Alexander Maso
ECE-527 Fall 2023

Project 2 – Pipelined Carry Select Adder

Honor Pledge

On the cover page of your project report, please type/write down the following sentence in the underlined section below and sign/type your name under it.

"On my honor, I have neither given nor received unauthorized aid on this project."

Do NOT copy and paste the sentence, please do type/write every word in this sentence. Failure to write down this sentence and sign your name or failure to honor the content of this sentence will result in 0 points in the project.

Write the above sentence in underlined section below:
On my honor, I have neither given nor received unauthorized aid on this project

Type or Sign Your Name here:

Alexander Maso

Alexander Maso
ECE-527 Fall 2023

Project 2 – Pipelined Carry Select Adder

Task 1 – Create a Pipelined Four Bit Carry Select Adder

The initial task for this project (ignoring the tutorials for Design Compiler and Prime Time) was to write a 4 bit pipelined carry select adder in system Verilog. We were asked to use two separate 2-bit adders as basic blocks and we also provide with a diagram, seen below.

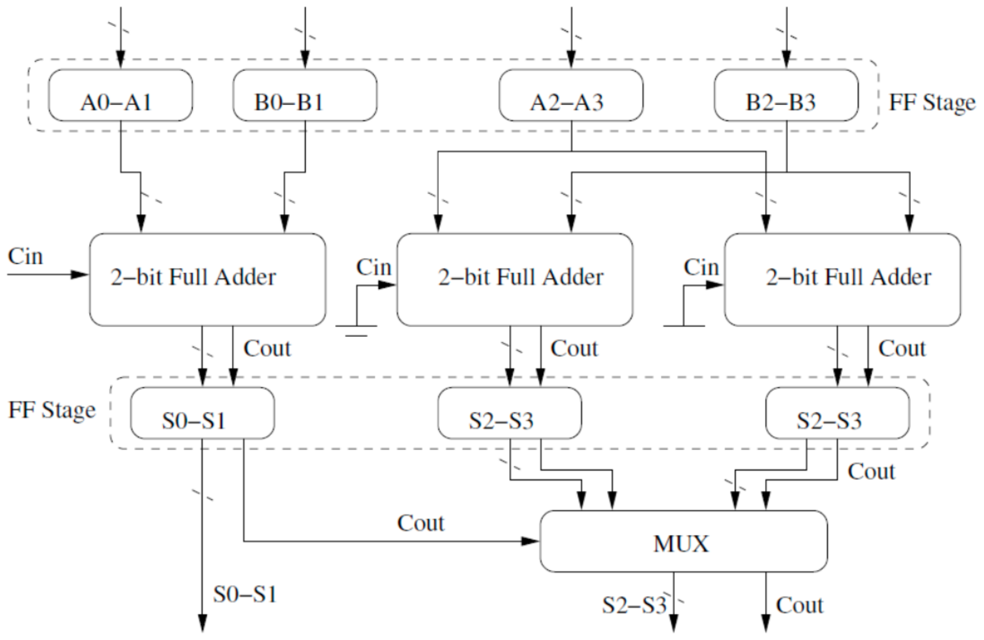


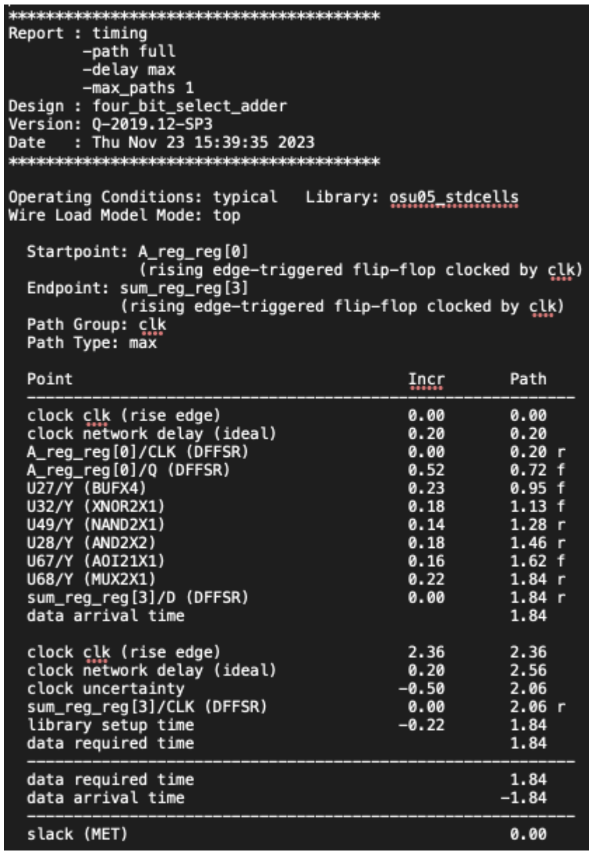
Fig 1

In addition to creating the adder itself, we were also asked to create a simple testbench to verify that it our design works correctly. This was one of the more challenging tasks due to the combinational input from Cin and the testbench has been included in the deliverables.

Task 2 – Synthesize with Design Compiler

Once we had our design working correctly, we were asked to synthesize to a netlist using Design Compiler and with the help of DC, mininize the clock period. The netlist and other DC files have been included in the deliverables. In additon to this, I also used Design Compiler to create a schematic of my design which I have included at the end of this report.

Task 3 – Static Timing Analysis with Prime Time

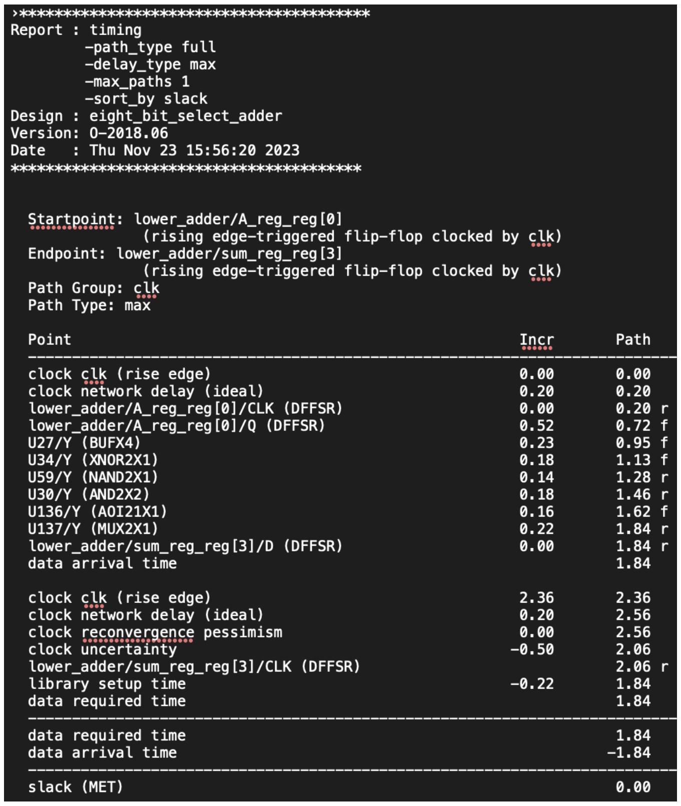


```
****************************************
Report : timing
        -path full
        -delay max
        -max_paths 1
Design : four_bit_select_adder
Version: Q-2019.12-SP3
Date   : Thu Nov 23 15:39:35 2023
****************************************

Operating Conditions: typical   Library: osu05_stdcells
Wire Load Model Mode: top

  Startpoint: A_reg_reg[0]
            (rising edge-triggered flip-flop clocked by clk)
  Endpoint: sum_reg_reg[3]
            (rising edge-triggered flip-flop clocked by clk)
  Path Group: clk
  Path Type: max

  Point                                    Incr       Path
  -----------------------------------------------------------
  clock clk (rise edge)                    0.00       0.00
  clock network delay (ideal)              0.20       0.20
  A_reg_reg[0]/CLK (DFFSR)                 0.00       0.20 r
  A_reg_reg[0]/Q (DFFSR)                   0.52       0.72 f
  U27/Y (BUFX4)                            0.23       0.95 f
  U32/Y (XNOR2X1)                          0.18       1.13 f
  U49/Y (NAND2X1)                          0.14       1.28 r
  U28/Y (AND2X2)                           0.18       1.46 r
  U67/Y (AOI21X1)                          0.16       1.62 f
  U68/Y (MUX2X1)                           0.22       1.84 r
  sum_reg_reg[3]/D (DFFSR)                 0.00       1.84 r
  data arrival time                                   1.84

  clock clk (rise edge)                    2.36       2.36
  clock network delay (ideal)              0.20       2.56
  clock uncertainty                       -0.50       2.06
  sum_reg_reg[3]/CLK (DFFSR)               0.00       2.06 r
  library setup time                      -0.22       1.84
  data required time                                  1.84
  -----------------------------------------------------------
  data required time                                  1.84
  data arrival time                                  -1.84
  -----------------------------------------------------------
  slack (MET)                                         0.00
```

Next we were asked to run Static Timing Analysis on our netlist using the Prime Time. We used the minimum clock period from Design Compiler initially but were encouraged to further minimize the clock period in Prime Time. Next we had to report our timing and determine the throughput of our adder. For my design, the minimum clock period that I was able to achieve was 2.36 ns (as show in the timing report below). My design has two flip flop stages, as shown in the spec above, and therefore takes two whole clock cycles to calculate each sum. However, the maximum throughput of each additional operation (the time between sequential outputs) is one clock cycle per sum, or 2.36 ns.

Task 4 – Eight Bit Carry Select Adder

The last requirement of this project was to repeat the process but this time create a pipelined eight bit carry select adder. I chose to follow the same implementation I used when creating the four bit carry select adder, but instead of instantiating three modules of a two bit adder (as in Task 1), I instantiated my pipelined four bit adder three times. This meant the primary difference between my adders was that the eight bit adder had basic blocks that were pipelined as well. Therefore my eight bit adder design took an additional two cycles for each operation, for a total of four clock cycles for a single addition. As can be seen in the timing report shown at the right, the minimum clock period was the same as it was for my four bit adder: 2.36 ns. Therefore, this adder requires four clock cycles, or 9.44ns, to calculate each individual sum. However, since it is pipelined, the maximum throughput is a single clock cycle since a sum is produced every clock cylce or 2.36ns

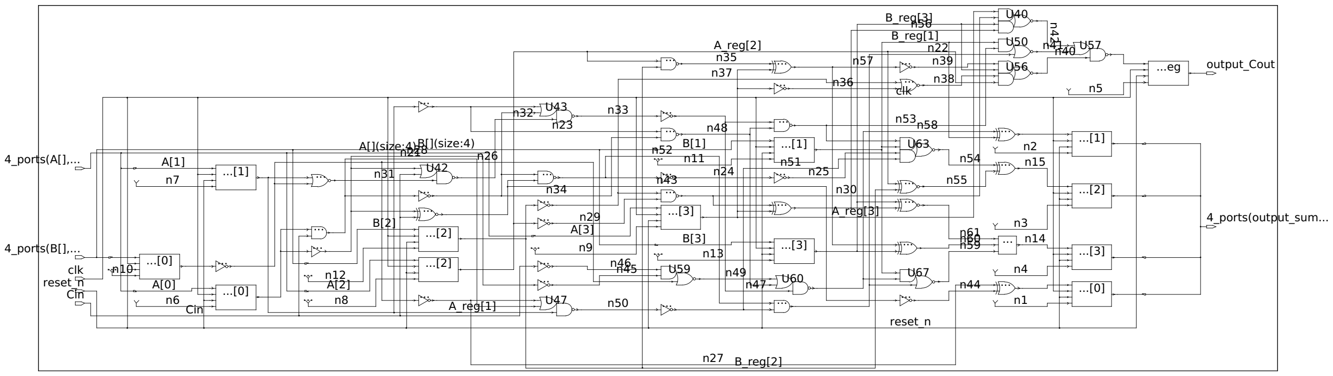


```
>****************************************
Report : timing
        -path_type full
        -delay_type max
        -max_paths 1
        -sort_by slack
Design : eight_bit_select_adder
Version: O-2018.06
Date   : Thu Nov 23 15:56:20 2023
****************************************


  Startpoint: lower_adder/A_reg_reg[0]
              (rising edge-triggered flip-flop clocked by clk)
  Endpoint: lower_adder/sum_reg_reg[3]
              (rising edge-triggered flip-flop clocked by clk)
  Path Group: clk
  Path Type: max

  Point                                            Incr       Path
  ---------------------------------------------------------------------
  clock clk (rise edge)                            0.00       0.00
  clock network delay (ideal)                      0.20       0.20
  lower_adder/A_reg_reg[0]/CLK (DFFSR)             0.00       0.20 r
  lower_adder/A_reg_reg[0]/Q (DFFSR)               0.52       0.72 f
  U27/Y (BUFX4)                                    0.23       0.95 f
  U34/Y (XNOR2X1)                                  0.18       1.13 f
  U59/Y (NAND2X1)                                  0.14       1.28 r
  U30/Y (AND2X2)                                   0.18       1.46 r
  U136/Y (AOI21X1)                                 0.16       1.62 f
  U137/Y (MUX2X1)                                  0.22       1.84 r
  lower_adder/sum_reg_reg[3]/D (DFFSR)             0.00       1.84 r
  data arrival time                                           1.84

  clock clk (rise edge)                            2.36       2.36
  clock network delay (ideal)                      0.20       2.56
  clock reconvergence pessimism                    0.00       2.56
  clock uncertainty                               -0.50       2.06
  lower_adder/sum_reg_reg[3]/CLK (DFFSR)                      2.06 r
  library setup time                              -0.22       1.84
  data required time                                          1.84
  ---------------------------------------------------------------------
  data required time                                          1.84
  data arrival time                                          -1.84
  ---------------------------------------------------------------------
  slack (MET)                                                 0.00
```

Alexander Maso
ECE-527 Fall 2023

Conclusion and Schematics:

Pipelined Four Bit Carry Select Adder
Maximum Throughput = 2.36ns ≈ 423728814 sums/second



Pipelined Eight Bit Carry Select Adder
Maximum Throughput = 2.36ns ≈ 423728814 sums/second