Frequency Control Word Multiplexer

ECE 429 Project Report

Aaron Roggow
Chase Greenhagen
Youssef Yazqi
Sam Mindas

Table of Contents

Functional Description

The Frequency Control Word (FCW) Multiplexer is an 8-bit, 2 to 1 mux with a synchronizable output. The inputs can be selected using the *sel* line. The device supports input selection changes to rates of at least 20 MHz.

Within the context of the Backscatter RFID Chip, the device is intended to interface directly with one instance of the Numerically Controlled Oscillator and is controlled by both the SPI Interface and the Data Transmit Controller. The device should be synchronized with the NCO's source clock. The output signal is designed to be accurate at the rising clock edge. The system is intended to operate at 5 volts.

This data sheet details the functionality of the FCW Multiplexer based on a 0.5µm process.

Block Diagram

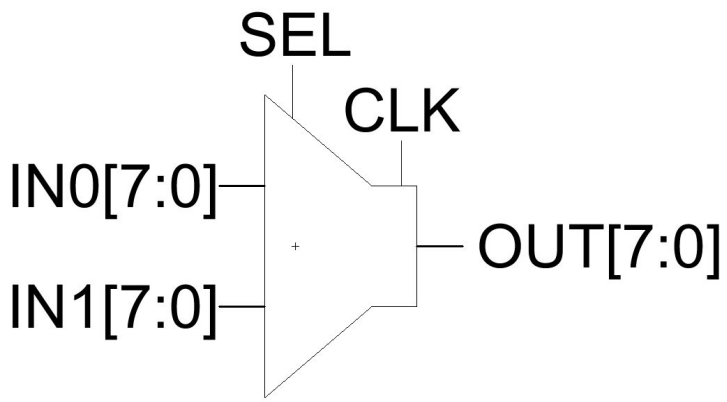


Figure 1: Block diagram of the FCW Multiplexer

Signal Descriptions

The external signals as shown in the block diagram are as follows:

- **sel:** a single bit selection line to select an output between the options of *in0[7:0]* and *in1[7:0].* Use an input of 0 to select in0, and an input of 1 to select in1. This input is expected to be provided by the Data Transmit Controller.
- **in1[7:0]:** An 8-bit bus that is selectable for output via an input of '1' on the *sel* input. This input is expected to be provided by the SPI Interface.

- **in0[7:0]:** An 8-bit bus that is selectable for output via an input of ‘0’ on the *sel* input. This input is expected to be provided by the SPI Interface.
- **clk:** The clock signal driving the output synchronization. This is expected to be provided by the NCO clock source.
- **out[7:0]:** An 8-bit output that will reflect the selection between the two inputs at the time of the most recent rising clock edge.

This data is summarized in the following table:

Signal Name	Direction	Bus Size
sel	input	1
in0	input	8
in1	input	8
clk	input	1
out	output	8

Specifications

The provided specifications are available in the ‘429 final project’ Google Drive folder or on the GitHub repository under the name ‘specifications.pdf’. The following is the portion relevant to the FCW Multiplexer:

> "A multiplexer SHALL switch its output between two 8-bit values *in0[7:0]* and *in1[7:0]* according to the state of input signal sel. When *sel* = 0, the *in0* vector SHALL be passed to the output.
>
> When *sel* = 1, the *in1* vector SHALL be passed to the output. All input signals to the multiplexer, *in0*, *in1*, and *sel* MUST be synchronized by the same clock used to drive the NCO."

This can be summarized in the following truth table:

sel	out[7:0]
0	in0[7:0]
1	in1[7:0]

Our interpretation of these specifications are also illustrated in the following timing diagram:

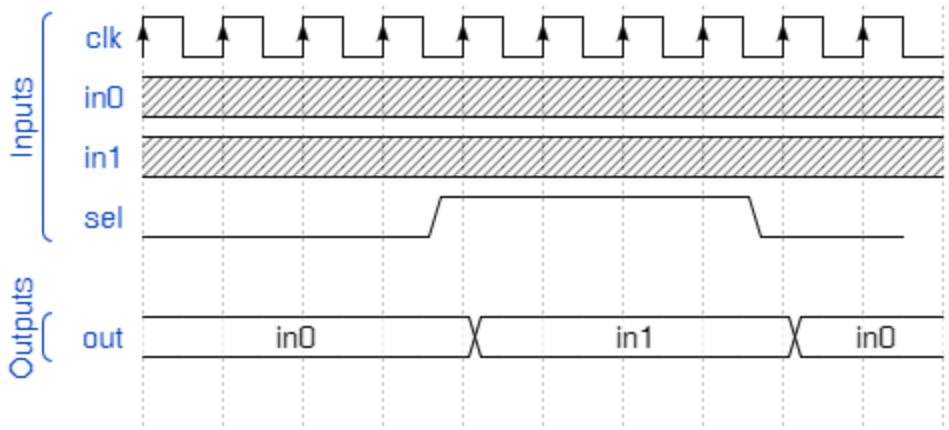


Figure 2: Timing diagram of operation specifications. Created with wavedrom.com. See Appendix B.1 for more information.

As already noted in the functional description, this device must be able to operate at a minimum frequency of 20MHz.

Schematic Simulations

Included in this section is the schematic for each sub block of the system, and an example operation verifying our measurements. Further details for each subsection can be found in Appendix A.

1. FCW Multiplexer

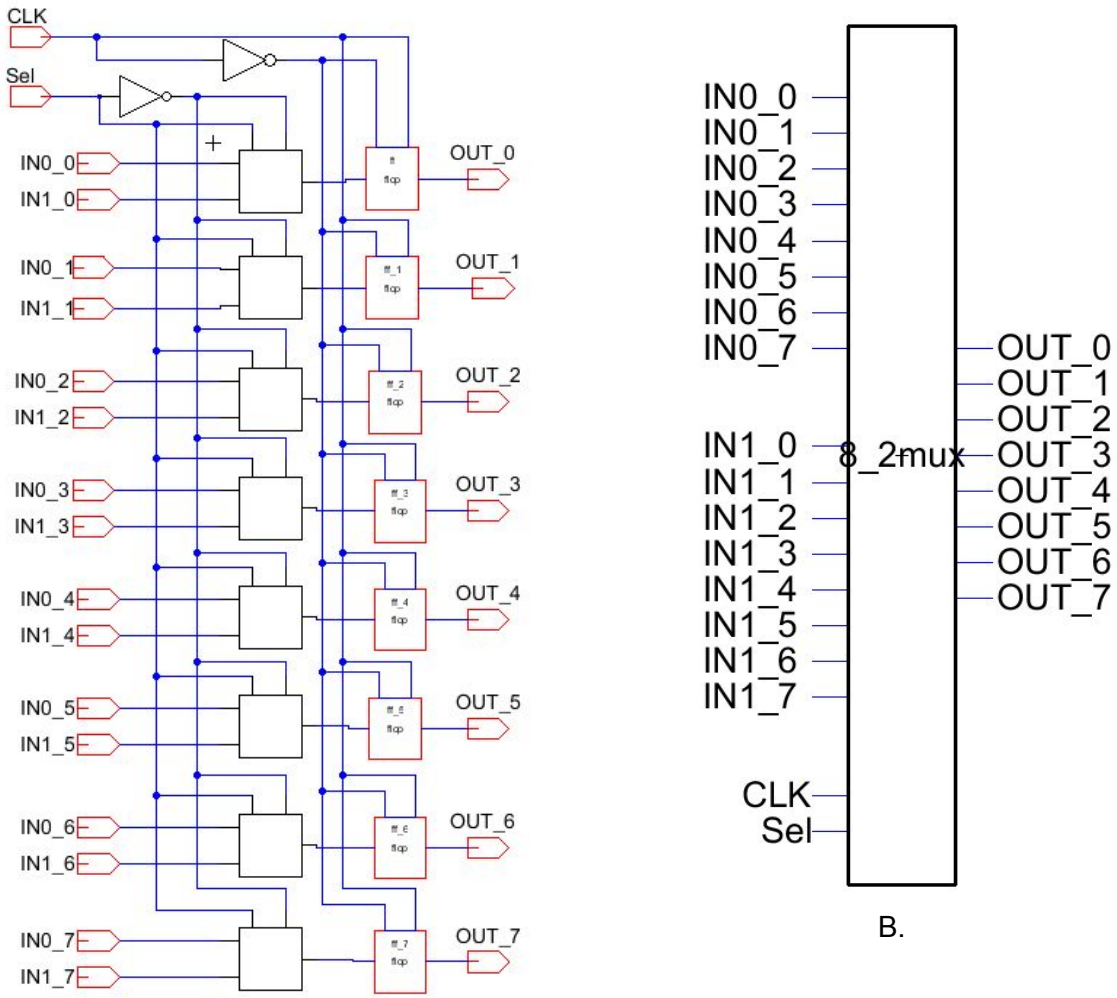


Figure 3: The schematic (A) and icon (B) for our FCW Multiplexer. A timing diagram for the system can be found in **Figure 2**.

Simulation

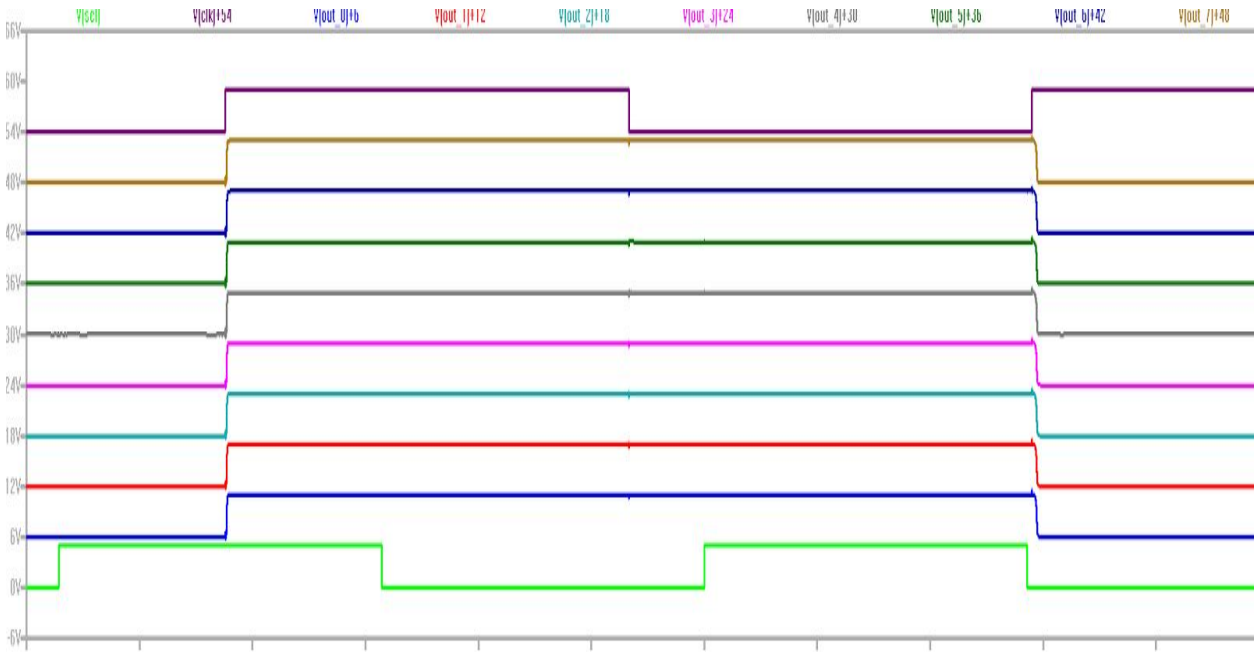
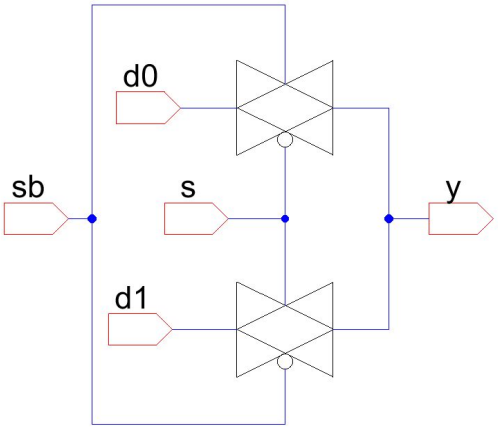


Figure 4: Simulation of FCW Multiplexer. Simulation of FCW Multiplexer. Clk is shown on bottom, and signal is on top. In the middle are the output signals. For this simulation all in1 signals were 1 and all in0 signals were 0.

A simulation of the FCW Multiplexer is shown above. From this simulation we gathered the following measurements.

Setup Time	400 ps
Hold Time	-190 ps
Low-High Delay	108.6 ps
High-Low Delay	293.1 ps

1(A). 2-1 Mux



A.

B.

S	d0	d1	y
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	1

C.

Figure 5: The schematic (A), icon (B), and truth table (C) for the 2-1 Mux

Simulation

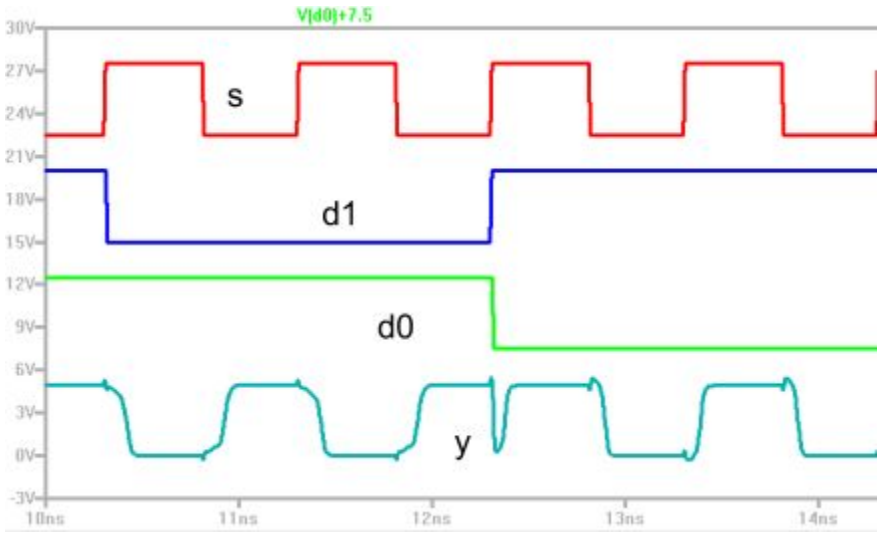
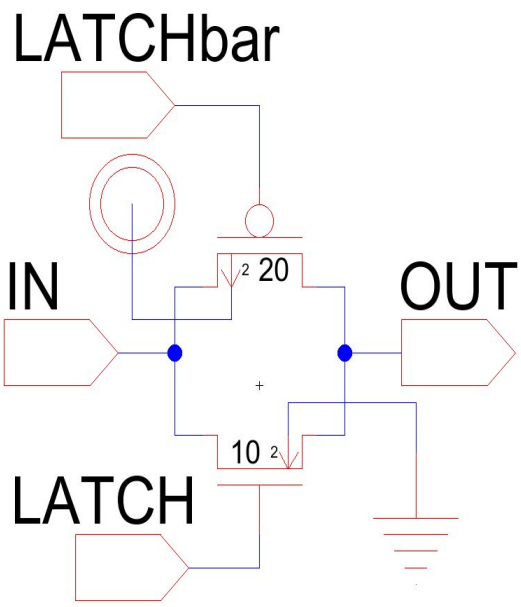


Figure 6: Simulation of 2-1 Mux operation. As can be seen above, *y* mimics *d0* when *s* is low and *d1* when *s* is high.

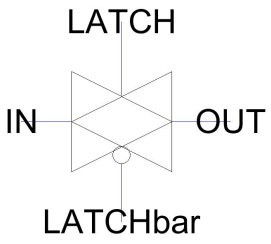
A simulation of the 2-1 Mux is shown above. From this simulation we gathered the following measurements.

Signal High to Low (D0 = 0, D1 = 1)	79.0ps
Signal Low to High (D0 = 0, D1 = 1)	86.5ps
Signal High to Low (D0 = 1, D1 = 0)	111.1ps
Signal Low to High (D0 = 1, D1 = 0)	117.5ps

1(A.i). Transmission Gate



A.



B.

IN	LATCHbar	LATCH	OUT
0	0	1	0
1	0	1	1
0	1	0	z
1	1	0	z

C.

Figure 7: (A) The schematic for our Transmission Gate. (B) Transmission Gate Icon (C) Transmission Gate Truth Table

Simulation

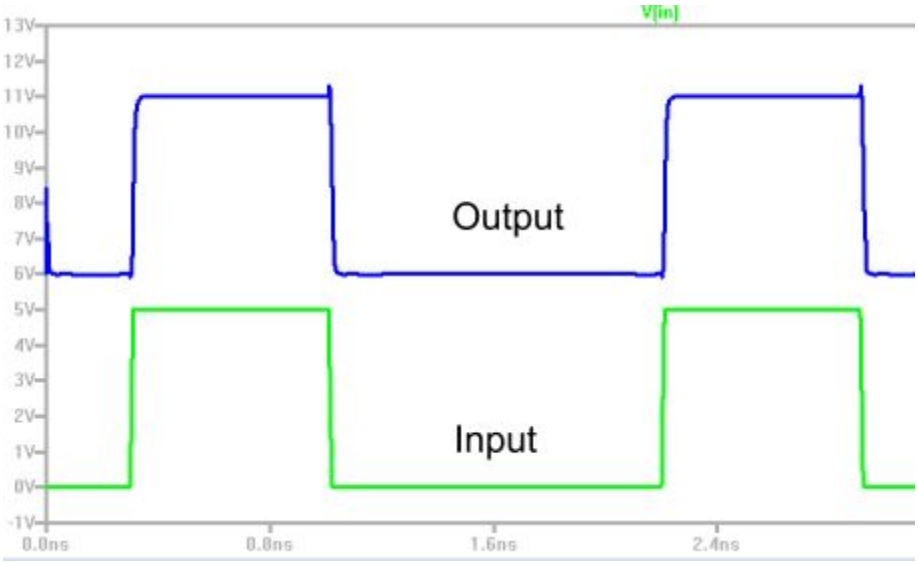
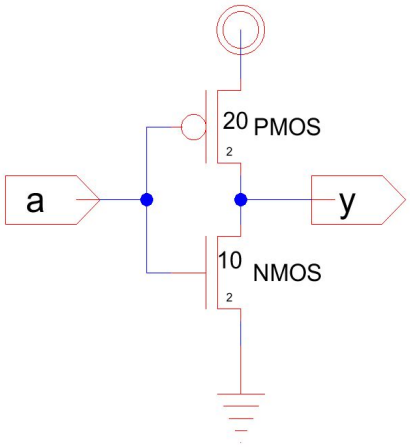


Figure 8: Simulation of Transmission gate operation. The LATCHbar input remained 0 and the LATCH input remained 1 during this simulation. As can be seen, the output mimics the input. When the values for the LATCHbar and LATCH inputs are swapped, the output would freeze and not change regardless of the input. This output could then be pulled up or down by whatever else is connected to it.

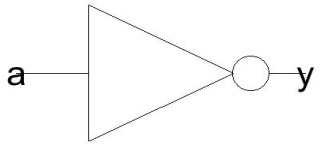
A simulation of the transmission gate is shown above. From this simulation we gathered the following measurements.

Low-High Delay	7.01 ps
High-Low Delay	6.66 ps

1(A.ii). Inverter



A.



B.

a	y
0	1
1	0

C.

Figure 9: Our inverter's schematic (A), icon (B), and truth table (C).

Our inverter was a decidedly standard inverter, with a boolean expression of y=!a. The truth table is shown in **Figure 9**.

Simulation

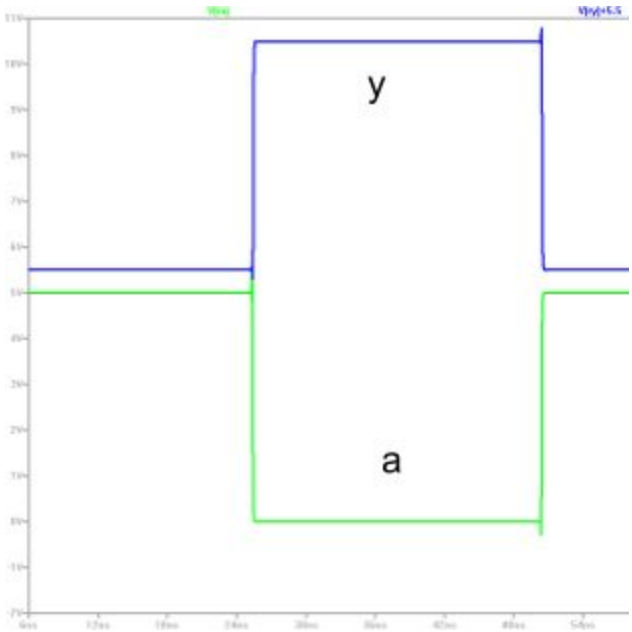
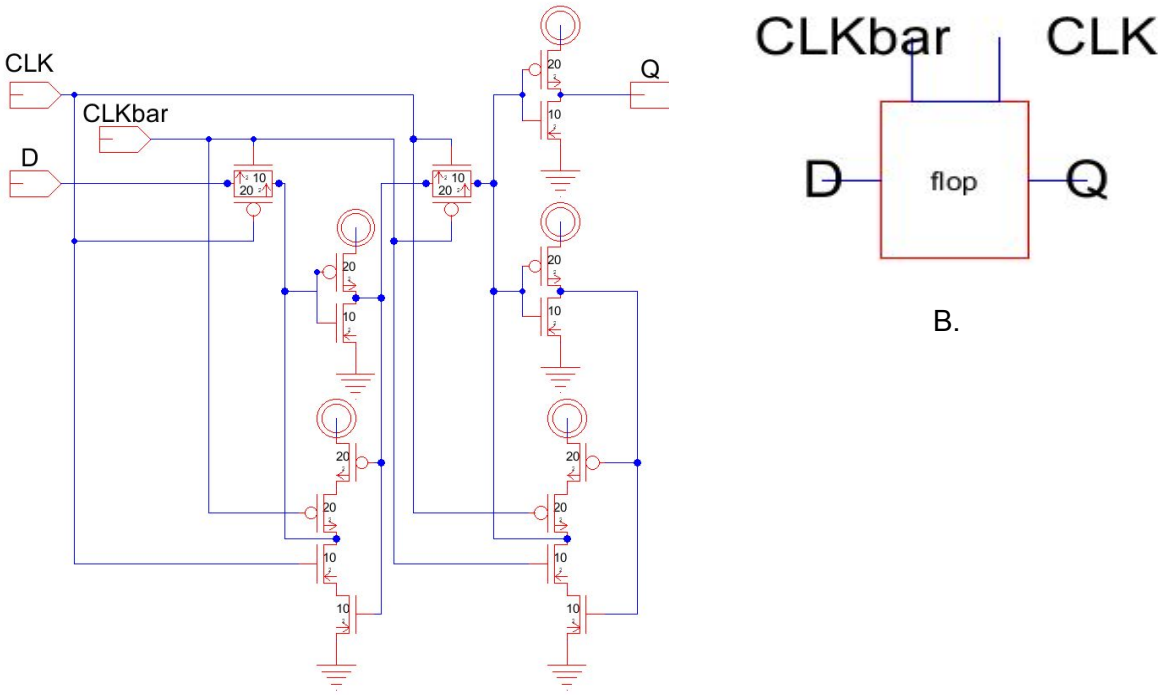


Figure 10: Simulation of inverter operation. As can be seen above, *y*, the output, takes the negated form of *a*, the input.

A simulation with two load inducing inverters on the input and output are shown above. From this simulation we gathered the following measurements.

Low-High Delay	8.95 ps
High-Low Delay	9.84 ps

1(B). D Flip Flop



A.

B.

Figure 11: The schematic (A) and icon (B) for our D Flip Flop

Simulation

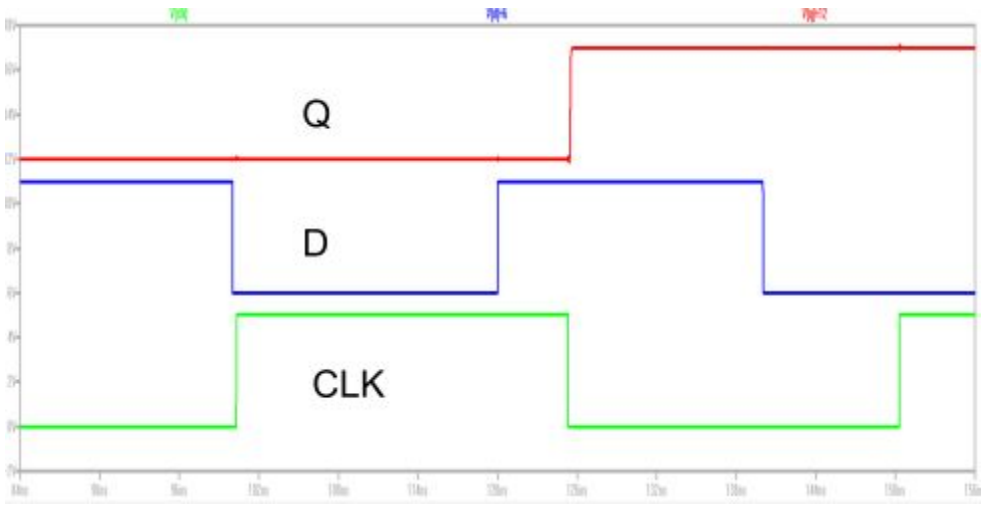


Figure 12: Simulation of D Flip Flop operation. The output, *Q*, properly goes from low to high on the positive edge of the clock while the input, *D*, is high. This shows the D Flip Flop is functioning properly.

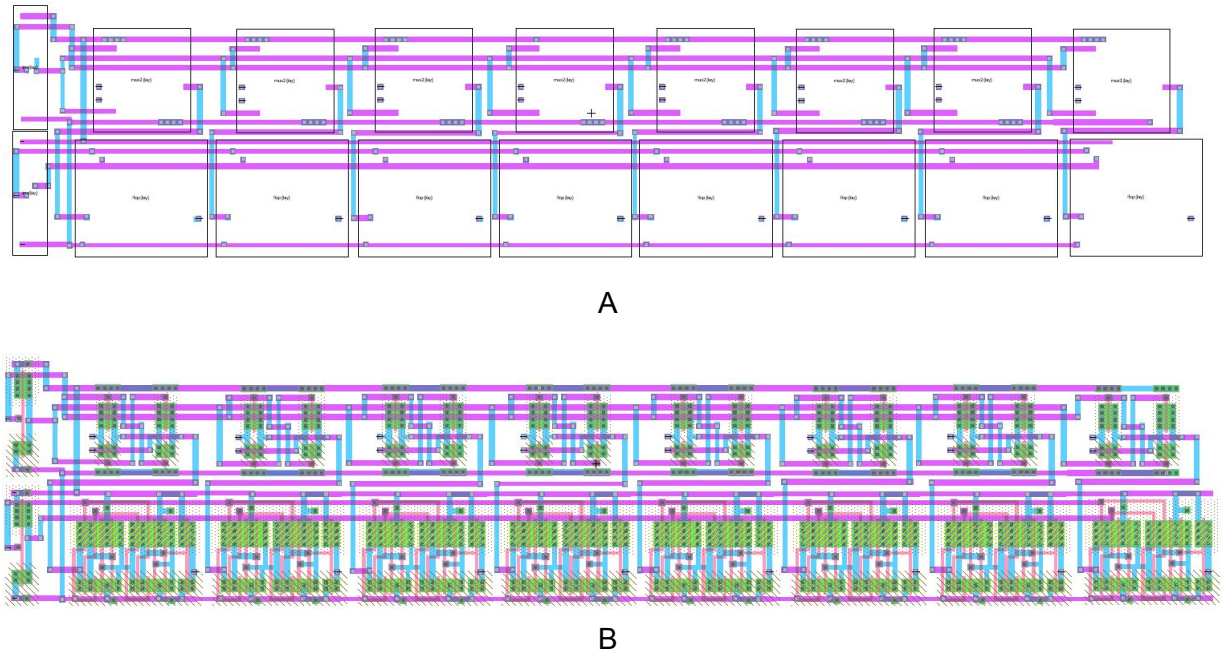
A simulation of the D-Flip Flop is shown above. From this simulation we gathered the following measurements.

Hold Time	152.2ps
Setup Time	251.5 ps
Clock-Q Delay	175.0 ps

Layout Simulations

Included in this section is the layout for each sub block of the system, and an example operation verifying our measurements. Further details for each subsection can be found in Appendix A.

1. FCW Multiplexer



A

B

Figure 13: Layout of the FCW Multiplexer in block view (A) and at the transistor level (B). Die area of 935 x 321.25 λ

Simulation

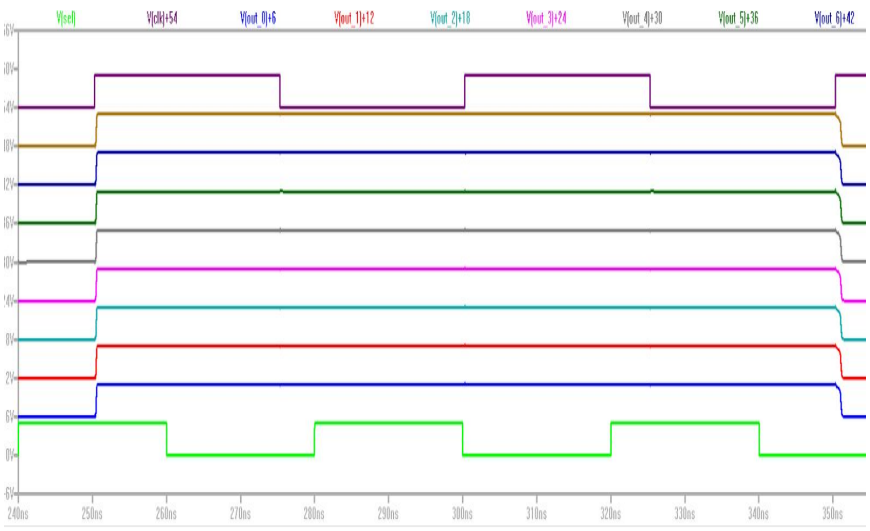
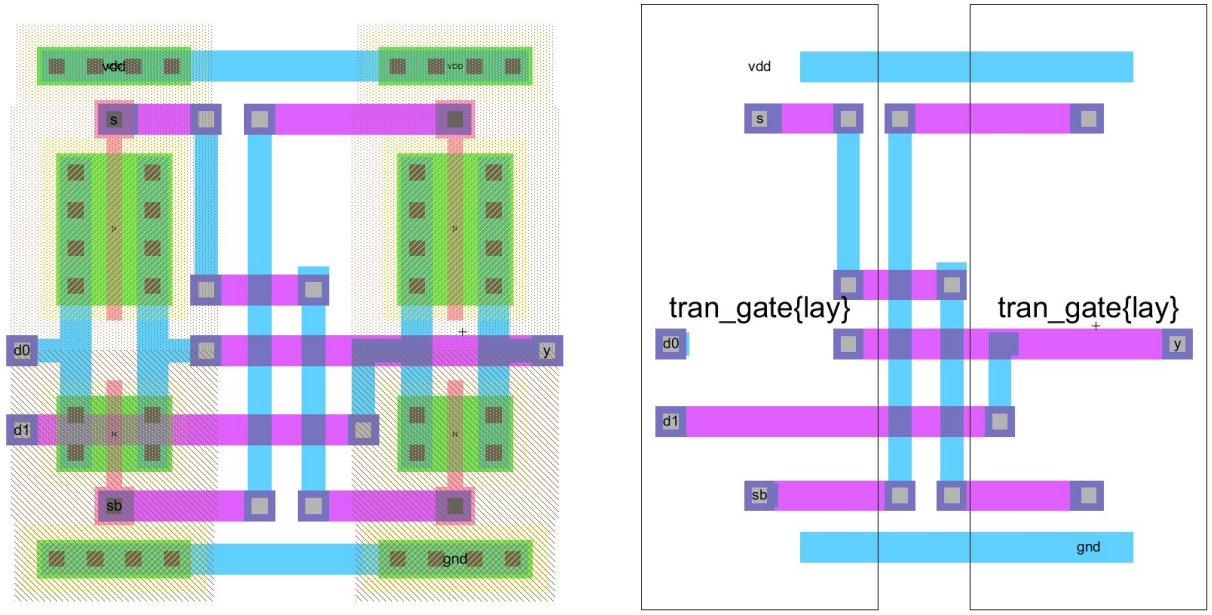


Figure 14: Simulation of FCW Multiplexer. Clk is shown on top, and signal is shown on bottom. In the middle are the output signals. For this simulation all in1 signals were 1 and all in0 signals were 0. As can be seen in the simulation, the output reflects in0 when the signal input is 0 and in1 when the signal input is 1. The output changes on the clock's rising edge.

A simulation of the FCW Multiplexer is shown above. From this simulation we gathered the following measurements. It is worth noting that the layout simulation shows a significantly higher delay than the schematic simulation did, though it is still well within a 20 MHz range.

Setup Time	1.2µs
Hold Time	-590ps
Low-High Delay	235.9ps
High-Low Delay	825.0ps

1(A). 2-1 Mux



A. B.

Figure 15: Layout of the 2-1 MUX at the transistor level (A) and the block level (B). Die area 76.5 x 82 λ

Simulation

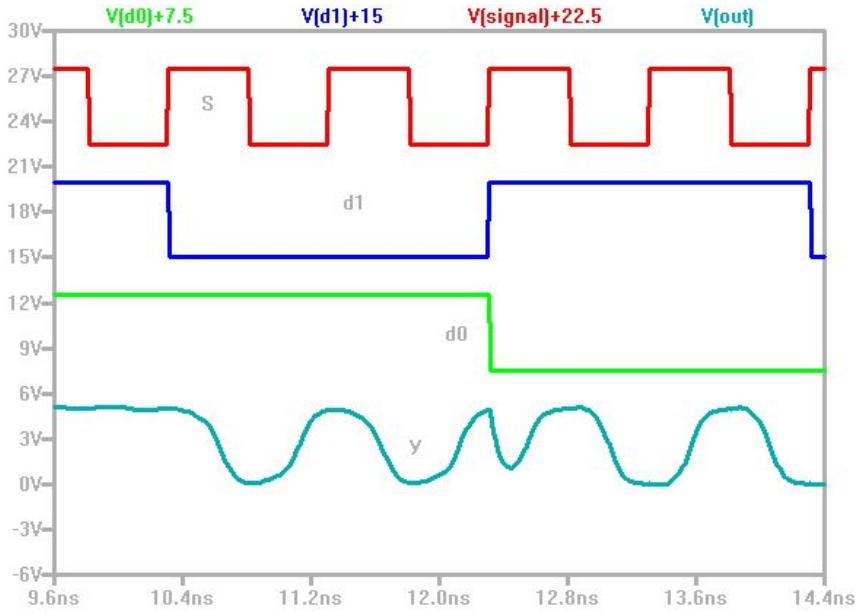


Figure 16: Simulation of the 2-1 Mux. Although the output appears to be struggling with the

given clock frequency (1GHz), the actual frequency that will be used (20 MHz) is much slower than the one simulated above.

We gathered the following measurements from the simulation. It is worth noting that the layout simulation shows a significantly higher delay than the schematic simulation did, though it is still well within a 20 MHz range.

High-Low Delay when d = 01	246.4ps
Low-High Delay when d = 01	264.4ps
High-Low Delay when d = 10	310.0ps
Low-High Delay when d = 10	330.9ps

1(A.i). Transmission Gate

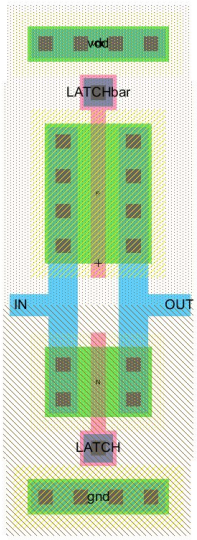


Figure 17: Layout of our transmission gate. Die area 32 x 82 λ

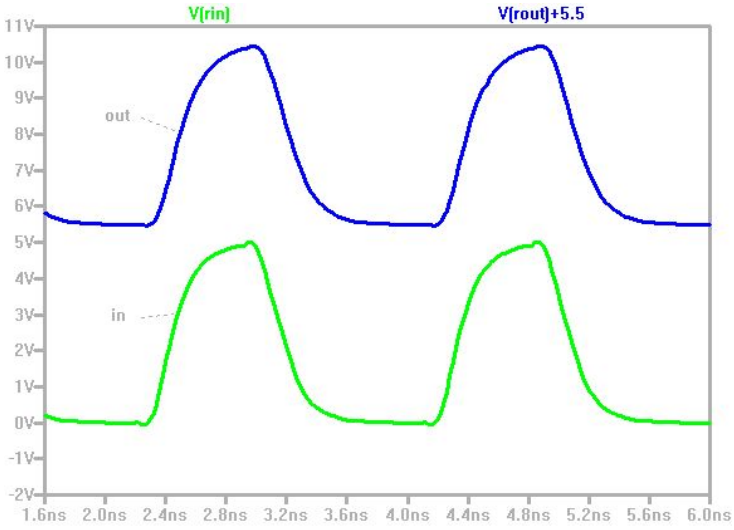


Figure 18: Simulation of the transmission gate. The input LATCHbar is kept low and the input LATCH is kept high throughout this simulation. When LATCH and LATCHbar swap values, the output freezes regardless of the input.

A simulation with two load inducing inverters on the input and output are shown above. From this simulation we gathered the following measurements. The delays for the layout are much higher than those from the schematic, but they are still very low so it is not quite as significant as what we are seeing in the entities above.

Low-High Delay	45.4 ps
High-Low Delay	48.5 ps

1(A.ii). Inverter

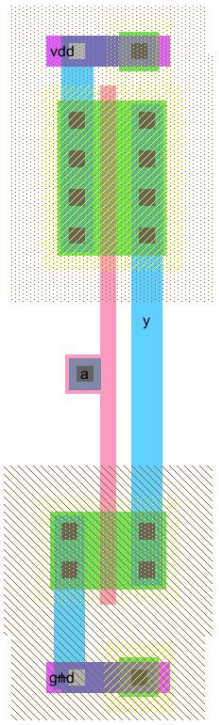


Figure 19: Layout of our inverter. Die area 27 x 98.5 λ

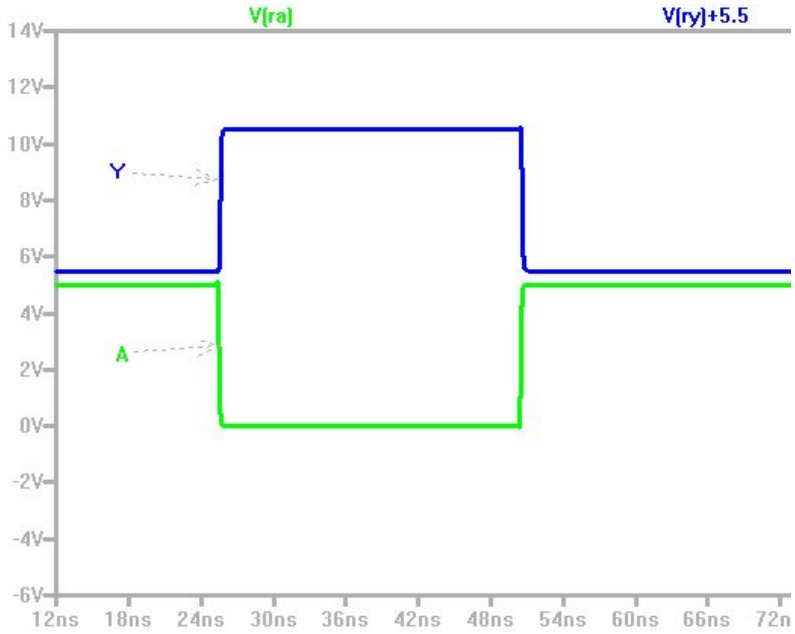


Figure 20: Simulation of the inverter. As can be seen above, the output, *y*, reflects the negated value of the input, *a*.

A simulation with two load inducing inverters on the input and output are shown above. From this simulation we gathered the following measurements. It is important to note that these time delays are much larger than the delays found in the schematic simulation.

Low-High Delay	99.4 ps
High-Low Delay	111.8 ps

1(B). D Flip Flop

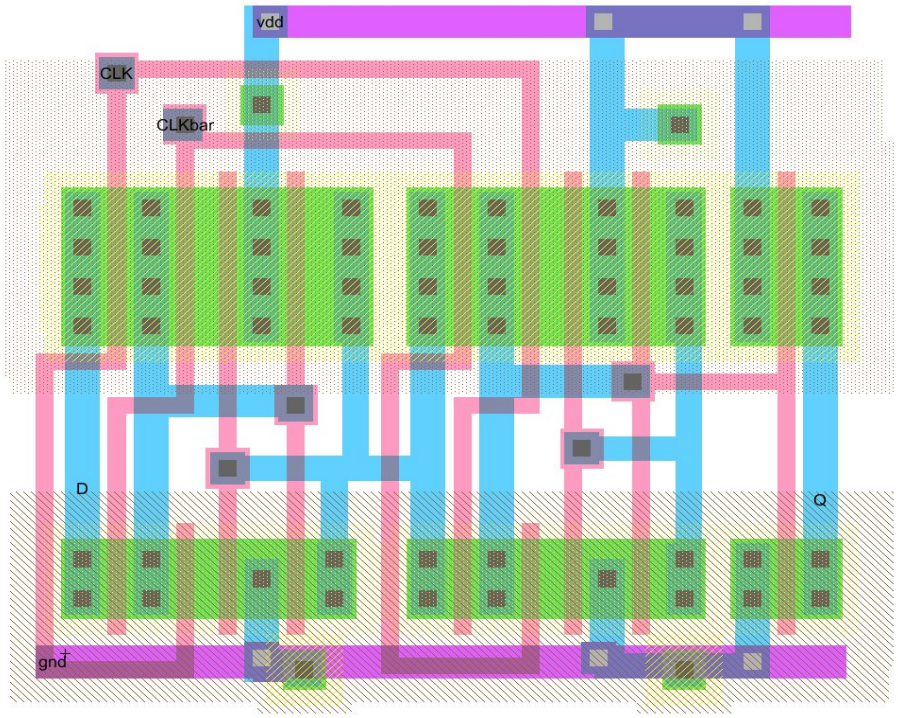


Figure 21: Layout of the D Flip Flop. Die area of 104 x 93 λ

Simulation

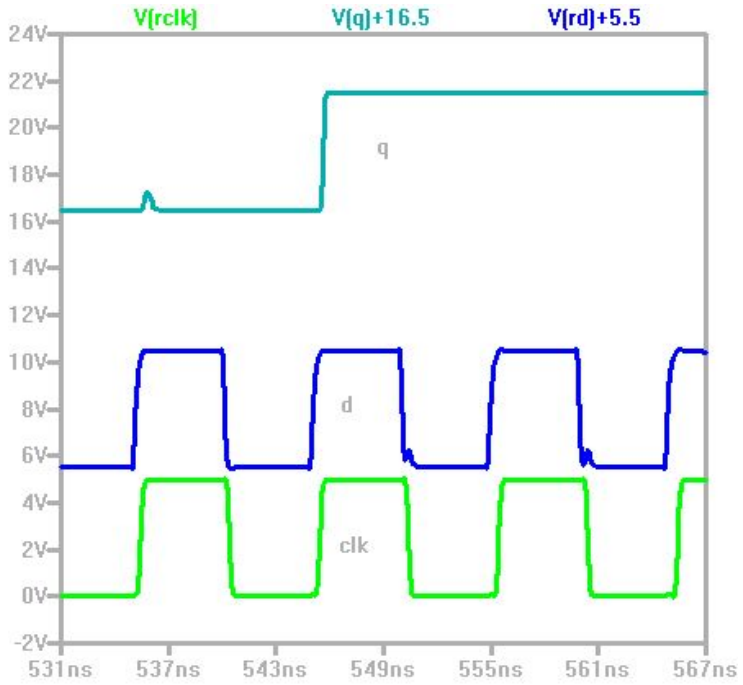


Figure 22: Simulation of D Flip Flop operation. As can be seen above, the *Q* value reflects the D value on the positive clock edge.

A simulation of the D-Flip Flop is shown above. From this simulation we gathered the following measurements. It is important to note that these delay times are much larger than their schematic simulation counterparts.

Hold Time	377.2ps
Setup Time	422.6ps
Clock-Q Delay	527.3ps

Die Area Summary

Inverter	27 x 98.5 = 2659.5 λ^2
Transmission Gate	32 x 82 = 2624 λ^2
2-1 Mux	76.5 x 82 = 6273 λ^2

D-FF	104 x 93 = 9672 λ^2
FCW Multiplexer	935 x 321.25 = 300,368.75 λ^2

*The area for each layout was calculated by Electric and found under the layout tab.

Project Commentary

It is important to note that it is possible to implement this multiplexer in much less die area. Whereas we implemented it with eight D flip-flops, the overall space of the project could be reduced if one D flip-flop was used to control the 1-bit signal line rather than to control the 8-bit output of the multiplexer. Both of these implementations force the multiplexer to be synchronized with the NCO clock; however, they each have unique timing delays. Whereas the implementation outlined in this datasheet has a single delay of clk to q (assuming that the signal has already generated the appropriate output of the multiplexer), the alternative multiplexer mentioned above would have a longer delay time of clk to q plus the combinational logic delay from the signal change to the output of the multiplexer. We chose not to use this alternative design because of time constraints for our deadline.

Another alternative implementation that reduces this delay would be to add the D flip-flops after the output of the multiplexer in addition to the D flip-flop that controls the signal value. This implementation would have the same delay as the implementation outlined in this datasheet, but the real-time output of the multiplexer/D flip-flop combination would be one clock cycle delayed.

Our execution of this project has given us a good understanding of how to approach the next large project we are given. For us, the most difficult portion of the project was creating the layout of the multiplexer and wiring everything without spacing or NCC errors. The most time-consuming portion was organizing all of our gathered information into this datasheet. We modified our design when moving from the schematic to the layout in order to synchronize the multiplexer on the positive clock-edge rather than the negative clock-edge. Although we began by working together on the schematic and layout portion of this project all at the same time (which was largely limited to taking turns working on the project while the others offered suggestions), our strategy improved by breaking the project into three tasks: one that worked on the layout, another that created the datasheet, and a third that worked on the presentation. Although we all switched between the three tasks, Youssef and Chase led in the construction of the layout, Aaron led in the writing of the datasheet, and Sam led in the creation of the presentation. In any future projects, we will consider changing our project organization so that one person is managing each section of the project, and delegating tasks to the other people in the group. This will ensure efficiency and help us keep track of everything that needs to be done.

Appendix A

Appendix A is intended for resources that may be relevant to an end user hoping to replicate results shown in this document.

1. FCW Multiplexer

For Delay

```
VVDD VDD 0 DC 5
VGND GND 0 DC 0
VCLK CLK 0 DC 0 PULSE 0 5 300p 10p 10p 25n
50n
VSEL Sel 0 DC 0 PULSE 0 5   0p 10p 10p 20n
40n
VIN1_0 IN1_0 0 DC 5
VIN1_1 IN1_1 0 DC 5
VIN1_2 IN1_2 0 DC 5
VIN1_3 IN1_3 0 DC 5
VIN1_4 IN1_4 0 DC 5
VIN1_5 IN1_5 0 DC 5
VIN1_6 IN1_6 0 DC 5
VIN1_7 IN1_7 0 DC 5
VIN2_0 IN0_0 0 DC 0
VIN2_1 IN0_1 0 DC 0
VIN2_2 IN0_2 0 DC 0
VIN2_3 IN0_3 0 DC 0
VIN2_4 IN0_4 0 DC 0
VIN2_5 IN0_5 0 DC 0
VIN2_6 IN0_6 0 DC 0
VIN2_7 IN0_7 0 DC 0
   .TRAN 300N UIC
.include C:/Electric/c5.txt
   .END
```

For setup and hold time

```
VVDD VDD 0 DC 5
VGND GND 0 DC 0
VCLK CLK 0 DC 0 PULSE 0 5 0p 10p 10p 25n
50n
VSEL Sel 0 DC 0 PULSE 0 5 0p 10p 10p 24.9n
49.9n
VIN1_0 IN1_0 0 DC 5
VIN1_1 IN1_1 0 DC 5
VIN1_2 IN1_2 0 DC 5
VIN1_3 IN1_3 0 DC 5
VIN1_4 IN1_4 0 DC 5
VIN1_5 IN1_5 0 DC 5
VIN1_6 IN1_6 0 DC 5
VIN1_7 IN1_7 0 DC 5
VIN2_0 IN0_0 0 DC 0
VIN2_1 IN0_1 0 DC 0
VIN2_2 IN0_2 0 DC 0
VIN2_3 IN0_3 0 DC 0
VIN2_4 IN0_4 0 DC 0
VIN2_5 IN0_5 0 DC 0
VIN2_6 IN0_6 0 DC 0
VIN2_7 IN0_7 0 DC 0
   .TRAN 100000N UIC
.include C:/Electric/c5.txt
   .END
```

Spice Code

```
.meas TRAN afall v(clk) when v(clk) = 2.5
td = 250n
.meas TRAN yrise v(out_0) when v(out_0) =
2.5 td = 250n
.meas TRAN lhdelay PARAM yrise-afall
```

```
.meas tran clkrise1 v(clk) when v(clk) =
2.5 td = 590n
.meas tran selrise1 v(sel) when v(sel) =
2.5 td = 590n
.meas tran setuptime param
clkrise1-selrise1
```

```
.meas TRAN arise v(clk) when v(clk) = 2.5
td = 300n
.meas TRAN yfall v(out_0) when v(out_0) =
2.5 td = 300n
.meas TRAN hldelay PARAM yfall-arise
*Note timing must be changed between layout
and schematic
```

```
.meas tran clkrise2 v(clk) when v(clk) =
2.5 td = 12.74u
.meas tran selfall1 v(sel) when v(sel) =
2.5 td = 12.74u
.meas tran holdtime param clkrise2-selfall1
```

.Meas Code

Schematic	Layout
afall: v(clk)=2.5 AT 2.50305e-007 yrise: v(out_0)=2.5 AT 2.50414e-007 **lhdelay: yrise-afall=1.08689e-010** arise: v(clk)=2.5 AT 3.00305e-007 yfall: v(out_0)=2.5 AT 3.00598e-007 **hldelay: yfall-arise=2.93092e-010**	afall: v(clk)=2.5 AT 2.50305e-007 yrise: v(out_0)=2.5 AT 2.50541e-007 **lhdelay: yrise-afall=2.35862e-010** arise: v(clk)=2.5 AT 3.50305e-007 yfall: v(out_0)=2.5 AT 3.5113e-007 **hldelay: yfall-arise=8.25018e-010**
clkrise1: v(clk)=2.5 AT 2.00005e-007 selrise1: v(sel)=2.5 AT 1.99605e-007 setuptime: clkrise1-selrise1=4e-010 clkrise2: v(clk)=2.5 AT 1.255e-005 selfall1: v(sel)=2.5 AT 1.25498e-005 holdtime: clkrise2-selfall1=1.9e-010	clkrise1: v(clk)=2.5 AT 6.00005e-007 selrise1: v(sel)=2.5 AT 5.98805e-007 setuptime: clkrise1-selrise1=1.2e-009 clkrise2: v(clk)=2.5 AT 1.275e-005 selfall1: v(sel)=2.5 AT 1.27494e-005 holdtime: clkrise2-selfall1=5.9e-010

.Meas Output

1(A). 2-1 Mux

```
Vvdd vdd 0 DC 5
Vgnd gnd 0 DC 0
Vd0 d0 0 DC 5 PULSE 0 5 300p 10p 10p 4n 8n
Vd1 d1 0 DC 5 PULSE 0 5 300p 10p 10p 2n 4n
vSignal Signal 0 DC 0 PULSE 0 5 300p 10p 10p 500p 1n
.tran 20n UIC
.include C:/Electric/c5.txt
```

Spice Code

```
*calculate when d0 = 0 and d1 = 1
.meas Tran A_Sign_HighToLow TRIG V(signal) VAL = 2.5      TD = 12.5n     FALL = 1
+                           TARG V(out)    VAL = 2.5      TD = 12.5n     FALL = 1

.meas Tran A_Sign_LowToHigh TRIG V(signal) VAL = 2.5      TD = 13n       RISE = 1
+                           TARG V(out)    VAL = 2.5      TD = 13n       RISE = 1
```

```
*calculate when d0 = 1 and d1 = 0
.meas Tran B_Sign_HighToLow TRIG V(signal) VAL = 2.5      TD = 10.5n     FALL = 1
+                           TARG V(out)    VAL = 2.5      TD = 10.5n     RISE = 1

.meas Tran B_Sign_LowToHigh TRIG V(signal) VAL = 2.5      TD = 11n       RISE = 1
+                           TARG V(out)    VAL = 2.5      TD = 11n       FALL = 1
```

.Meas Code

Schematic	Layout
A_Sign_HighToLow=7.89837e-011 FROM 1.2815e-008 TO 1.2894e-008 **A_Sign_LowToHigh=8.64721e-011** FROM 1.3305e-008 TO 1.33915e-008 **B_Sign_HighToLow=1.1109e-010** FROM 1.0815e-008 TO 1.09261e-008 **B_Sign_LowToHigh=1.17505e-010** FROM 1.1305e-008 TO 1.14225e-008	**a_sign_hightolow=2.46436e-010** FROM 1.2815e-008 TO 1.30614e-008 **a_sign_lowtohigh=2.64439e-010** FROM 1.3305e-008 TO 1.35694e-008 **b_sign_hightolow=3.1004e-010** FROM 1.0815e-008 TO 1.1125e-008 **b_sign_lowtohigh=3.30957e-010** FROM 1.1305e-008 TO 1.1636e-008

.Meas Output

1(A.i). Transmission Gate

```
VGND GND 0 DC 0
VVDD VDD 0 DC 5
VIN in 0 DC 0 PULSE 0 5 300p 10p 10p 700p 1.9n
VINA latch 0 DC 5
VINB latchbar 0 DC 0
.include C:/Electric/c5.txt
.tran 100n UIC
.END
```

Spice Code

```
.meas TRAN afall v(in) when v(in) = 2.5 td = .9n
.meas TRAN yrise v(out) when v(out) = 2.5 td = .9n
.meas TRAN lhdelay PARAM yrise-afall

.meas TRAN arise v(in) when v(in) = 2.5 td = 2.1n
.meas TRAN yfall v(out) when v(out) = 2.5 td = 2.1n
```

```
.meas TRAN hldelay PARAM yfall-arise
```

.Meas Code

Schematic	Layout
* C:\Users\smindas\Desktop\meas.mout afall: v(in)=2.5 AT 1.015e-009 yrise: v(out)=2.5 AT 1.02201e-009 **lhdelay: yrise-afall=7.01372e-012** arise: v(in)=2.5 AT 2.205e-009 yfall: v(out)=2.5 AT 2.21166e-009 **hldelay: yfall-arise=6.65776e-012**	afall: v(rin)=2.5 AT 1.27006e-009 yrise: v(rout)=2.5 AT 1.3155e-009 **lhdelay: yrise-afall=4.5438e-011** arise: v(rin)=2.5 AT 2.446e-009 yfall: v(rout)=2.5 AT 2.4945e-009 **hldelay: yfall-arise=4.84938e-011**

.Meas Output

1(A.ii). Inverter

```
VGND GND 0 DC 0
VVDD VDD 0 DC 5
VIN a 0 DC 0 PULSE 0 5 300p 10p 10p 25n 50n
.include C:/Electric/c5.txt
.tran 200n UIC
.END
```

Spice Code

```
.meas TRAN afall v(a) when v(a) = 2.5 td = 20n
.meas TRAN yrise v(y) when v(y) = 2.5 td = 20n
.meas TRAN lhdelay PARAM yrise-afall

.meas TRAN arise v(a) when v(a) = 2.5 td = 40n
.meas TRAN yfall v(y) when v(y) = 2.5 td = 40n
.meas TRAN hldelay PARAM yfall-arise
```

.Meas Code

Schematic	**Layout**
afall: v(a)=2.5 AT 2.5315e-008 yrise: v(y)=2.5 AT 2.5324e-008 **lhdelay: yrise-afall=8.95775e-012** arise: v(a)=2.5 AT 5.0305e-008 yfall: v(y)=2.5 AT 5.03148e-008 **hldelay: yfall-arise=9.84823e-012**	afall: v(ra)=2.5 AT 2.54988e-008 yrise: v(ry)=2.5 AT 2.55982e-008 **lhdelay: yrise-afall=9.93787e-011** arise: v(ra)=2.5 AT 5.0495e-008 yfall: v(ry)=2.5 AT 5.06068e-008 **hldelay: yfall-arise=1.11829e-010**

.Meas Output

1(B). D Flip Flop

```
VGND GND 0 DC 0
VVDD VDD 0 DC 5
VIN d 0 DC 0 PULSE 0 5 300p 10p 10p 5n 9.9n
VCLK clk 0 DC 0 PULSE 5 0 100p 10p 10p 5n 10n
VCLK_b clkbar 0 DC 0 PULSE 0 5 100p 10p 10p 5n 10n
.include C:/Electric/c5.txt
.tran 600n UIC
.END
```

Spice Code

```
.meas TRAN clkfall1 v(rclk) when v(rclk) = 2.5 td = 38n
.meas TRAN drise1 v(rd) when v(rd) = 2.5 td = 38n
.meas TRAN setuptime PARAM clkfall1-drise1

.meas TRAN clkfall2 v(rclk) when v(rclk) = 2.5 td = 529n
.meas TRAN dfall2 v(rd) when v(rd) = 2.5 td = 529n
.meas TRAN holdtime PARAM clkfall2-dfall2
.meas TRAN afall v(clk) when v(clk) = 2.5 td = 525n
.meas TRAN yrise v(q) when v(q) = 2.5 td = 525n
.meas TRAN clk-Q PARAM yrise-afall
*note timing will need to be changed to capture measurements for layout vs.
schematic measurements.
```

.Meas Code

Schematic	Layout
* C:\Users\aroggow\Desktop\meas.mout clkfall1: v(rclk)=2.5 AT 4.02425e-008 drise1: v(rd)=2.5 AT 3.99909e-008 **setuptime: clkfall1-drise1=2.51516e-010** clkfall2: v(rclk)=2.5 AT 5.30257e-007 dfall2: v(rd)=2.5 AT 5.30105e-007 **holdtime: clkfall2-dfall2=-1.52205e-010** afall: v(clk)=2.5 AT 1.25315e-007 yrise: v(q)=2.5 AT 1.2549e-007 **clk-Q: yrise-afall=1.75011e-010**	clkfall1: v(rclk)=2.5 AT 6.54243e-008 drise1: v(rd)=2.5 AT 6.50017e-008 **setuptime: clkfall1-drise1=4.22607e-010** clkfall2: v(rclk)=2.5 AT 5.45446e-007 dfall2: v(rd)=2.5 AT 5.45069e-007 **holdtime: clkfall2-dfall2=-3.77258e-010** afall: v(clk)=2.5 AT 5.45115e-007 yrise: v(q)=2.5 AT 5.45642e-007 **clk-q: yrise-afall=5.27052e-010**

.Meas Output

Appendix B

Appendix B is intended for resources pertaining to the presentation of this report that are not necessarily relevant to an end user.

1. Figure 2 Wavedrom code

The following was used at http://wavedrom.com/editor.html to create the waveform found in Figure 2:

```
{signal: [
  ['Inputs',
   {name: 'clk', wave: 'P.........'},
   {name: 'in0', wave: 'xxxxxxxxxx'},
   {name: 'in1', wave: 'xxxxxxxxxx'},
   {name: 'sel', wave: '0...1...0.', phase: 0.5},
  ],
  {},
  ['Outputs',
   {name: 'out', wave: '=...=...=.', data: ['in0', 'in1', 'in0']},
  ],
   {}
],
  config: {hscale: 1}}
```