implementations.

@ 55.56 GHz, area = 53.7320, number of cells used = 37.

@ 50.00 GHz, area = 54.7960, number of cells used = 39.

**Step B2: Place and route using Cadence Encounter**

Physical view without wire & via layers:

A screen shot of a computer

Description automatically generated

Implement the accumulator in a rectangular floorplan.

Physical view with wire & via layers:

A screen shot of a computer

Description automatically generated

SUBMIT: What is the silicon area utilization?

Area = 8.540 \* 6.995 = 59.7373

Part C: Comparing Results from Custom and Standard Cell

**Step C1: Compare results**

*SUBMIT: Table comparing Part A results to Part B results in terms of maximum frequency, power, area, metal layers used, and height of the cells in each row. Explain how you obtained the numbers for Part B.*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Part** | **Max freq** | **Power** | **Area** | **# metal layers** | **Cell Height** |
| A | 16.393 GHz | NA | 195.125 | 2 | 3.125 |
| B | 55.56 GHz | NA | 59.7373 | 10 | 1.399 |

For Part B, I already had max frequency calculated. I found area and cell heigh by using a ruler on the layout. I found metal layers by looking at the layers window in the Encounter. Power was not asked, so the professor told us not to worry about it, but I found you can generate a power.rpt file with the Power Libraries.

*Discussion of what accounts for the differences, and the implications of those differences.*

The area of the synthesized version was almost 4 times as small, because it used 10 metal layers while I only used 2 in my custom layout. This change accounts for all of the comparison metrics, including max frequency, because with a smaller area, wires are shorter and frequency can be increased.

**Step C2: Compare Critical Path**

*SUBMIT: Describe the critical path from Design Compiler and explain where you found it. Does it begin and end at the same place as your critical path? Explain.*

Using “timing\_max\_slow\_holdfixed\_tut1.rpt, this is the critical path:

A[0] (in)

U18/ZN (OAI21\_X1)

U20/ZN (NAND3\_X1)

U39/ZN (NAND4\_X1)

U40/ZN (OAI221\_X1)

COUT (out)

Yes, this seems to match the critical path that I found, since is starts at A0 and ends at the final COUT. This means it also goes through all the bitslices and ends in the same spot.