



Rotary Encoder

Yingcai Dong
2015/08/23

I. Introduction	2
II. Circuit design	3
Decisions based on the analysis	3
Circuit design of Delay flip flop	3
Circuit design for XOR	8
Up/Down counter circuit design	11
Direction detect Delay flip flop	11
III. Simulations	12
Counter clock wise	12
Delay flip flop	12
General readout circuit	13
Clock wise	14
Delay flip flop	14
General readout circuits	15
Characteristic analysis	16
Propagation delay	16
Total power dissipation	17
IV. Layout	20
Delay flip flop	20
Up/Down Counter	21
General readout circuits	23
Area	24
V. Observations	25
VII. Feedback	25

I. Introduction

This project is asking us to design an integrated circuit to mark the position of the wheel and then calculate the velocity of the wheel rotation. *Figure 1* shows the three-bit output readout circuits.

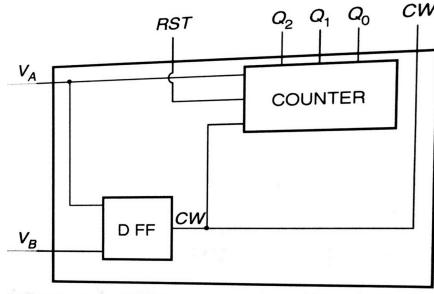


Figure 1: Block diagram of the readout circuits

Using three-bit output means it can present 3-bit binary number. In another word, it can present eight decimal number(000 refers 1 in decimal, 111 refers 7 in decimal). Thus the whole wheel can be decided into eight sectors. And the further conclusion we can get is every time the bit output change, i.e. from 101 to 110(refers to 5 to 6 in decimal) the angle of the wheel changes 45° .

But the wheel is divided into 16 sectors that mean if we use the three-bit output, the number will exceed. Because three-bit output only counts from 0 to 7, but the sector changes from 0 to 15. So three bit can't present all sectors. But it's my fault not to think these before I start to do the circuit design. So that's why in my analysis, I think the 4-bit output is better than 3-bit output, but still implement in three bit. To fix the exceed problem, my circuit is designed to receive every two impulses to count once, so if there are 16 sectors, it will count eight times and each time represent 45° .

In section II, I will discuss how to choose the size of the transistors, how to schedule the hierachic of each block. And the trade-offs that I made.

In section III, I will do the simulations of each sub-blocks and also for the whole block. Also, explain how I adjust the input signal to simulate the requirement. After that, I will calculate the delay and the power consumption of the circuit.

The circuit layout will be introduced in section IV. Also, I will show you the size that I achieved.

In the last three section, I will report the problem I had during this project, and any forward improvement I may able to achieve. And some of the feed backs for this course.

II. Circuit design

Decisions based on the analysis

The circuit design will be divided into two big blocks: Up/Down counter and Delay flip-flop. I will first discuss the Delay flip-flop because it is also an essential part of Up/Down counter.

Circuit design of Delay flip flop

At the very beginning of the design, I should first make a decision on what kind of Delay flip-flop I should use. There are three major types of Delay flip-flop: The classic Delay flip-flop which doesn't have either set (when the set is logic 1, the output is 1 regardless of the input) or reset (when reset is logic 1, the output is 0 regardless of the input). Delay flip-flop with reset or set function. And the Delay flip-flop has both set and reset function.

It is clearly indicated in the project's demand, and there should have a rest input for the Up/Down counter. So instead of using some additional logic gate to achieve the function outside the Delay flip-flop, it's better to build in the Delay flip-flop.

The schematic layout of the Delay flip-flop with reset function is shown in *Figure 2*

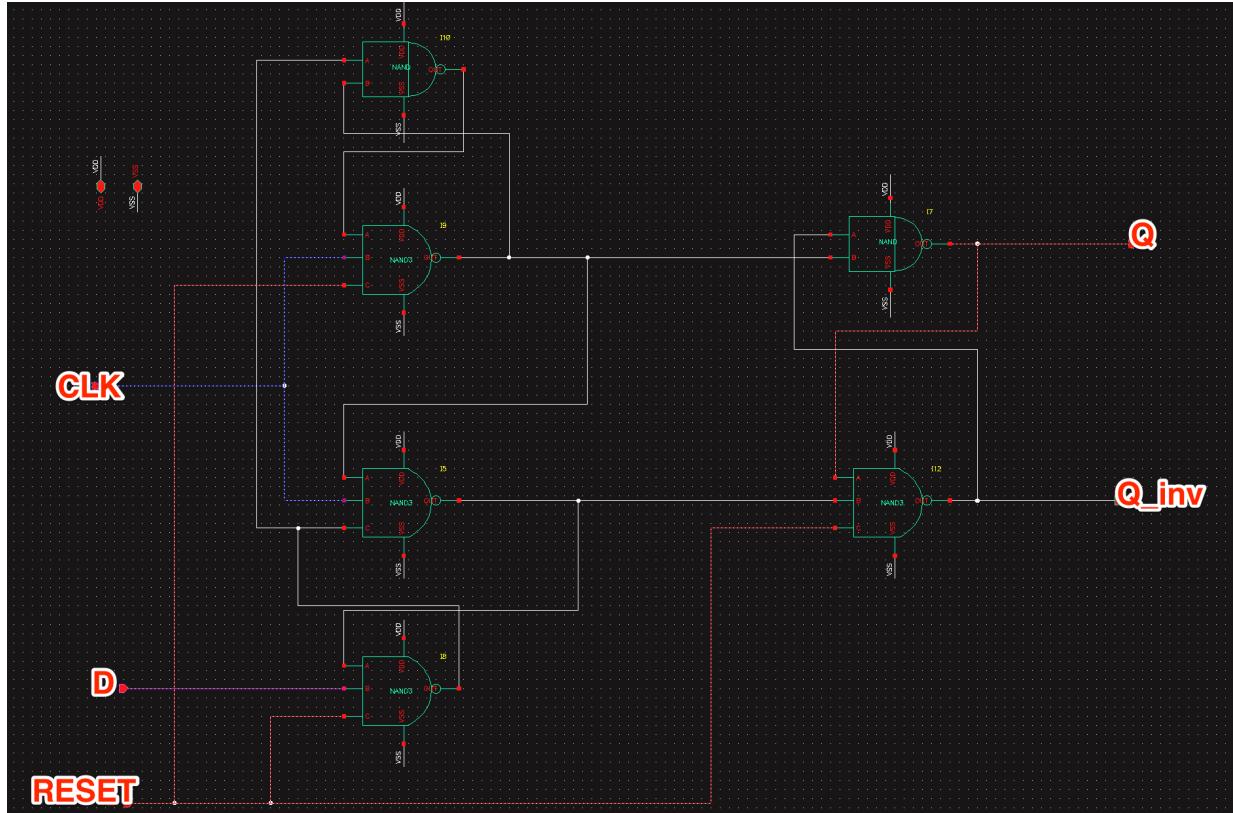


Figure 2: Delay flip flop with reset function

Hand calculations

As you can see in *Figure 2*, the Delay flip-flop is constructed by 2 NAND2 and 4 NAND3. In my design, I just designed the schematic of NAND2 and NAND3, and generate the symbol for each gate, and then connect them with wires to make the Delay flip-flop.

- NAND2 schematic

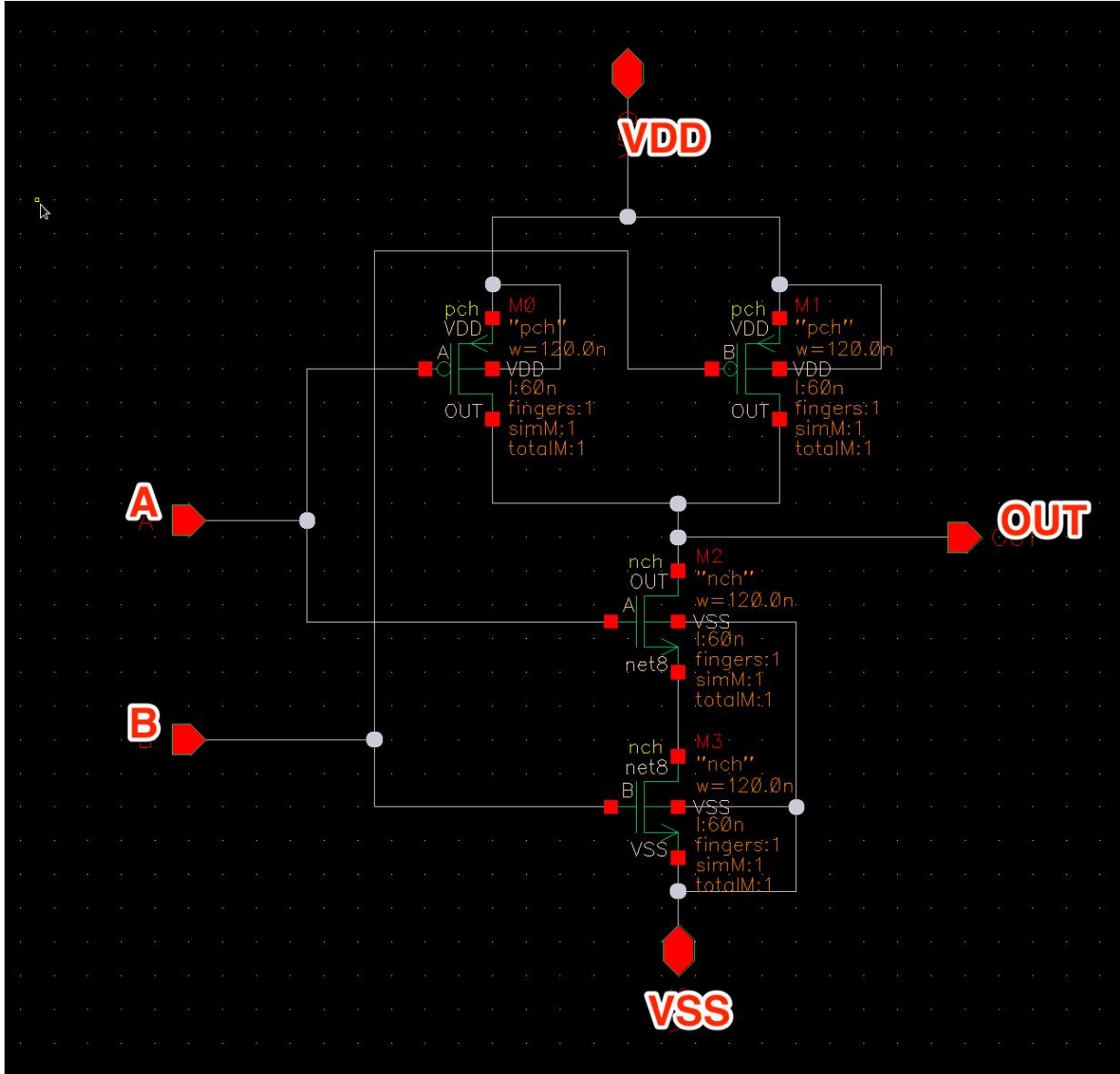


Figure 3: NAND2 schematic

The trade off in design this NAND2 gate is to achieve the minimum delay. To do that, I keep the resistance ratio of pMOS and nMOS in serial equals to 1. *Figure 4* shows how to calculate the ratio.

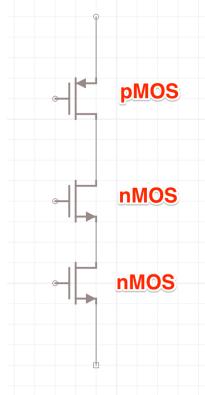


Figure 4: Demo of how to calculate the ratio of the size of the NAND2 gate

So as you can see:

$$\text{Resistance for pMOS: } \frac{2R}{h}$$

$$\text{Resistance for nMOS: } \frac{R}{k} + \frac{R}{k}$$

where h stands for the width of the pMOS, and k stands for the width of the nMOS. As for the length of both transistor, to achieve the minimum delay while achieving the small size, I just keep the length as small as possible which is 60 nm.

As mentioned previous, the ratio of resistance should be 1 thus the ratio between h and k should be:

$$\frac{h}{k} = 1 \text{ which means the width of pMOS and nMOS should be the same. And the unit width I choose 120 nm because this is the minimum width can be used in the transistors. So I pick the width of pMOS is 120 nm, and the width of nMOS is 120 nm.}$$

- NAND3 schematic

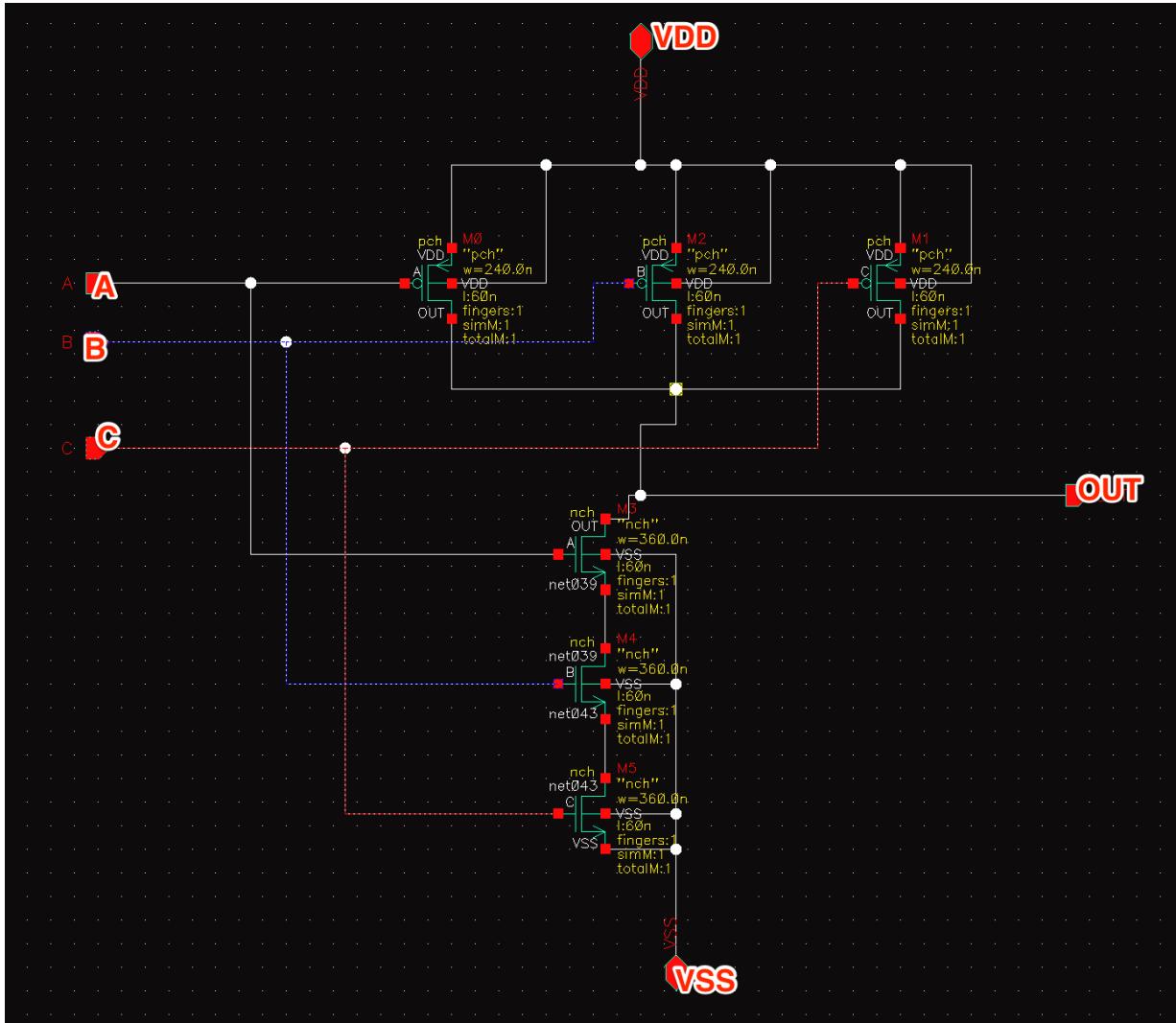


Figure 5: NAND3 schematic

The trade off in design the NAND3 gate is same to the NAND2. In *Figure 6* you will see the demonstration of how to calculate the ratio

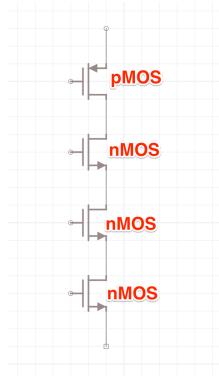


Figure 6: Demo of how to calculate the ratio of the size of the NAND3 gate

So as you can see:

$$\text{Resistance for pMOS: } \frac{2R}{h}$$

$$\text{Resistance for nMOS: } \frac{R}{k} + \frac{R}{k} + \frac{R}{k}$$

where h stands for the width of the pMOS, and k stands for the width of the nMOS. And the keep the minimum size, the length of both pMOS and nMOS is set to 60 nm.

And the ratio between h and k should be: $\frac{h}{k} = \frac{2}{3}$. So the width of pMOS is 2 times bigger than the unit width, which is 240 nm. The nMOS is 3 times bigger than the unit width, so it is 360 nm.

After the design of these two gates, I connected them based on *Figure 2* instruction.

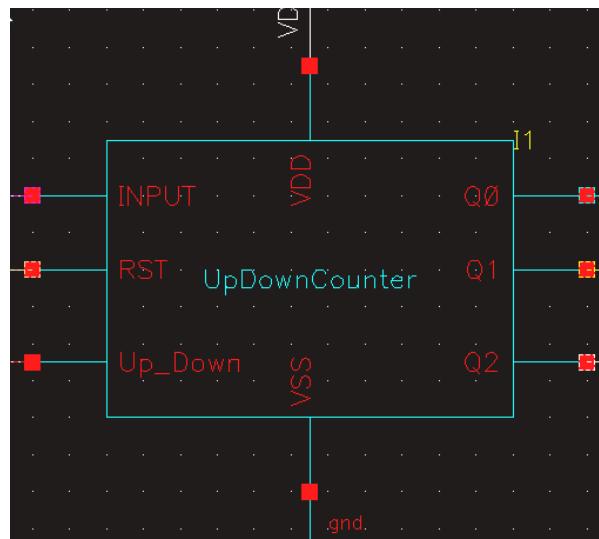


Figure 7: Block diagram for Delay flip flop

While designing this block, I was also thinking a question: Delay in Multistage logic networks. In this block, it is a multistage network; I should gain more delay decrease if I use the theory there. But I also face three biggest problems here: 1. In the textbook, it show us how to optimal the delay in a single path. But in this block for example, which path show I choose to optimal? In other words, is there a critical path? Or I should consider all the situations? 2. In the textbook, the examples show the path are all forward, but in flip flop, we can't avoid backward path, then who are we going to calculate that? 3. How to get the load at the output?

So I simply use the same size gate in within the block. But I'm sure that if I solve the problem using multistage logic networks, the delay should decrease significantly.

Circuit design for XOR

The XOR gate is actually made of 2 inverter and a XOR with 4 input(A, A_{_inv}, B, B_{_inv}). And I designed them separately.

- XOR schematic

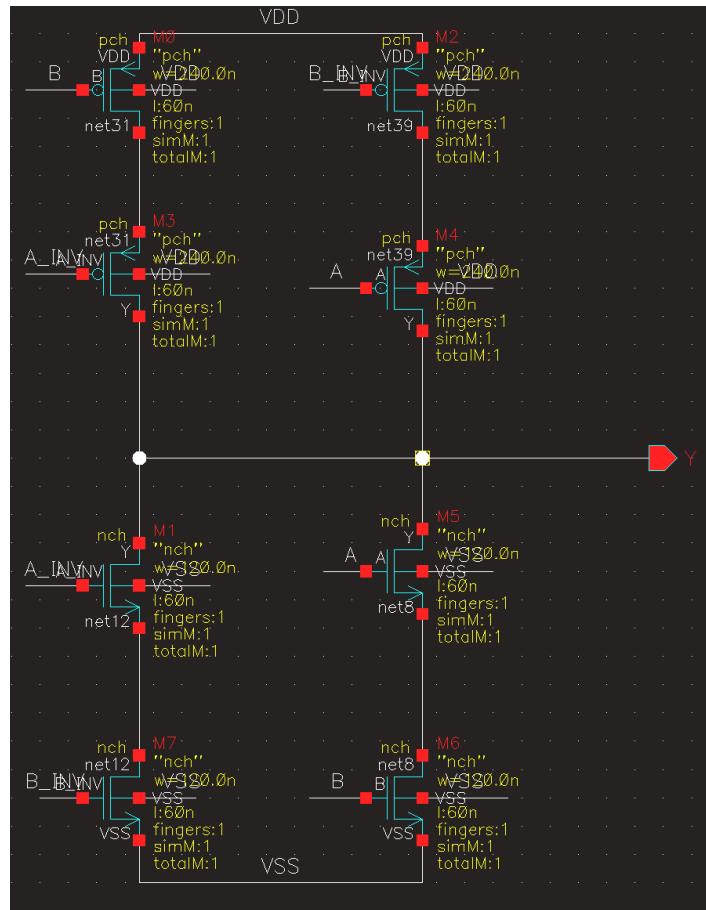


Figure 8: XOR schematic

Using the same method to get the width ratio for XOR:

$$\text{Resistance for pMOS: } \frac{2R}{h} + \frac{2R}{h}$$

$$\text{Resistance for nMOS: } \frac{R}{k} + \frac{R}{k}$$

And the ratio between h and k should be: $\frac{h}{k} = \frac{2}{1}$. So the width of pMOS is 2 times bigger than the unit width, which is 240 nm. And the nMOS is same big as unit width 120 nm.

And the block diagram of XOR is shown below.

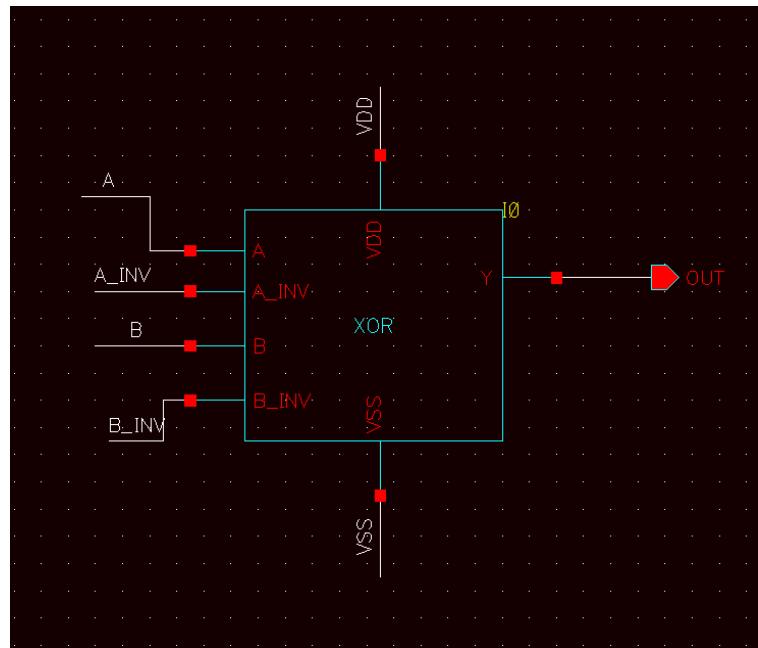


Figure 9: XOR block diagram

- Inverter schematic

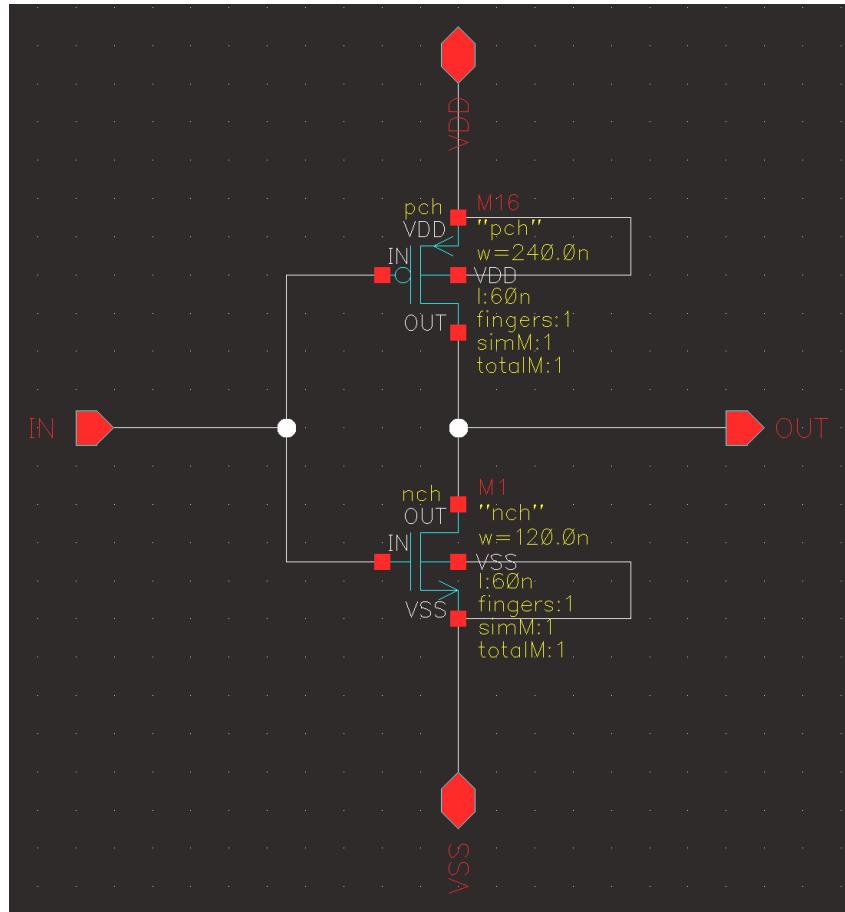


Figure 10: Inverter schematic

The ratio for the inver I choose is 2:1. Which means the pMOS width is 240 nm and width for nMOS is 120 nm.

And the block diagram for the complete XOR gate is shown below

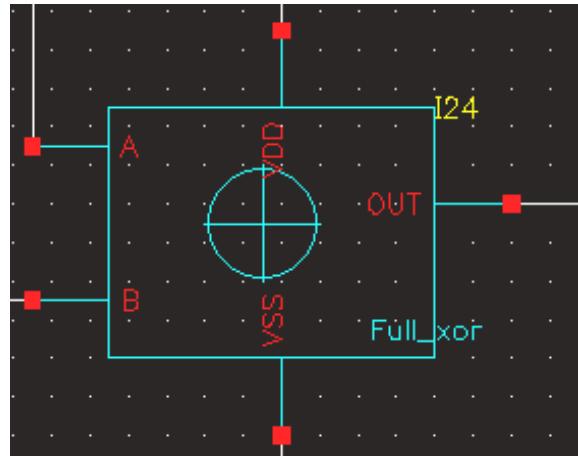


Figure 11: Block diagram for complete XOR gate

Up/Down counter circuit design

The Up/Down counter schematic is show below

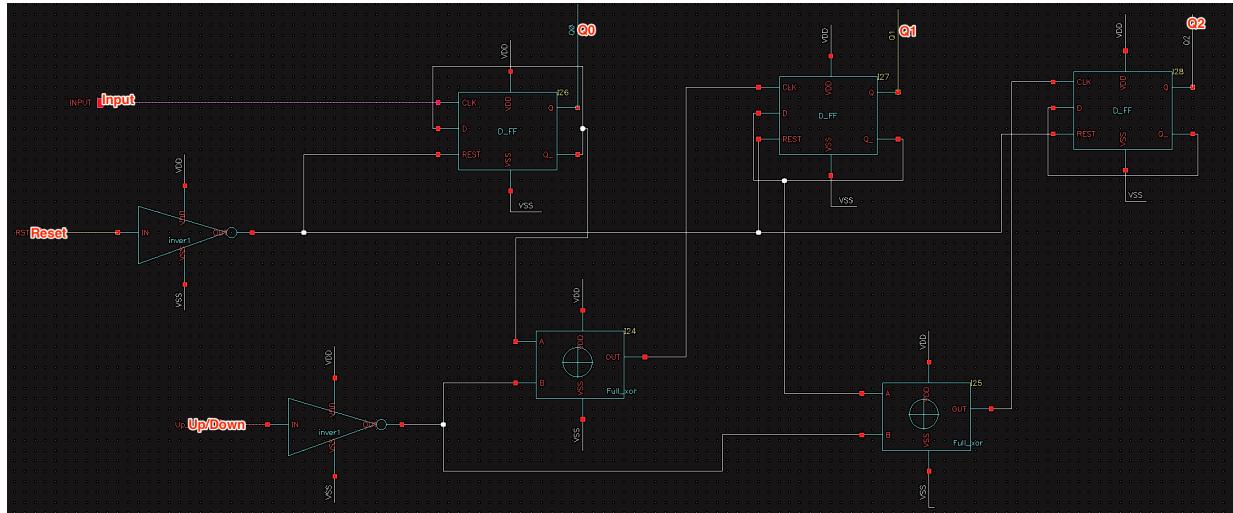


Figure 12: Up/Down counter schematic

Figure 12 is a three bit counter, if we want to make more bits, we can add more Delay flip flop and XOR to it.

Direction detect Delay flip flop

The Delay flip-flop used to detect directions has the same inner connections compared to the Delay flip-flop used in the Up/Down counter, the only difference is the REST pin is connected to the Q. And signal B is connected to CLK pin, signal A is connected to D pin. The output is Q pin.

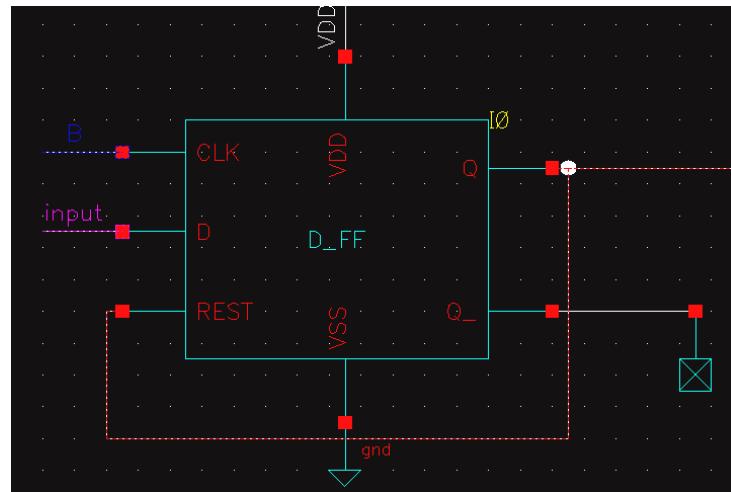


Figure 13: Block diagram of direction detection Delay flip flop

III. Simulations

The input signal I use

Voltage 1	Voltage 2	Rise time	Fall time	Pulse width	Period
0 V	1 V	40n s	40n s	4u s	8.08u s

Since the distance between A and B is half sector, which means the delay between them is quarter the period 2.02u s.

Counter clock wise

Clock wise means sensor B will first receive the impulse and then A will receive the impulse.

Delay flip flop

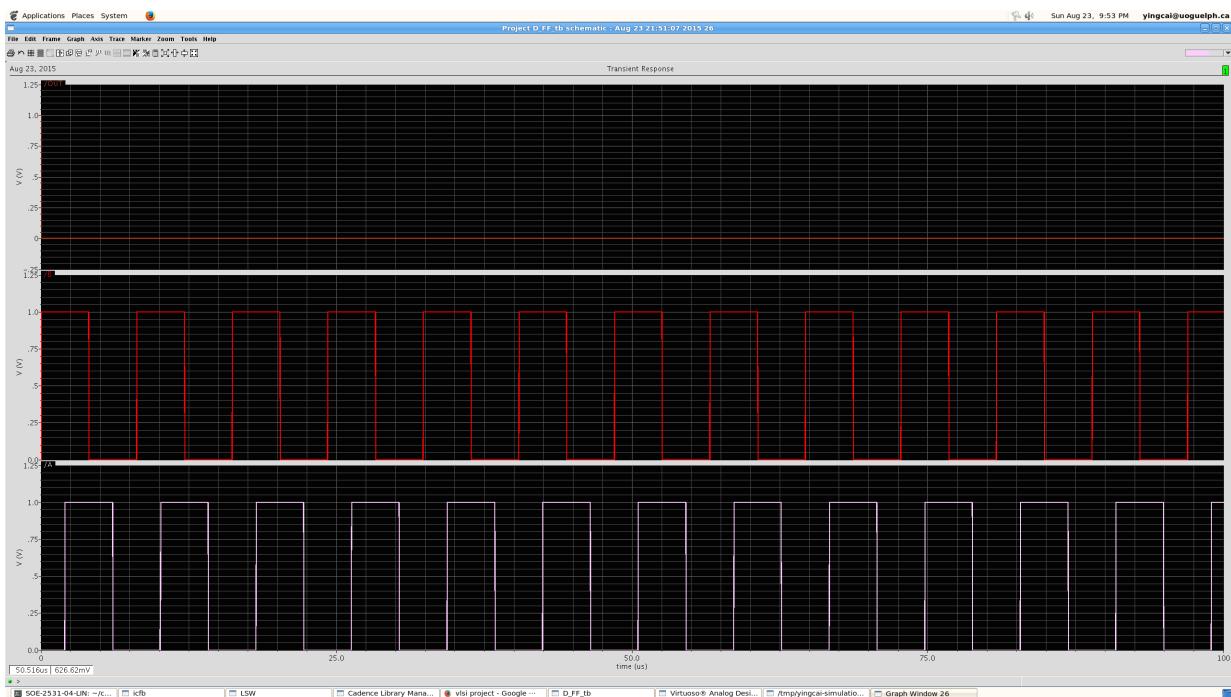


Figure 14: Counter clock wise Direction signal simulation

As you can see in *Figure 14* the direction signal is 0 when signal B is ahead of signal A. And it general at the rising edge of signal B.

General readout circuit

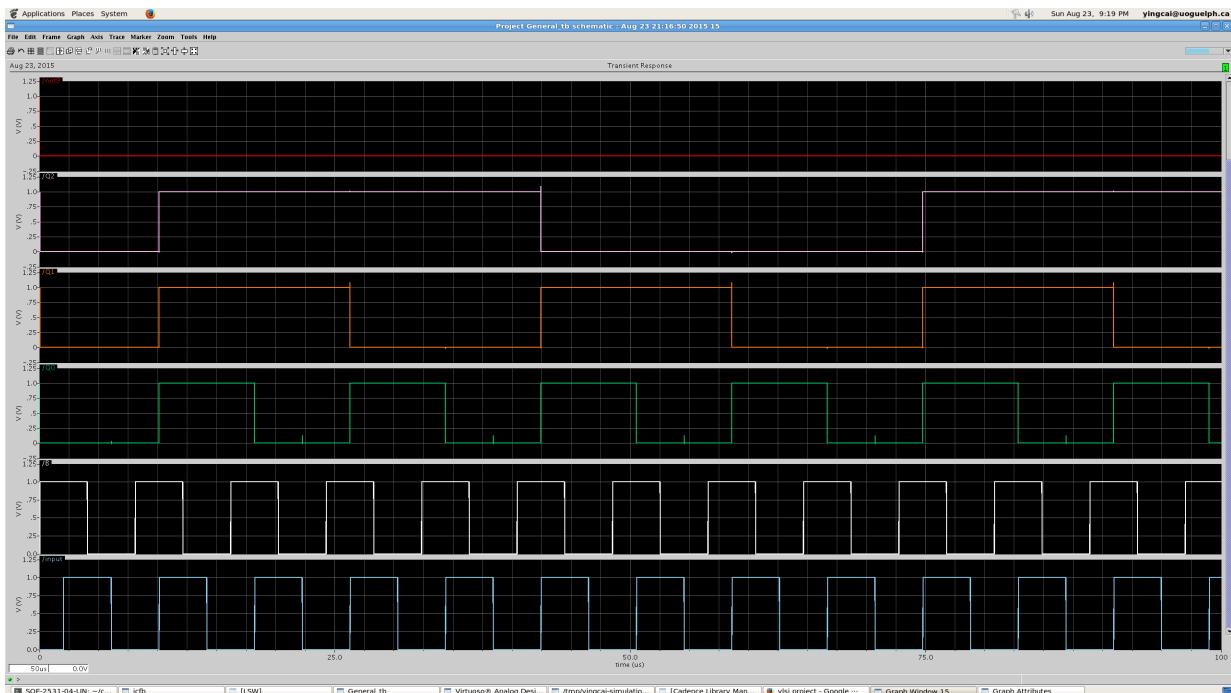


Figure 15: Dynamic simulation

In this picture, because I didn't realize that I should use four bits output, so when I designed the circuit, I use every two input count once, to not exceed for the wheel to rotate a full circle(As I explained in Section I).

Clock wise

Counter clock wise means the sensor A will receive the impulse first and then B receive the impulse.

Delay flip flop

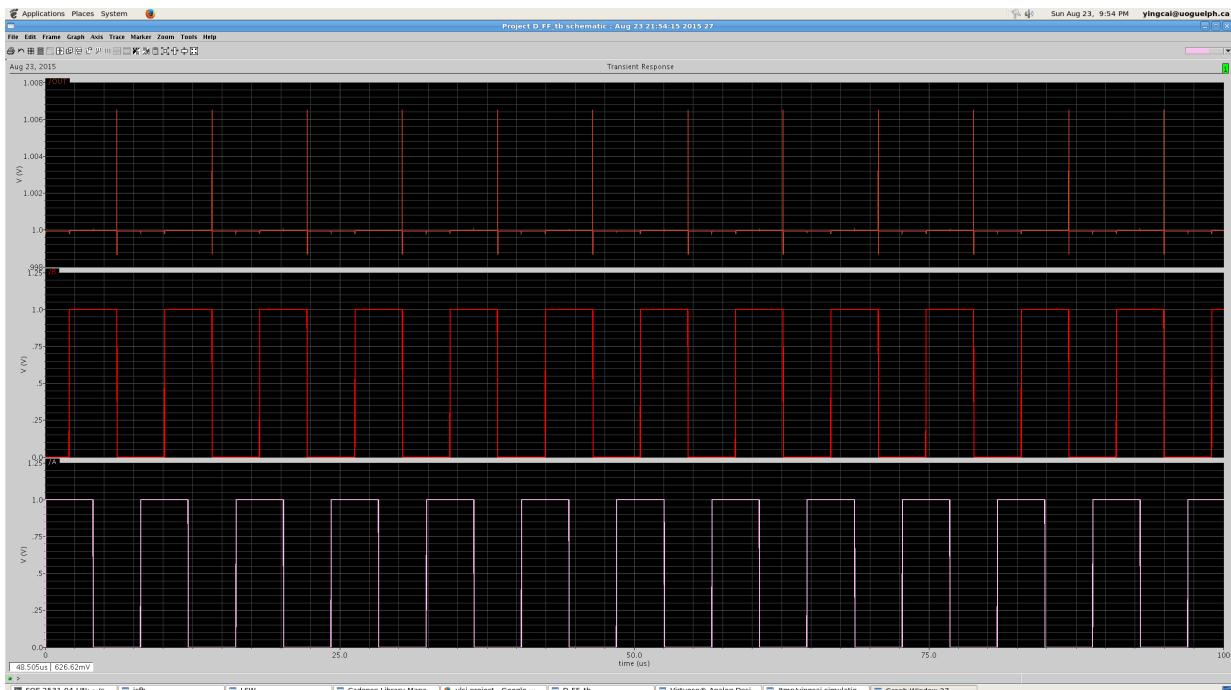


Figure 16: Clock wise Direction signal

As you can see in the figure, the output signal is 1 when signal A is ahead of signal B. The sharp tip you can observe in the figure, is only 0.0064 V.

General readout circuits

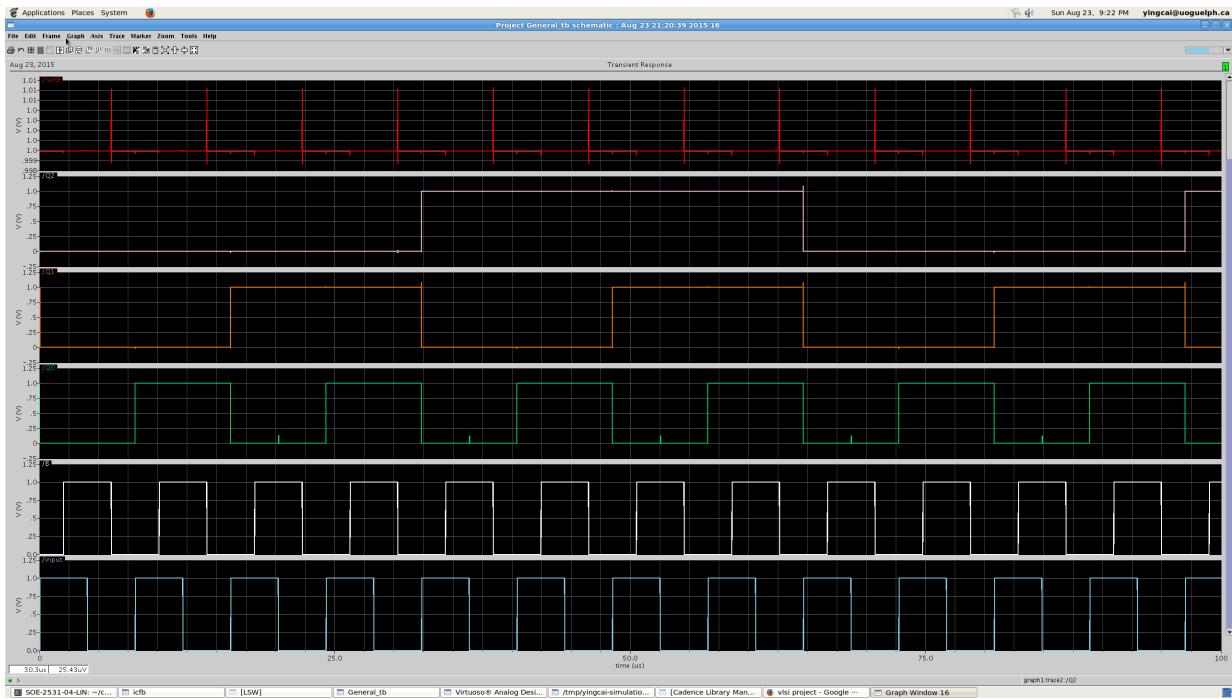


Figure 17: Clock wise dynamic simulation

Characteristic analysis

Propagation delay

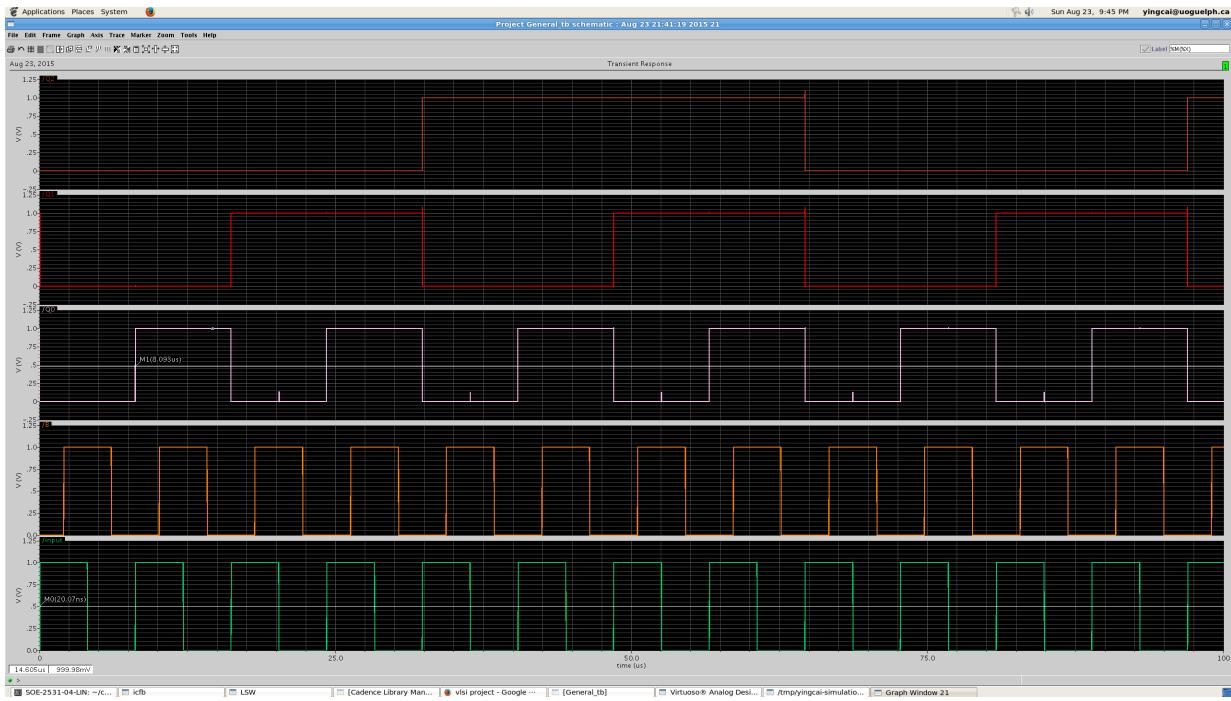


Figure 18: Clock wise delay

As shown in the *Figure 18*, the delay is $8.093\mu s - 0.02007\mu s = 8.07293\mu s$

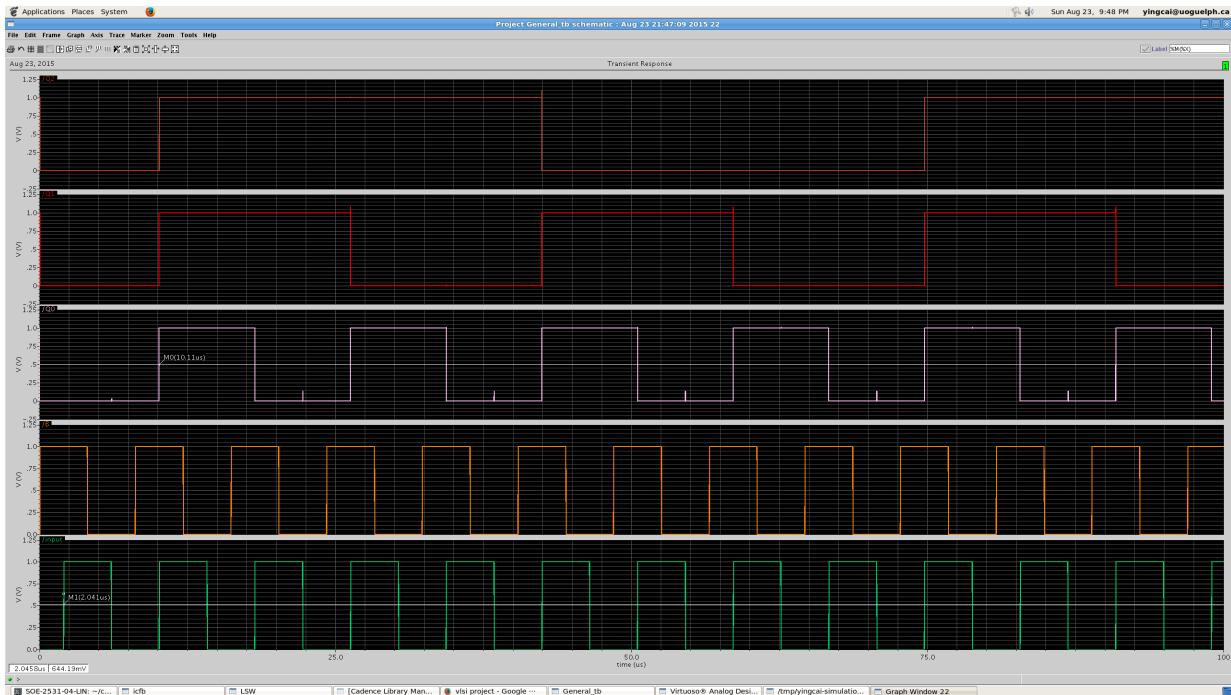


Figure 19: Counter clock wise delay

The delay of Counter clock wise delay is $10.11\mu s - 2.041\mu s = 8.069\mu s$

Total power dissipation

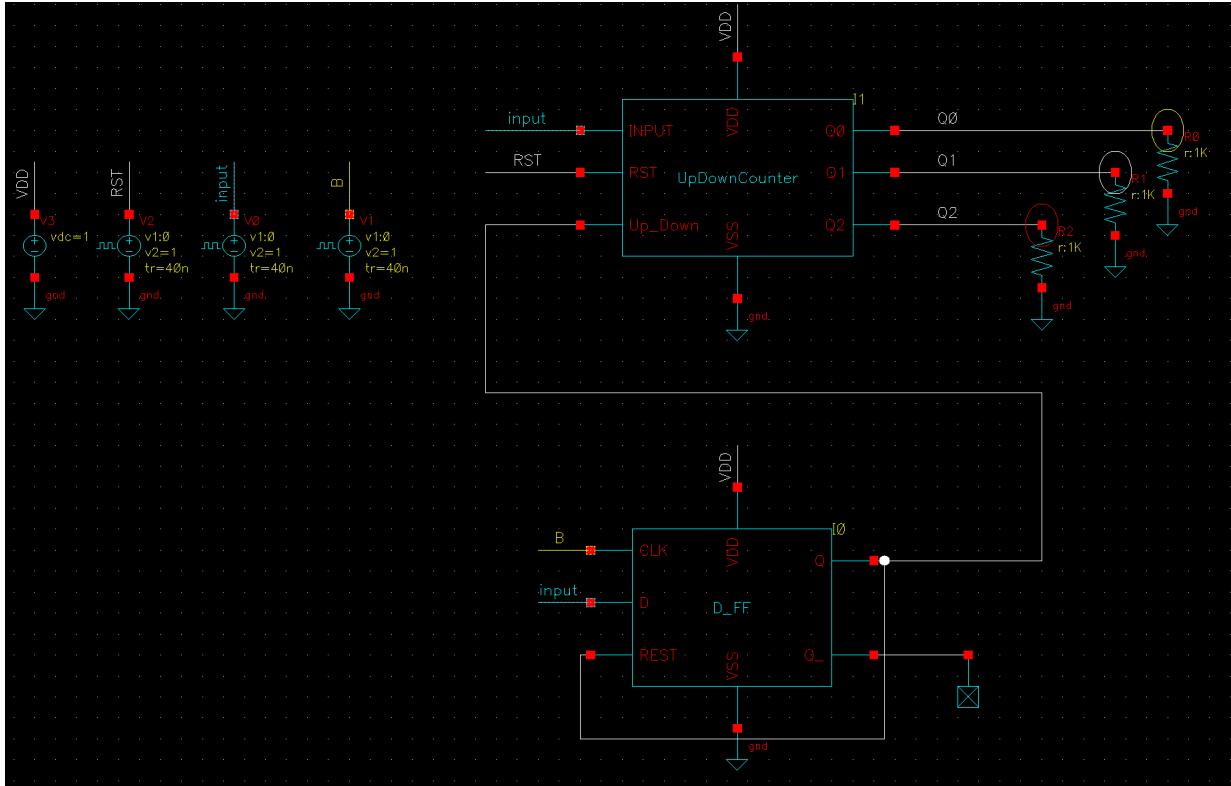


Figure 20: TestBench for power dissipation

Instead of put NO CONNECTION at the end of Q0, Q1and Q2, I attached three 1k ohm resistance. Because to calculate the power, the most efficient way is to use equation $P = UI$. The voltage we knew from the source, so if we know the current, then we can calculate the power. In the figure below, I showed the output of the current.

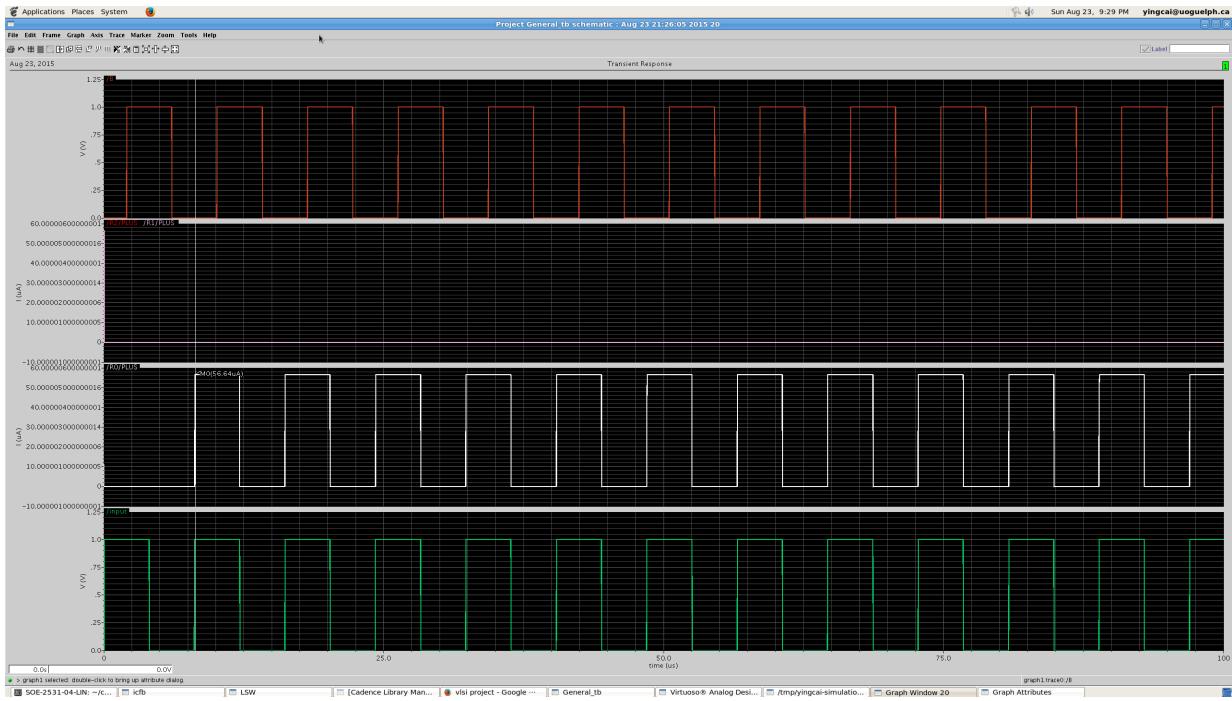


Figure 21: Clock wise power dissipation

So in *Figure 21* I only get the current from Q0. And as you can see in the figure, it's current follows the same pattern as the input signal A. The maximum current I observed is 56.64u A. Thus, it's maximum power dissipation is $P = UI = 1V \times 56.64 \times 10^{-6} A = 56.64 \times 10^{-6} W$

As for the Counter clock wise, the output is shown in the figure blow

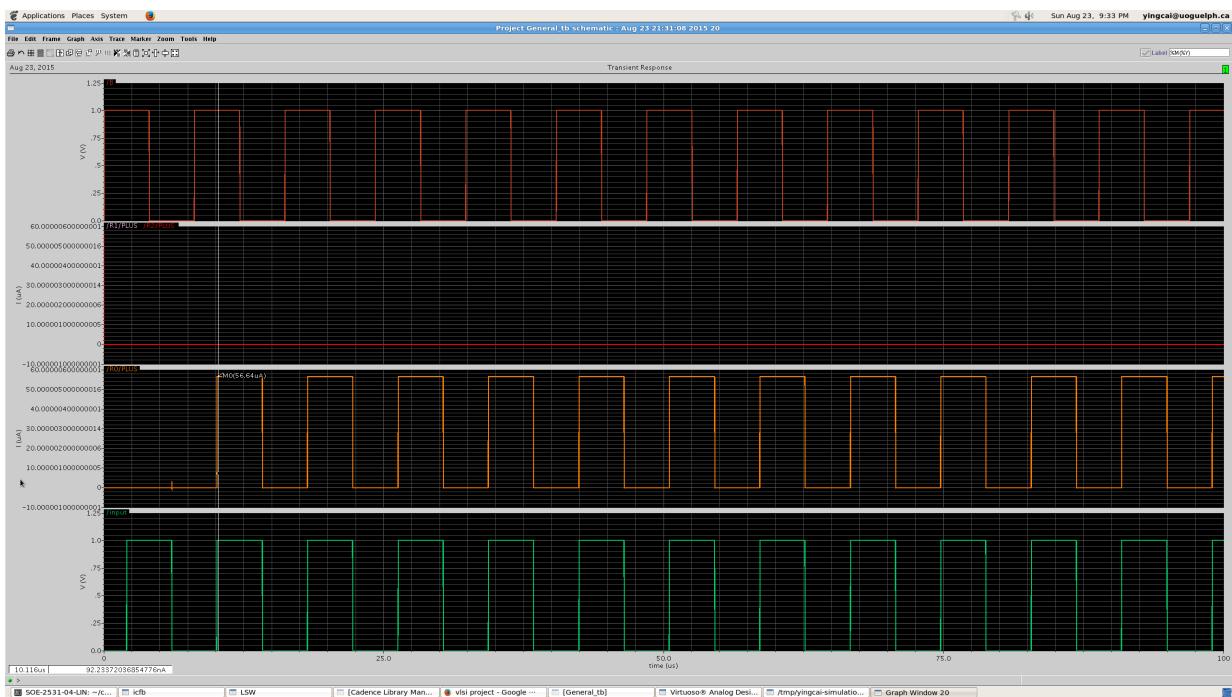


Figure 22: Counter clock wise power dissipation

So in counter clock wise current output I observe the maximum current is $56.64\mu A$, same with the clock wise situation. Thus the maximum power dissipation of counter clock wise is $56.64 \times 10^{-6} W$.

IV. Layout

Delay flip flop

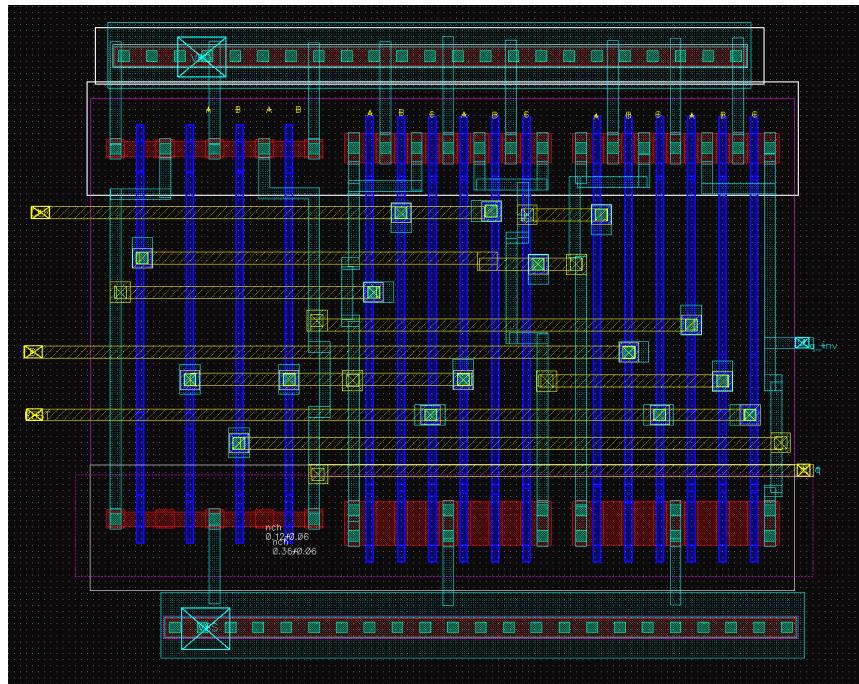


Figure 23: Delay flip flop layout

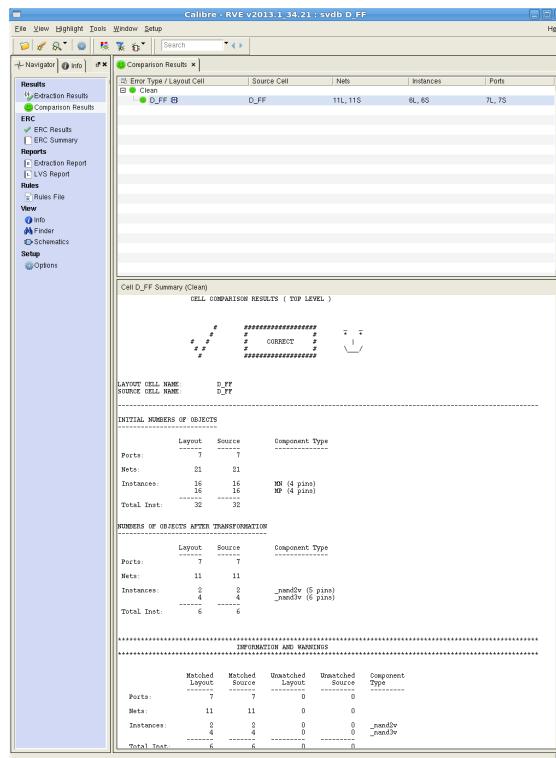


Figure 24: LVS verification

Up/Down Counter

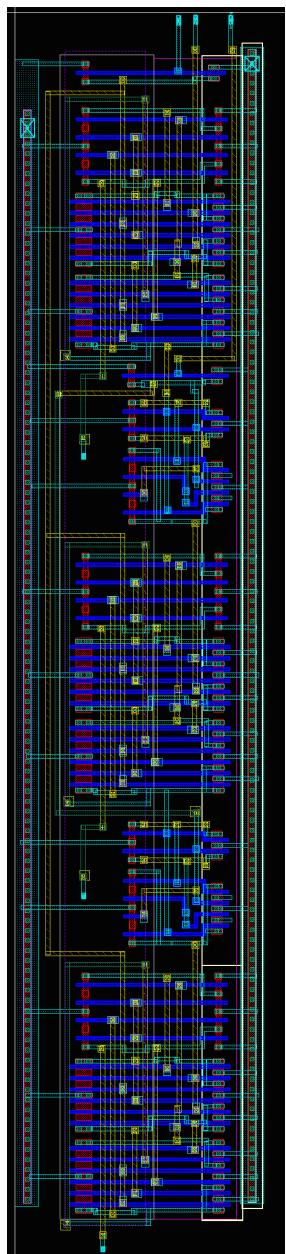


Figure 25: Up/Down counter layout

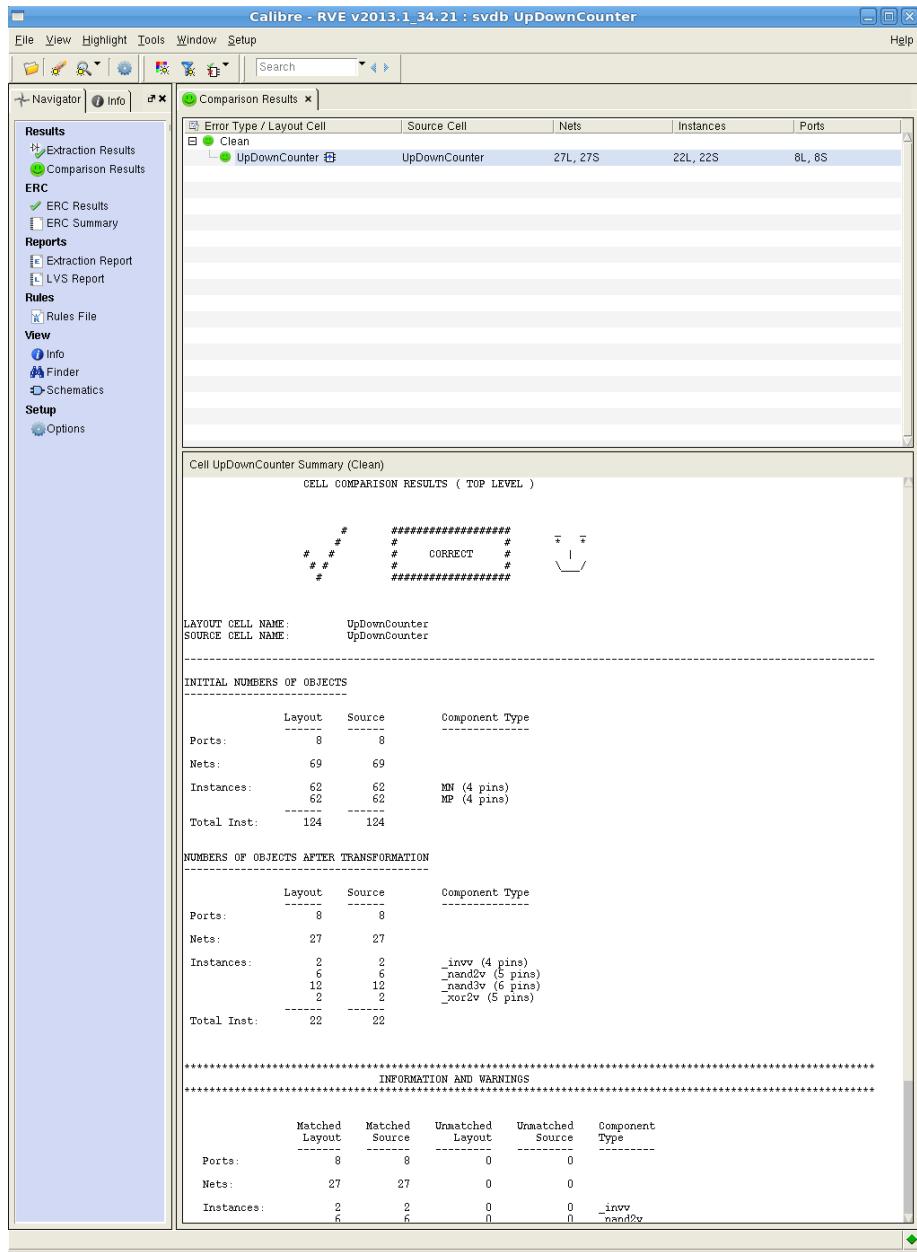


Figure 26: Up/Down counter LVS verification

General readout circuits

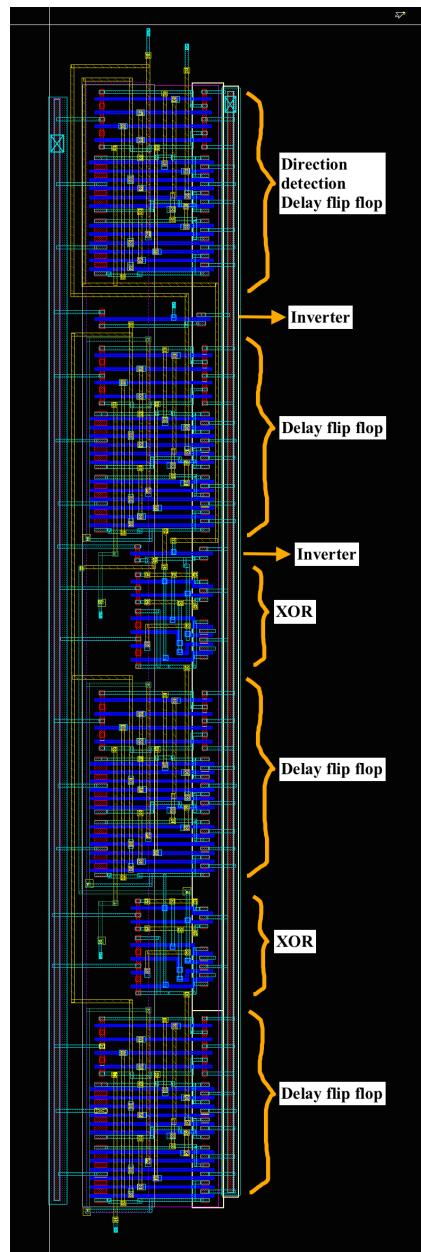


Figure 27: General readout circuit layout

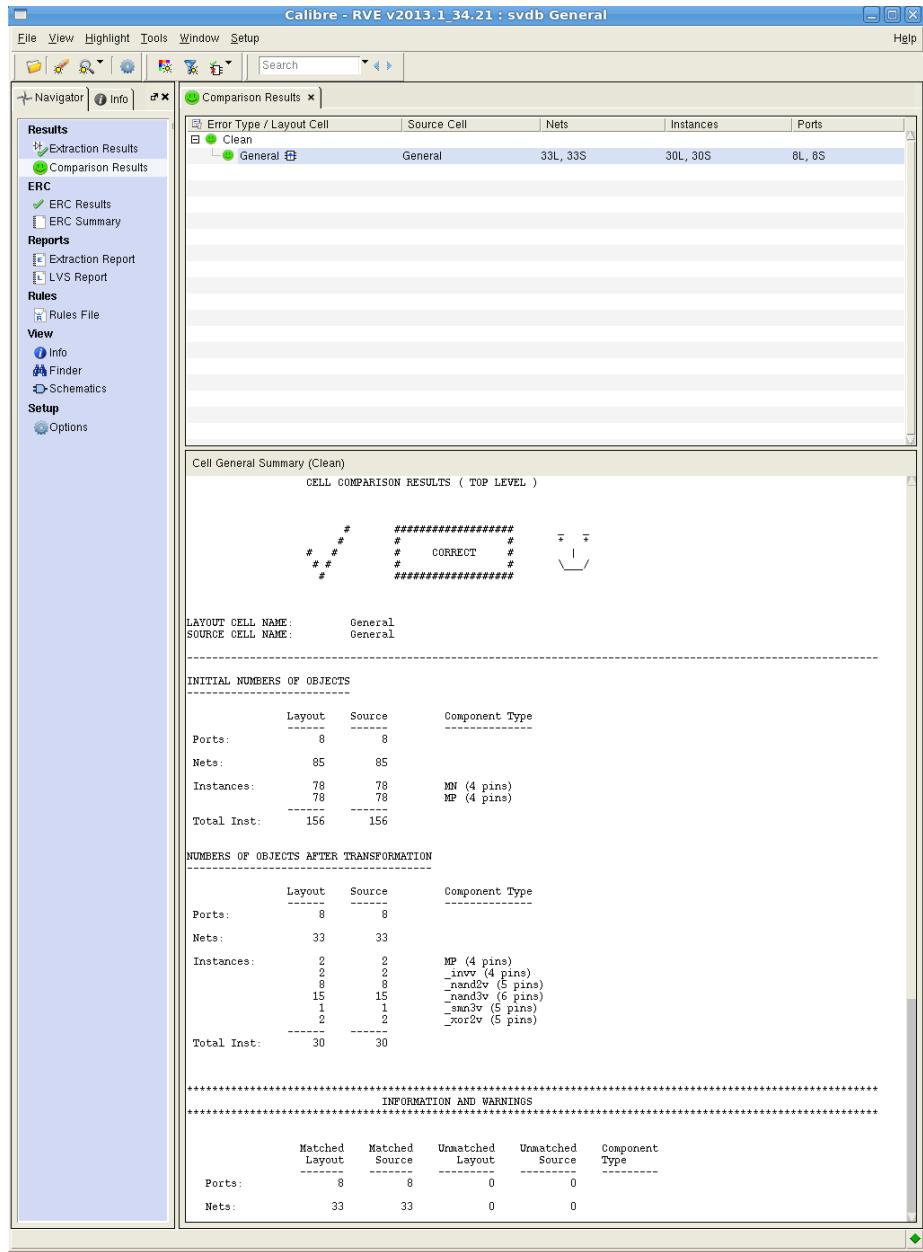


Figure 28: General readout circuit LVS verification

Area

And the area is $33.55\mu m \times 5.725\mu m = 191.906\mu m^2$

V. Observations

Through this project, I get familiar with the Cadence tools. And know how to draw the layout and how to use the DRC to check whether the rules are satisfied in the design, and use the LVS to compare the logic with the schematic. Until the lab 2, I still unable to pass the LVS verification. But in this project, I spend a good time on drawing the layout and correcting errors. These are all very useful practice.

For this project, there are still plenty of space to improve. For example, I can make 4 bits output compatible with 16 sectors. What's more, I wasn't able to make a circuit to calculate the velocity. Maybe in the future, I will realize it.

VII. Feedback

The instructions in the lab are clear enough and very practical. The knowledge I learned in class is useful, but when dealing with some complicated situations, I don't know how to use the knowledge properly. For example, the delay in Multistage logic networks. We practiced a lot in class, and I can solve some simple questions, but when I am facing the real problems, for example, the flip-flop, I don't know how to use that knowledge.