



Penn
Engineering
UNIVERSITY *of* PENNSYLVANIA

ESE5700 Project 2 Report

Design and Optimization of a CLB with SRAM-LUT

**DEPARTMENT OF ELECTRICAL AND SYSTEM
ENGINEERING**

UNIVERSITY OF PENNSYLVANIA

Table of Contents

I.	Design Schematics	4
1.	Overall CLB structure	4
1.1	CLB Overall schematic	4
1.2	CLB Test Schematic and Performance Verification.....	5
2.	Serial In Parallel Out (SIPO) Design.....	7
2.1	D Flip Flop	7
2.2	SIPO (Serial in Parallel Out shift register)	8
3.	6T SRAM Array Design	9
3.1	6T SRAM Cell	9
3.2	SRAM Array Schematic	13
4.	LUT Design	15
4.1	The 2:1 iMUX Schematic	15
4.2	The 16:1 LUT Schematic	16
5.	Clock Frequency Divider	17
II.	Design Description.....	19
1.	Overall Structure Description.....	19
2.	SRAM Cell Design Consideration.....	19
3.	Pre-charge Signal in SRAM.....	19
4.	Timing and Control Logic.....	21
III.	Design Validation/Verification	23
1.	LUT Validation.....	23
1.1	2:1 iMUX Validation	23
1.2	16:1 LUT Validation	24
2.	D Flip Flop Validation	28
3.	Serial In Parallel Out (SIPO) Shift Register Validation	30
4.	6T-SRAM Validation	31
5.	SRAM Array Validation (SIPO in to SRAM Array out).....	34
6.	Clock Frequency Divider Validation	35
IV.	Design Metric Test Cases.....	38
1.	Maximum operating frequency	38
2.	Average Energy	38

3. Area	38
4. FOM	39
V. Design Metrics	40
1. Maximum Switching Frequency	40
2. Average Energy	41
3. Area	42
4. FOM	42
5. Summary table	43
VI. Design Process Writeup	44
1. SRAM Optimization	44
1.1 Stability	44
1.2 Delay/switching speed	44
2. LUT Optimization.....	45
3. Pre-charge signal generation circuit	46
4. Double flipflop to regulate output timing (perfectly one-bit shift).....	46

I. Design Schematics

1. Overall CLB structure

The designed Configurable Logic Block (CLB) is composed of five key components:

1. **SIPO Shift Register:** Converts a serial data input into a parallel output.
2. **SRAM Array:** Stores data using 16 individual 6T SRAM cells.
3. **16:1 LUT:** Selects and outputs a single bit from the SRAM cells based on a 4-bit selection line.
4. **D Flip-Flop:** Temporarily holds the output of the LUT to ensure stable signal timing.

The design includes four input pins:

- **CLK:** Drives the timing for SIPO, SRAM, and the D Flip-Flop.
- **LOAD:** Signals whether the CLB is in write(1) or read(0) mode.
- **DATA:** Feeds serial data into the SIPO.
- **LUT Selection Line:** A 4-bit input that determines the specific SRAM cell to be read.

The final output, **Out**, provides the result of the selected SRAM data after processing.

1.1 CLB Overall schematic

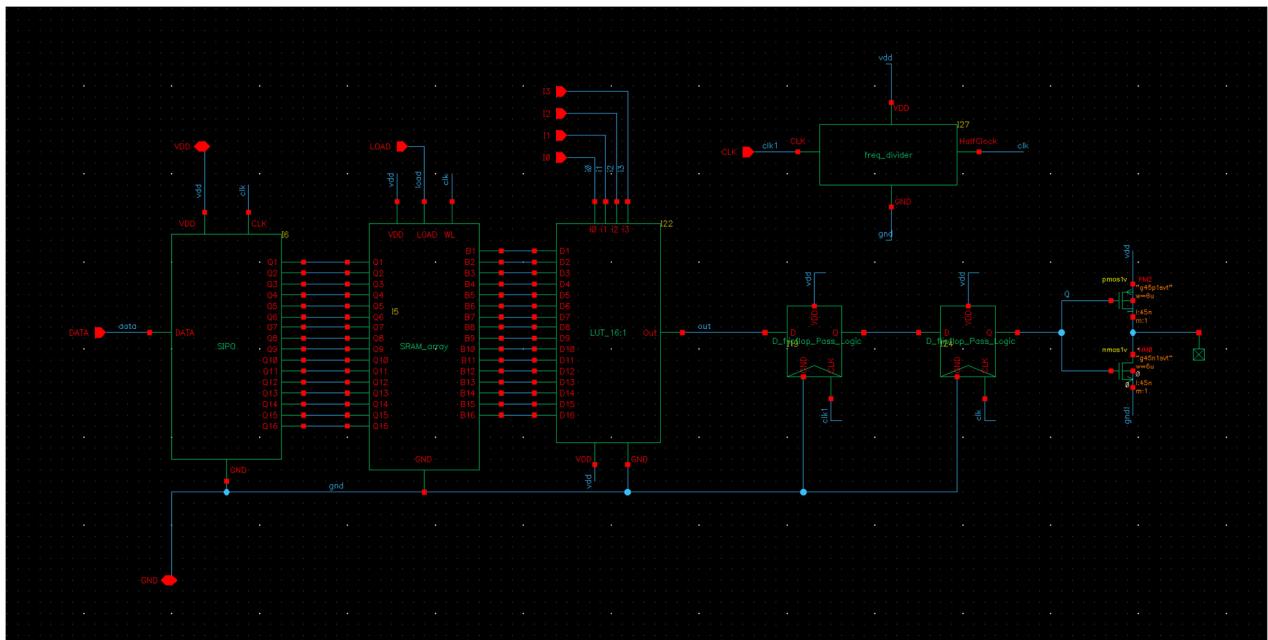


Figure 1. Configuration Logic Block Schematic

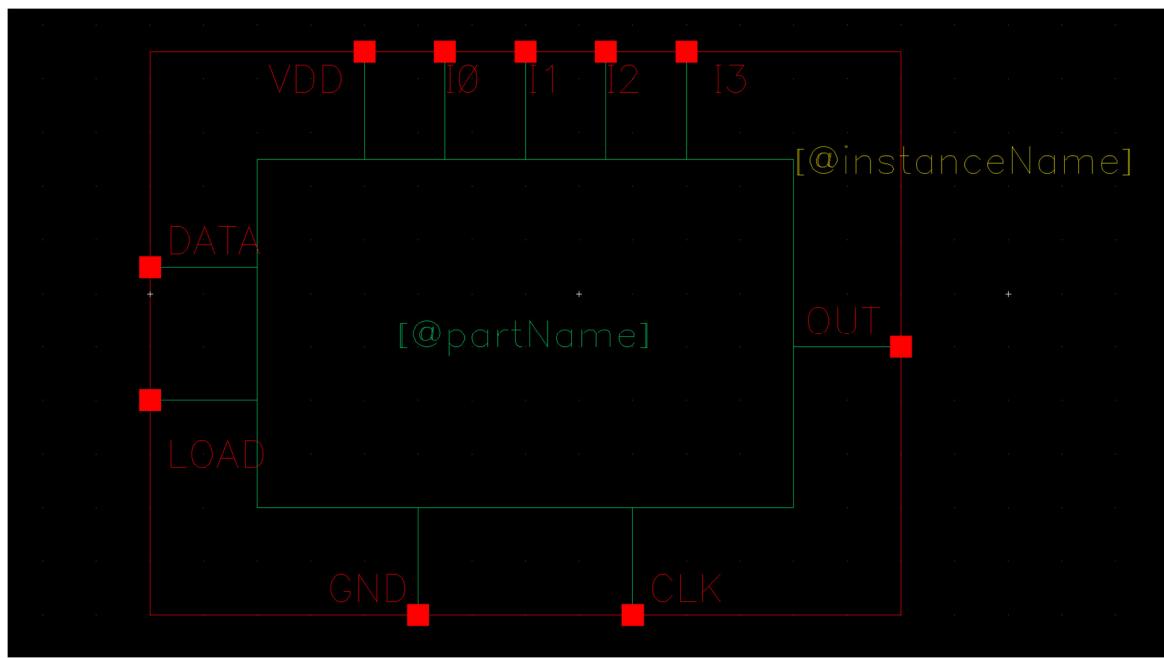


Figure2. Configuration Logic Block Symbol

1.2 CLB Test Schematic and Performance Verification

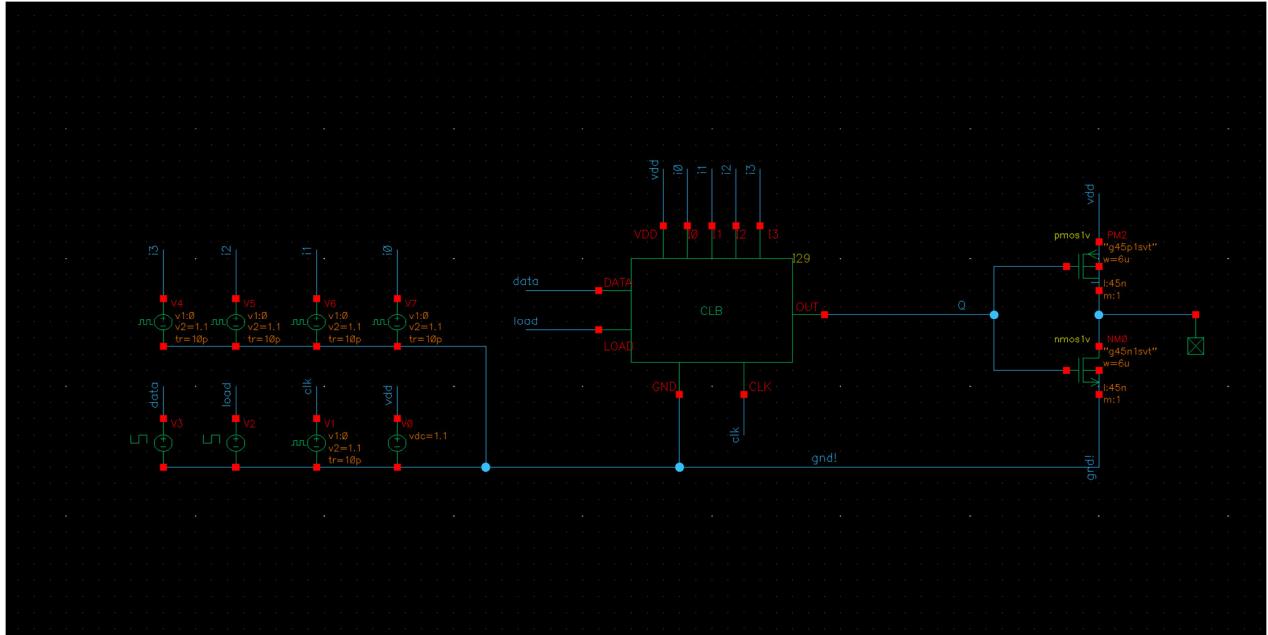


Figure3. CLB Test Schematic

Test Case Setup:

To validate the correctness of the CLB, the following test schematic and signal configuration are used:

- **Clock Signal (CLK):** The clk signal is generated using a vpulse source with a period of 10ns (tclk=10ns for validate functionality).
- **Data Input (DATA):** The data input is set to a non-repetitive pattern to ensure proper CLB functionality. The selected bitstream is 1010 1110 0010 1100, generated by a vbit source with a period of 20 ns.
- **Load Signal (LOAD):** The load signal is held high (LOAD = 1) for the first 16 clock cycles to enable data loading into the CLB and low for the next 16 clock cycles to enable read operation.
- **LUT Select Signals:** The LUT selection lines are driven by vpulse voltage sources:
 - LUT3 has the shortest period of 20ns (2tclk).
 - LUT2, LUT1, and LUT0 have periods that double sequentially, ensuring all address combinations from 0000 to 1111 are cycled through.

Expected Behavior:

1. Data Loading into SRAM

- On the 16th rising edge of the clock, the serial data input completes its journey through the SIPO shift register. The data bitstream (1010 1110 0010 1100) is then loaded into the SRAM.
- The half-cycle following the 16th rising edge is used to write the data into the SRAM, ensuring the stored values are stable for subsequent operations.

2. Output Data Behavior

- Starting from the 17th rising edge of the clock, the CLB output sequentially mirrors the input data bitstream (1010 1110 0010 1100) stored in the SRAM.
- The LUT select signals cycle through all address combinations (0000 to 1111), ensuring each bit of the stored data is correctly accessed and output.

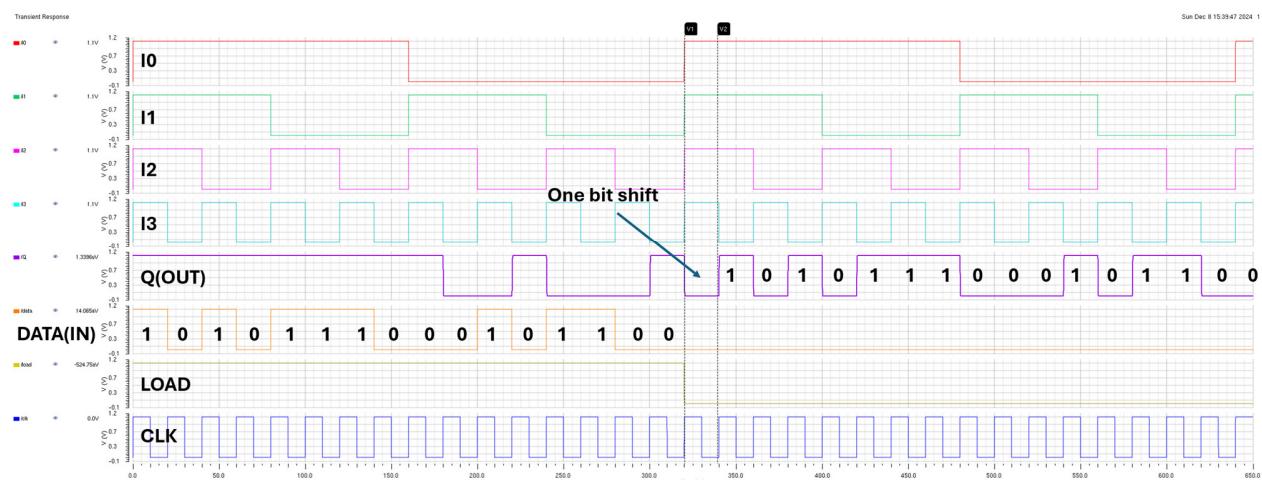


Figure 4. CLB Performance Verification Result

This test case verifies that the designed CLB performs as expected under typical operation: data is correctly loaded into the SRAM during the load phase and sequentially accessed based on the LUT select signals during the output phase.

2. Serial In Parallel Out (SIPO) Design

2.1 D Flip Flop

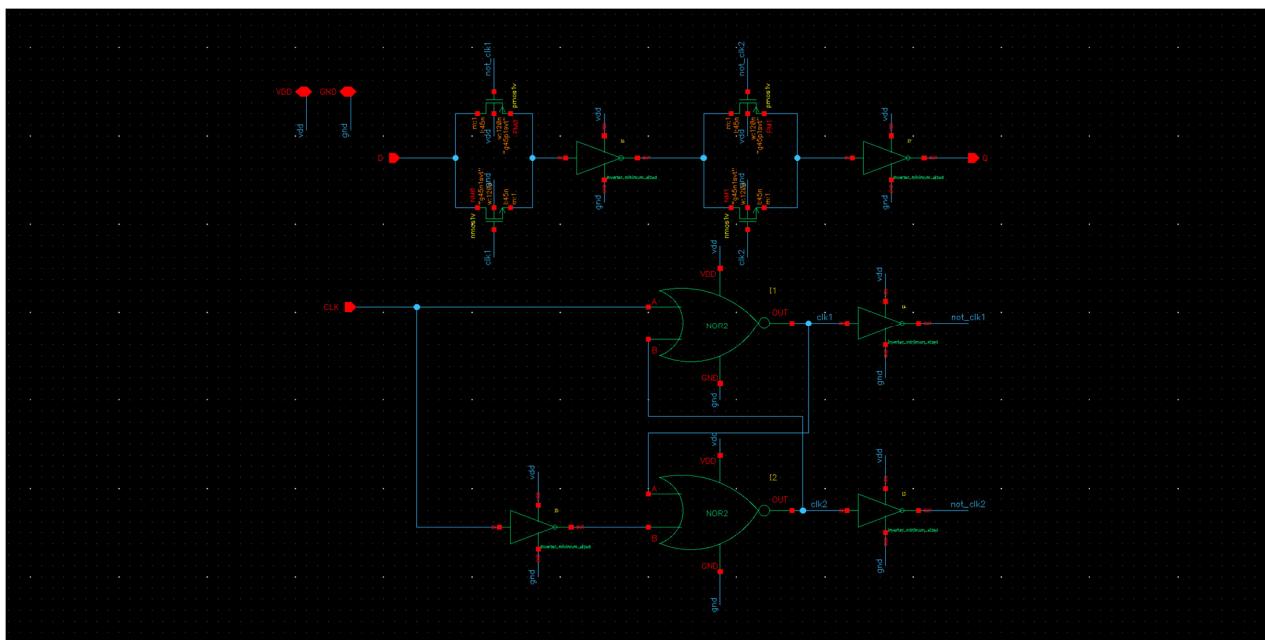


Figure 5. D Flip Flop Schematic

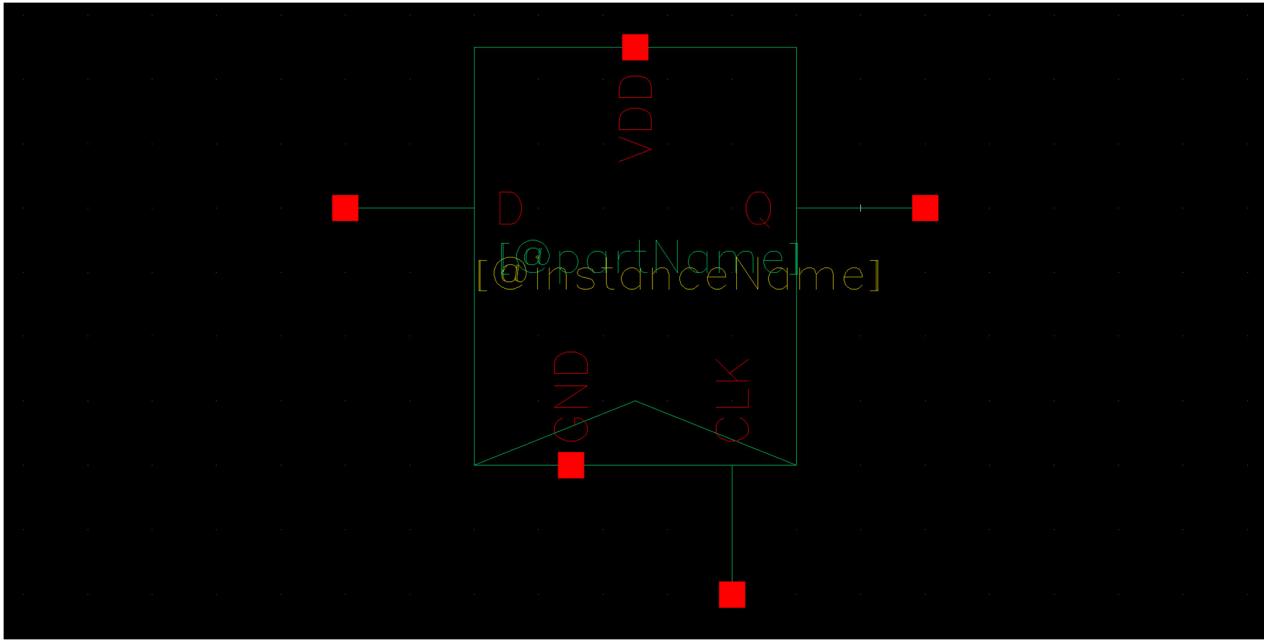


Figure 6. D Flip Flop Symbol

The D Flip-Flop serves as a timing element in the CLB design. It captures the output of the 16:1 LUT at the rising edge of the clock (CLK) and holds the data until the next clock cycle. This ensures signal stability and proper synchronization with the overall system timing.

2.2 SIPO (Serial in Parallel Out shift register)

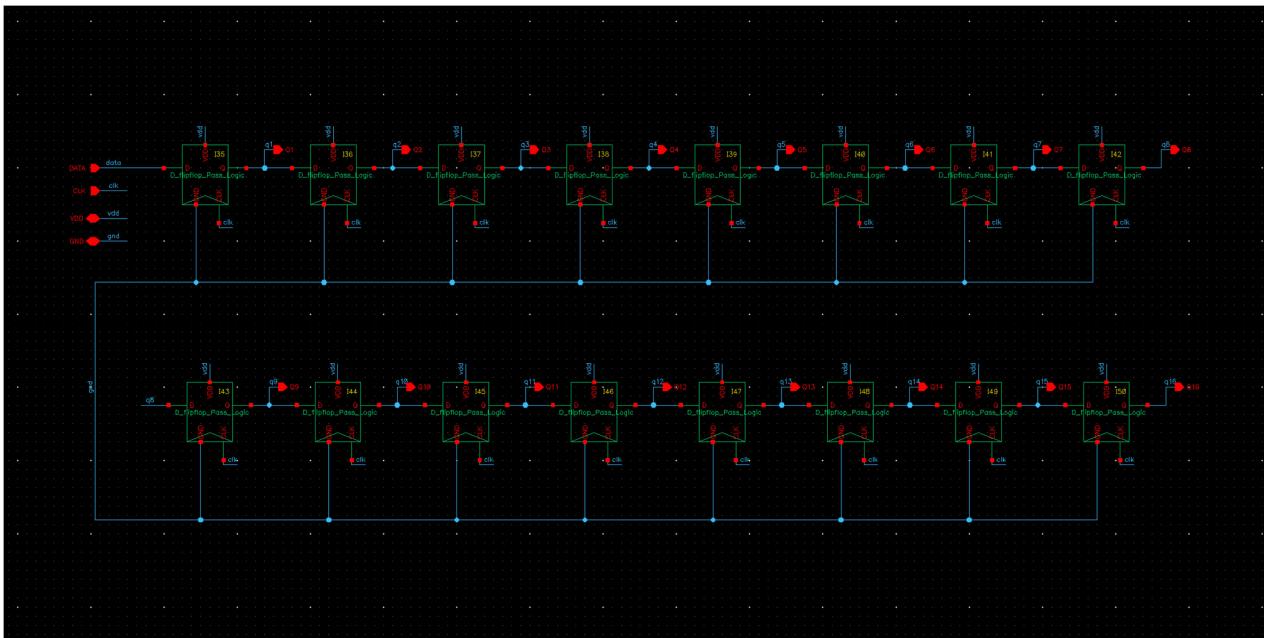


Figure 7. SIPO Schematic

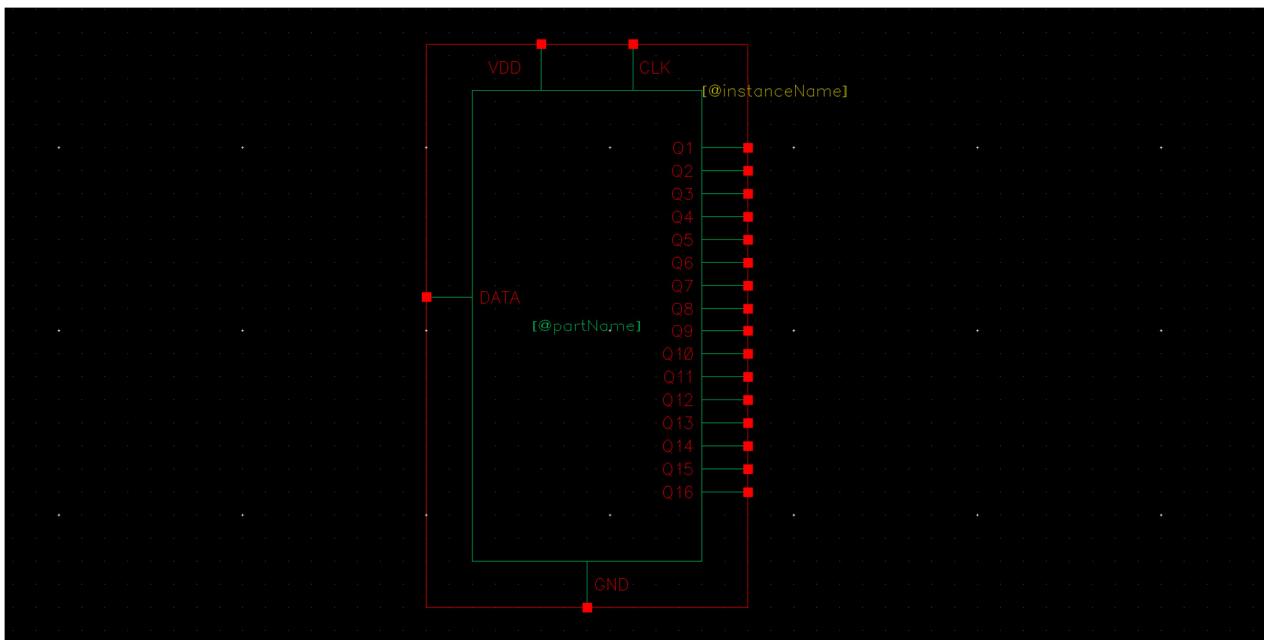


Figure8. SIPO Symbol

The Serial-In-Parallel-Out (SIPO) shift register takes serial data input and converts it into a parallel output. This parallel data is then stored in the SRAM array, enabling efficient data access for subsequent operations.

3. 6T SRAM Array Design

3.1 6T SRAM Cell

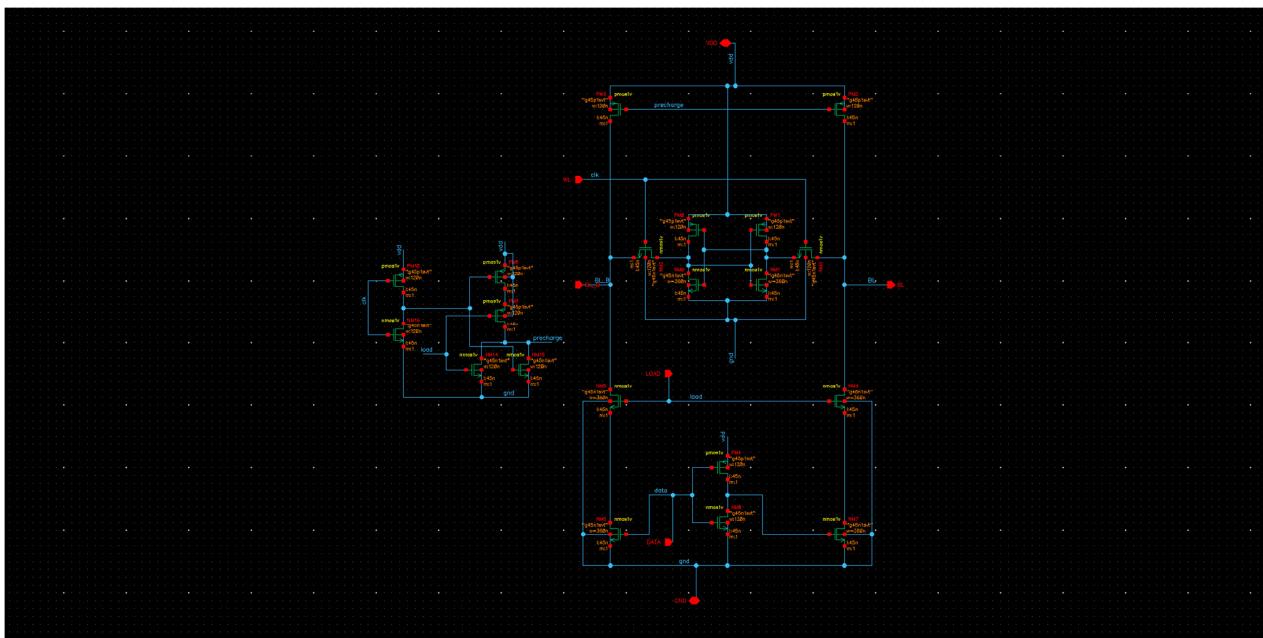


Figure9. SRAM Cell Schematic

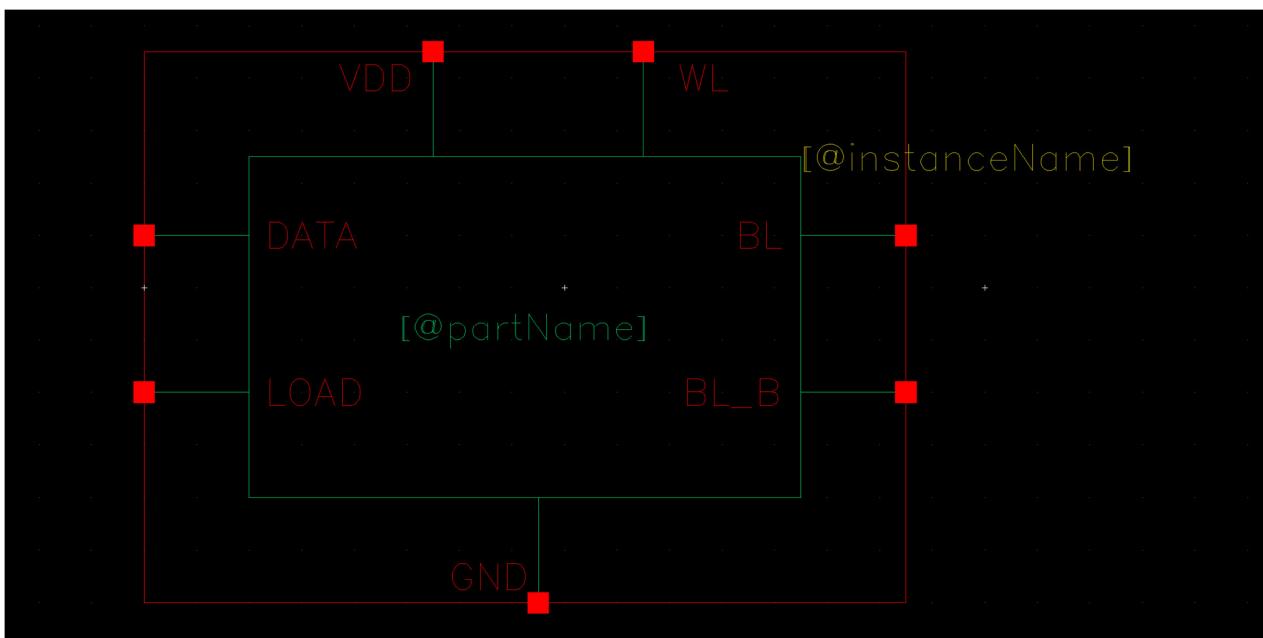


Figure10. SRAM Cell Symbol

6T-SRAM Circuit:

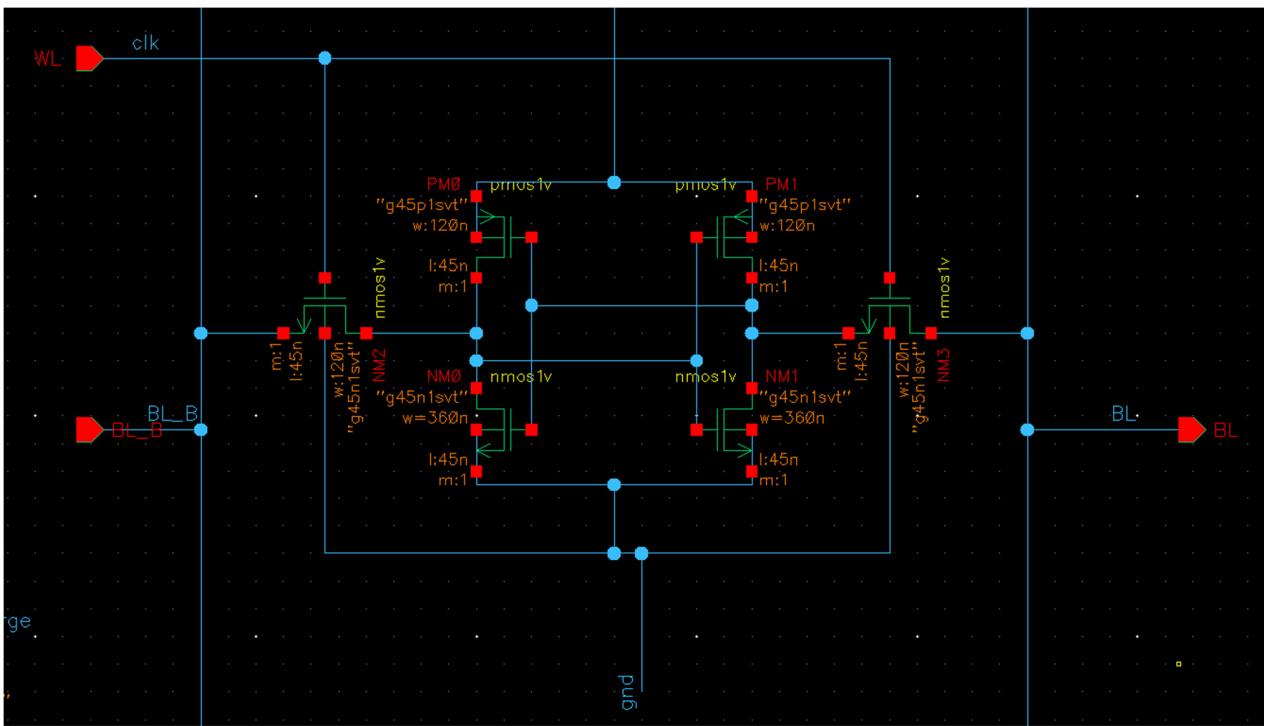


Figure 11. 6T-SRAM

Precharge Circuit:

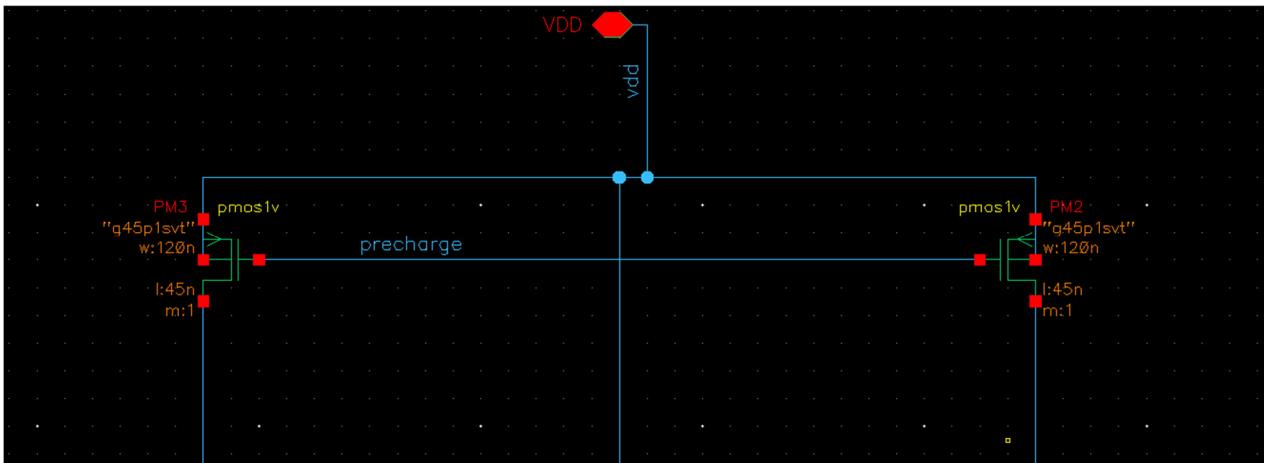


Figure 12. Pre-charge Circuit Schematic

Pre-charge Signal Generation Circuit:

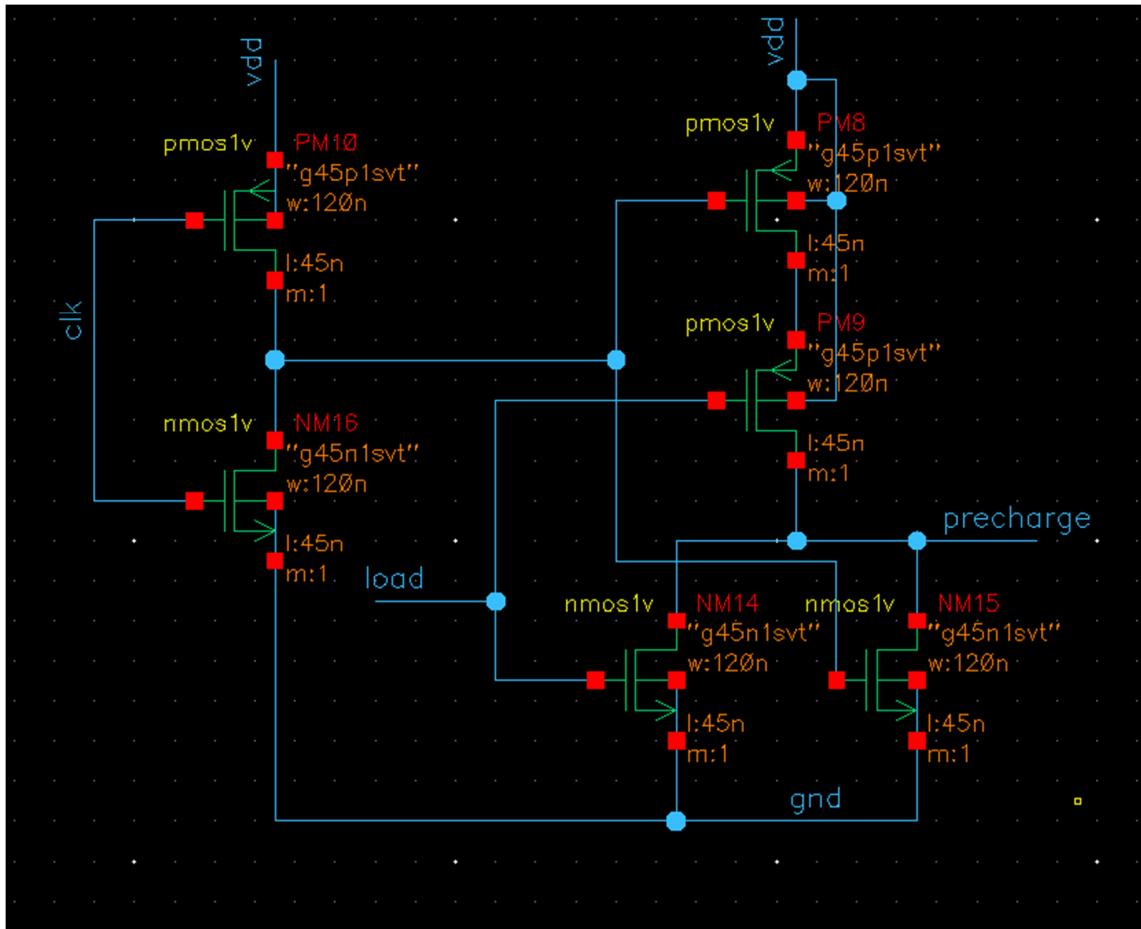


Figure13. Pre-charge Generation Circuit Schematic

Write Control Circuit:

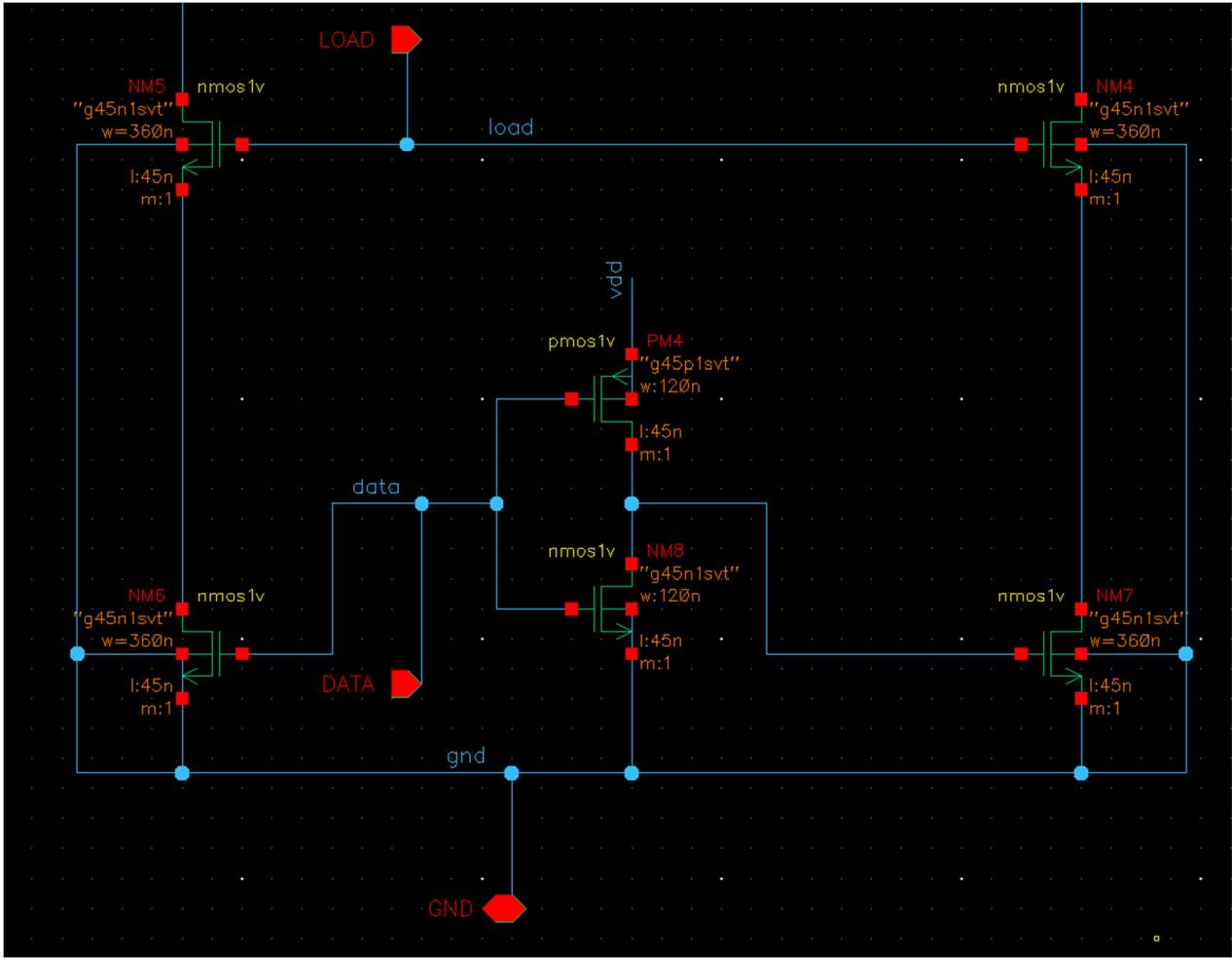


Figure 14. Pre-charge Control Circuit Schematic

The 6T SRAM cell is the fundamental storage unit of the SRAM array, designed for reliable read and write operations. The pre-charge circuit ensures proper initialization of bit lines before accessing the cell and the write control circuit is activated when doing write/load operation.

3.2 SRAM Array Schematic

Schematic:

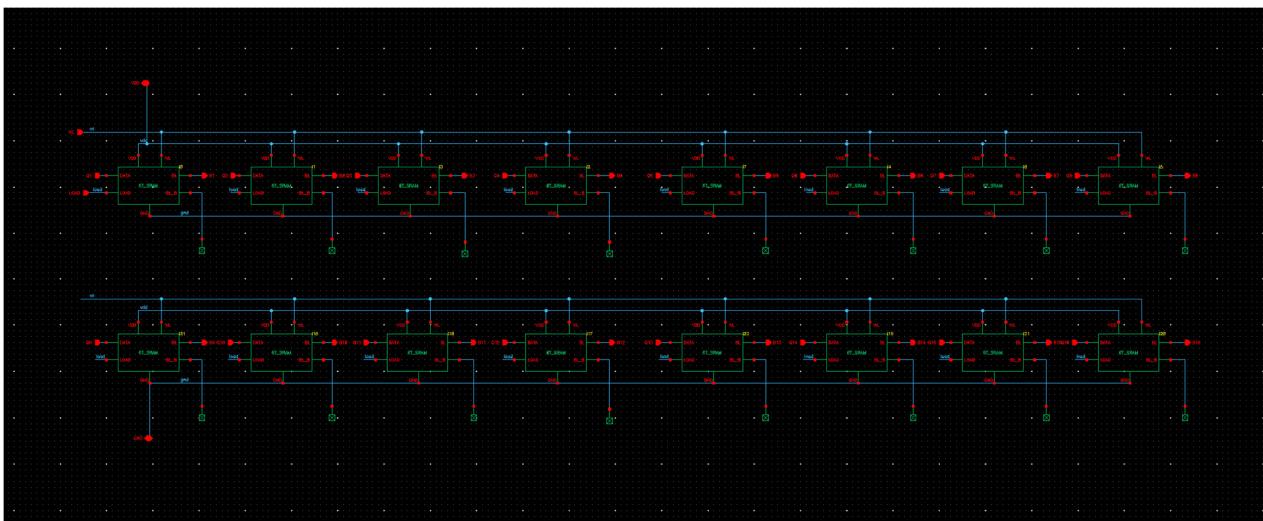


Figure 15. SRAM Array Schematic

Symbol:

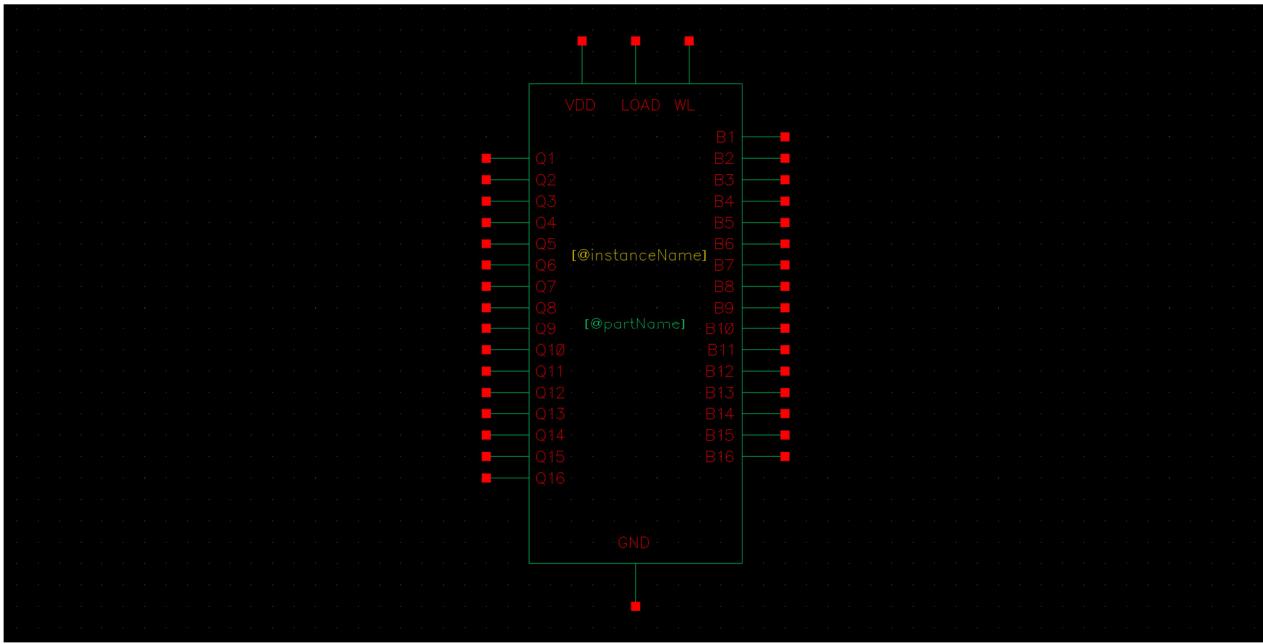


Figure 16. SRAM Array Symbol

SRAM array consists of 16 SRAM cell connected parallel (with shared WL/CLK and LOAD line). It will take the output of the SIPO (16 single data bit converted from a serial data stream) and then store in the SRAM cell, waiting for read command (feed to the input of next stage: 16:1 LUT)

4. LUT Design

4.1 The 2:1 iMUX Schematic

Schematic:

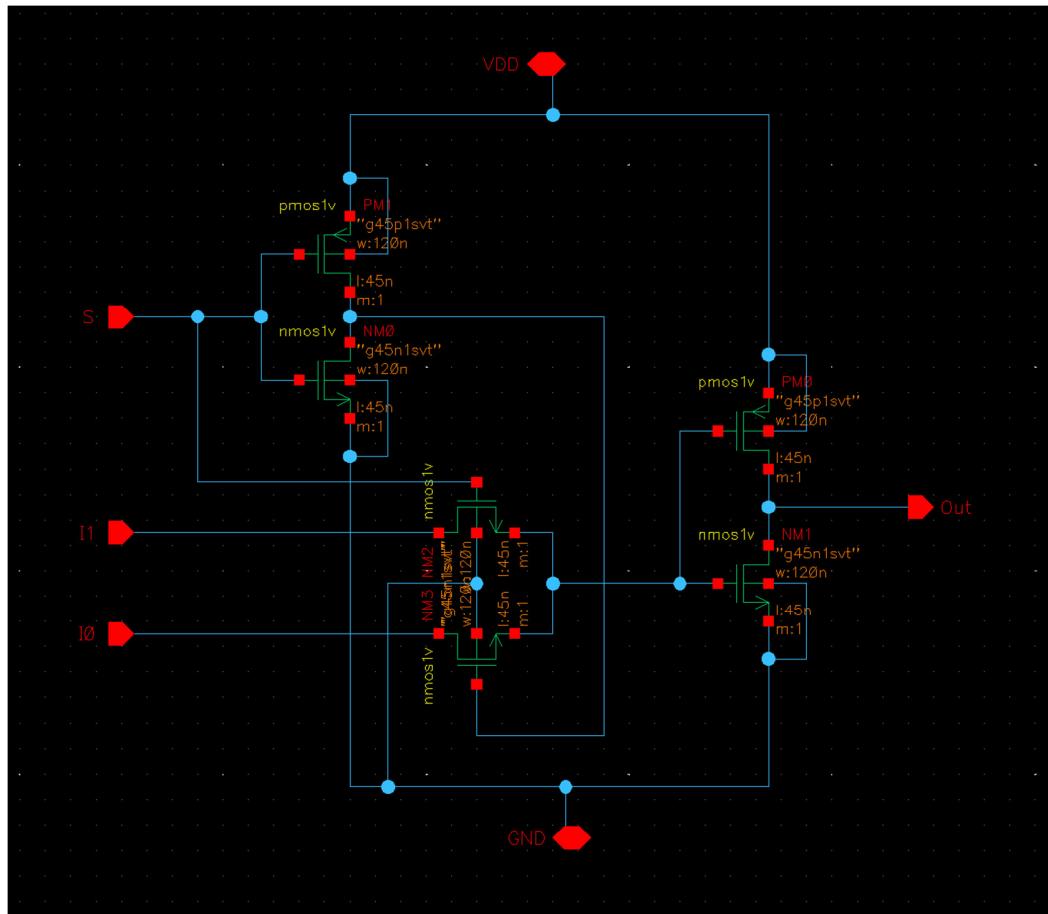


Figure 17. 2:1 iMUX Schematic

Symbol:

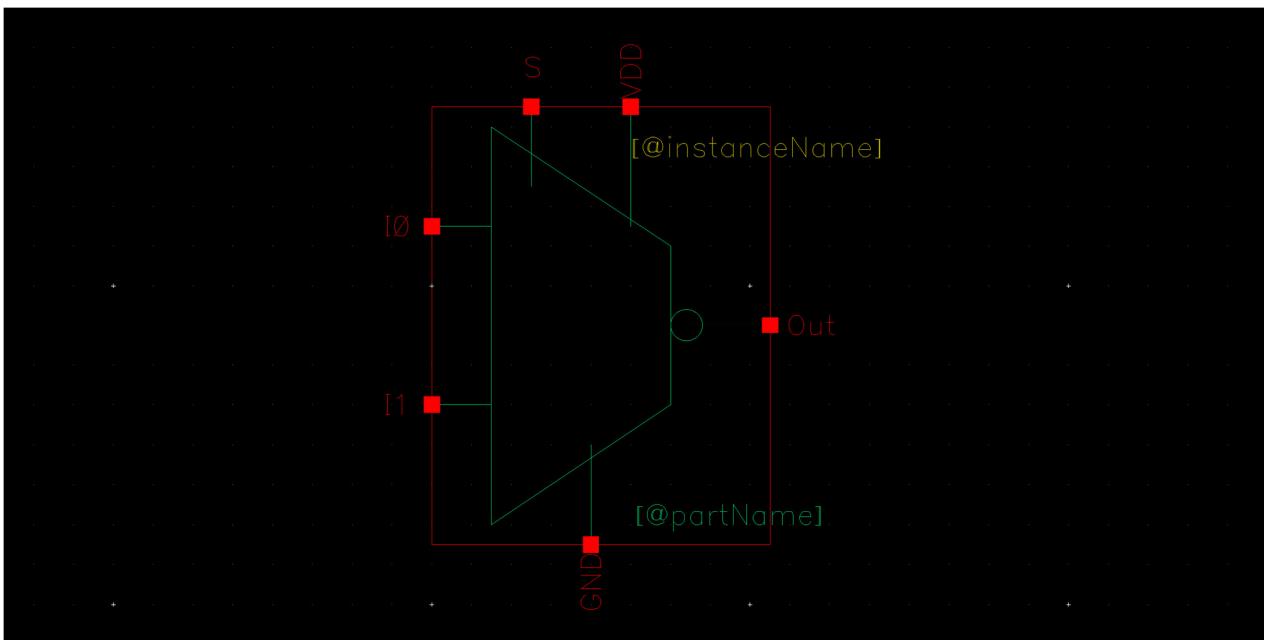


Figure 18. 2:1 iMUX Symbol

The 2:1 iMUX is a basic multiplexer used in the LUT to select between two input bits based on the control signal. A complete 16:1 LUT consists of 15 iMUX.

4.2 The 16:1 LUT Schematic

Schematic:

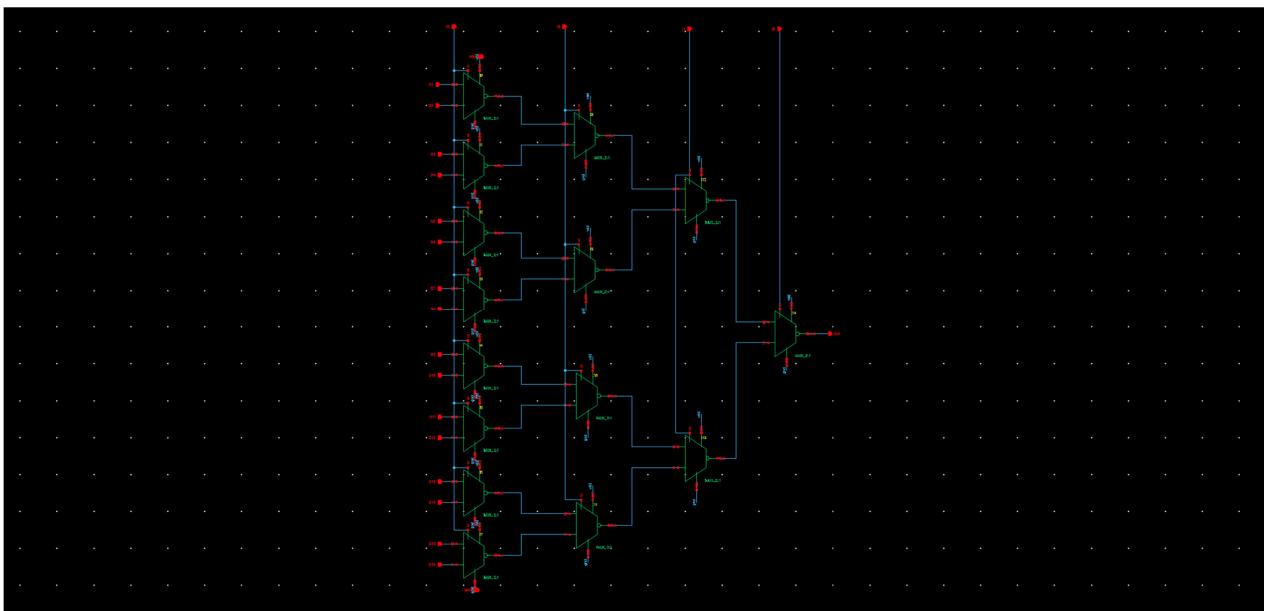


Figure 19. 16:1 LUT Schematic

Symbol:

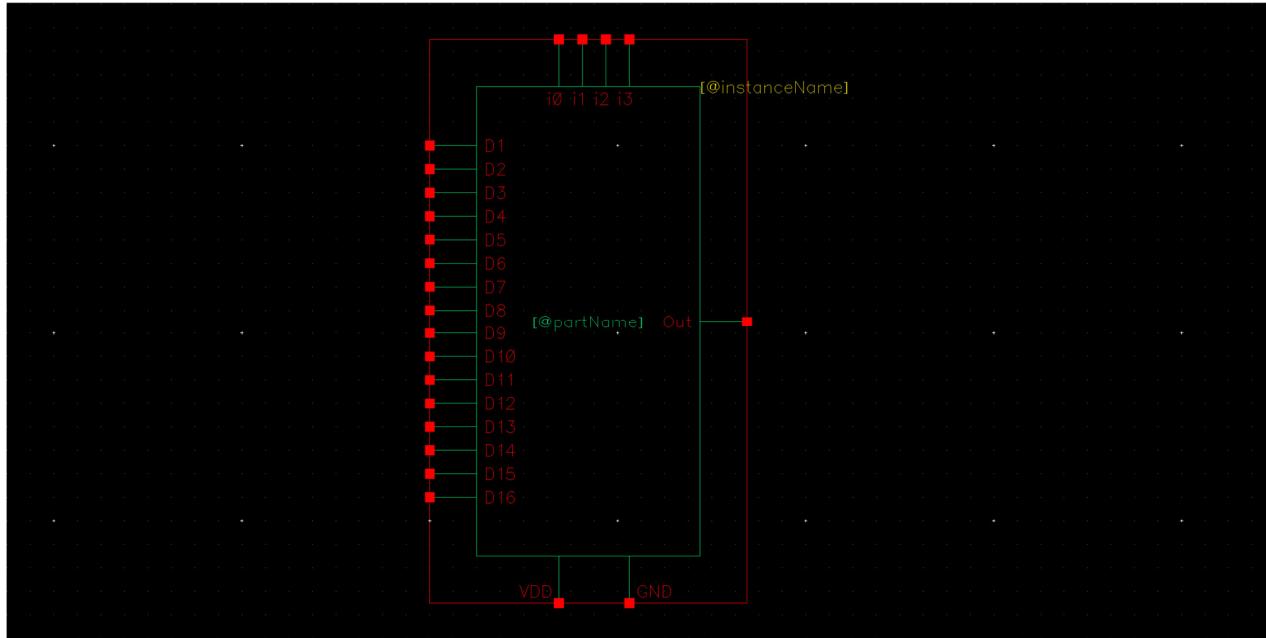


Figure 20. 16:1 LUT Symbol

The 16:1 LUT implements the lookup table functionality in the CLB, selecting one bit out of 16 based on the 4-bit address input (I0 to I3).

5. Clock Frequency Divider

Schematic:

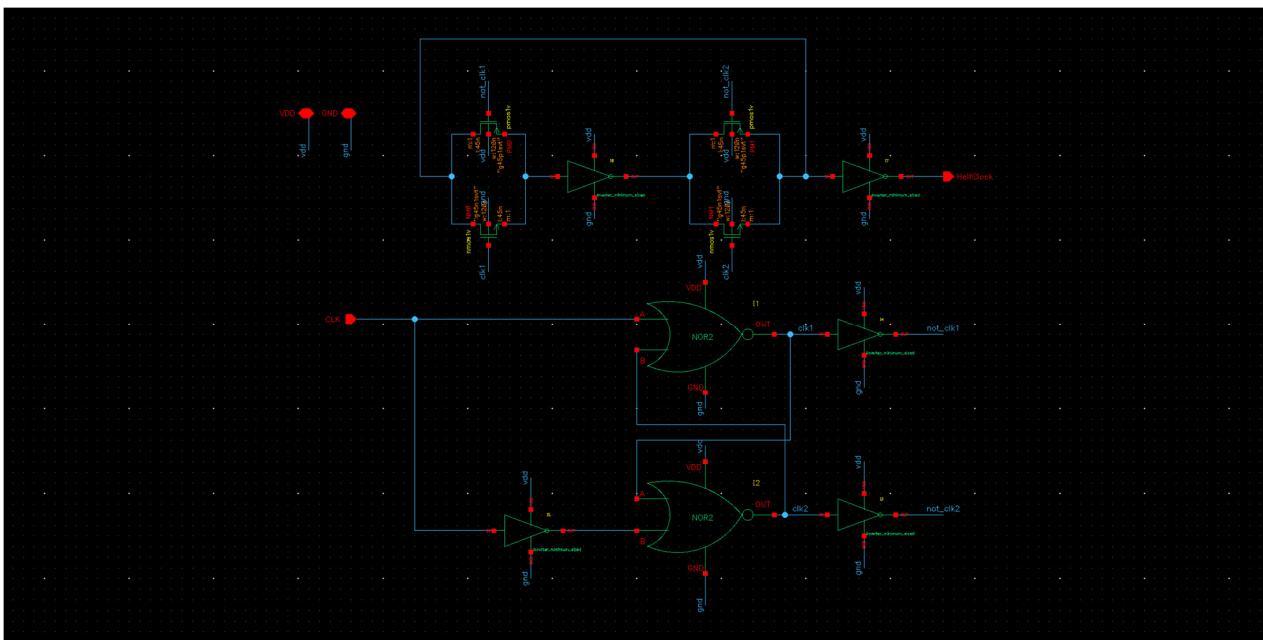


Figure21. Clock Frequency Divider Schematic

Symbol:

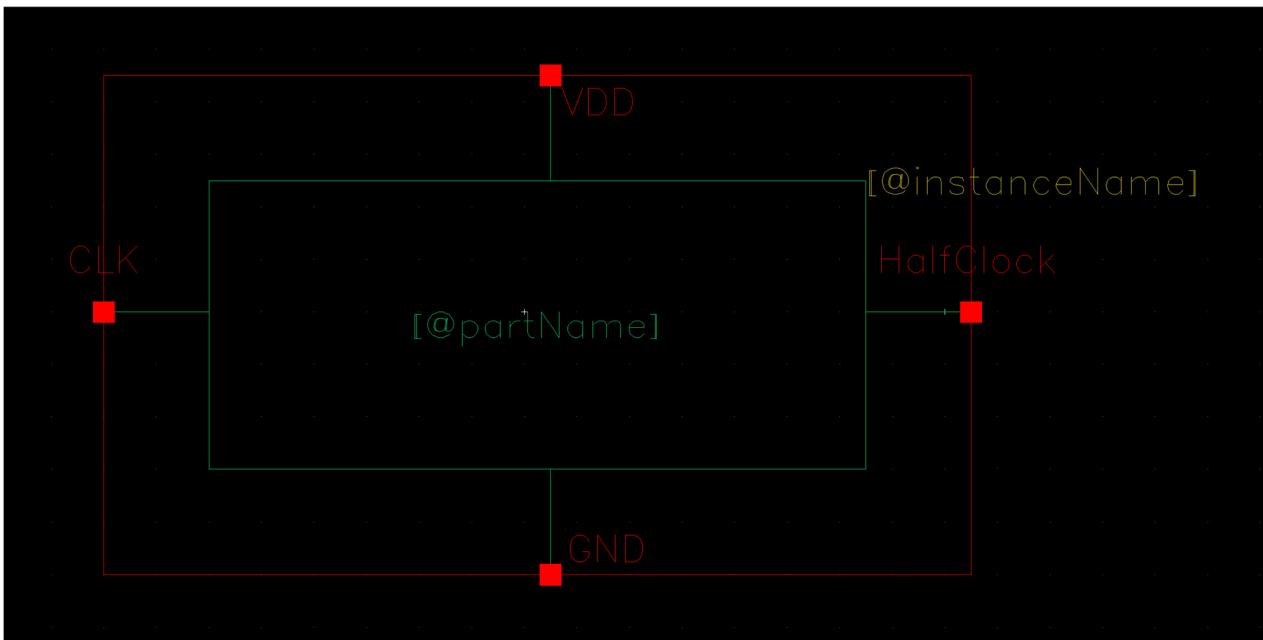


Figure22. Clock Frequency Divider Symbol

The Clock Frequency Divider is used to generate a half frequency clock by using D Flip-flop structure circuit. The two different frequency clock signals will be feed to two D Flip-flop to get a perfectly one-bit shift output, which will be detailed explained in the last part (optimization write up) of this report.

II. Design Description

1. Overall Structure Description

The whole CLB consists of four parts: SIPO (Serial in Parallel Out) shift register, SRAM array, 16:1 LUT and D Flip-flop. There are four input pins: CLK (synchronize SIPO, SRAM and D Flip-flop), LOAD, DATA, LUT selection line (4 bits) and a single output: Out.

In the first stage, a serial data input (one 16bits word) will be converted to parallel output (single bit) and fed to SRAM array. The second stage SRAM array consists of 16 SRAM cells which share a common WL (Word Line / CLK). Each SRAM cell provides one output as the next stage's input. The next stage is a 16:1 LUT, according to the 4-bits address line input, a single bit will be selected from 16 available SRAM cell as the output for LUT. Then the output will be stored in D Flip-flop, until a rising level of CLK to let CLB deliver the output signal to ensure the right timing for this design.

2. SRAM Cell Design Consideration

SRAM cell is designed based on the standard read/write timing logic. 6T-SRAM with a pre-charging circuit and a writing circuit is used.

Based on the desired inputs of SRAM, there will be an external circuit required as the interface for the data stream and the basic 6T SRAM cell.

- **LOAD:** load is connected to the lower side of BL and BL_bar as a write enable switch, it makes the BL able to write to SRAM cell only if the LOAD signal is high, therefore, a NMOS is a perfect device to implement this logic: connect the BL to external circuit when LOAD is high.
- **DATA:** input signals that writing data to SRAM shall be applied to this line. NMOS is also used here to drive the data into BL. It is worth noticing that the DATA signal is applied to pull down the BL, therefore an INV is added to DATA before it connects to BL.
- **CLK:** this signal is connected to Word Line of SRAM cell as the gateway to the storage Q and Q_bar of the 6T SRAM. Connect Q to BL only if the CLK signal is high.

Another input signal pre-charge will be detailedly explained in the next part.

3. Pre-charge Signal in SRAM

The generation of pre-charging signal is critical for SRAM design. During the reading process, we need to first enable pre-charging, then activate WL after disconnecting pre-charging. In addition, WL controls 2 pass logic NMOS and Pre-charging signal controls 2 PMOS. Therefore, WL and pre-charging signal should be same in reading situation (When LOAD is low). When LOAD is high, which means SRAM are doing writing operation, Pre-charge signal is set to be 0. This setting will make the

pre-charging PMOS always on, which makes bit-line (BL) remains a floating voltage (0 to 1.1V) when writing 0. To compensate this issue, we size the NMOS in writing circuit to be 3 times wider than the PMOS in pre-charging circuit, which allows bit-line can be pull very low when writing 0 even pre-charging circuit is on.

Another issue caused by the pre-charging circuit is the undesired high signal in the reading process. This phenomenon occurs when Pre-charging circuit is enabled and all the path to ground is blocked (Both of LOAD and WL is low). However, this bit belongs to the reading preparing stage instead of the actual reading. With the help of the CLK, we would not output this bit.

The pre-charge signal is generated by an OR gate whose inputs are CLK_bar and LOAD, the reason of choosing this combinational logic is that we need to pre-charge the signal in when CLK's low phase ($\text{CLK_bar} = 1$), and the pre-charge is only disabled when the CLK is high (enable WordLine) and LOAD is low (read operation).

Pre-charge signal distortion: the reading process of a SRAM requires pre-charge the BL to high before read operation. Therefore, when reading 0 from a stored 0, the output of BL will be experiencing an undesired voltage high signal at CLK low stage

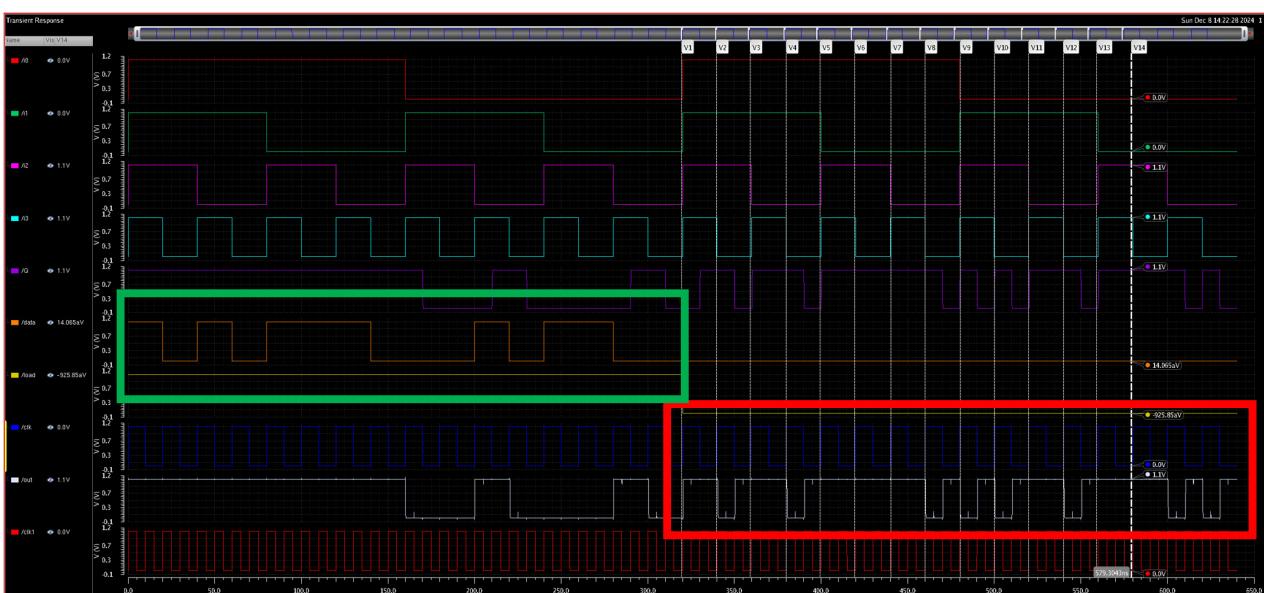


Figure 23. Pre-charge signal distortion pattern (raw simulation)

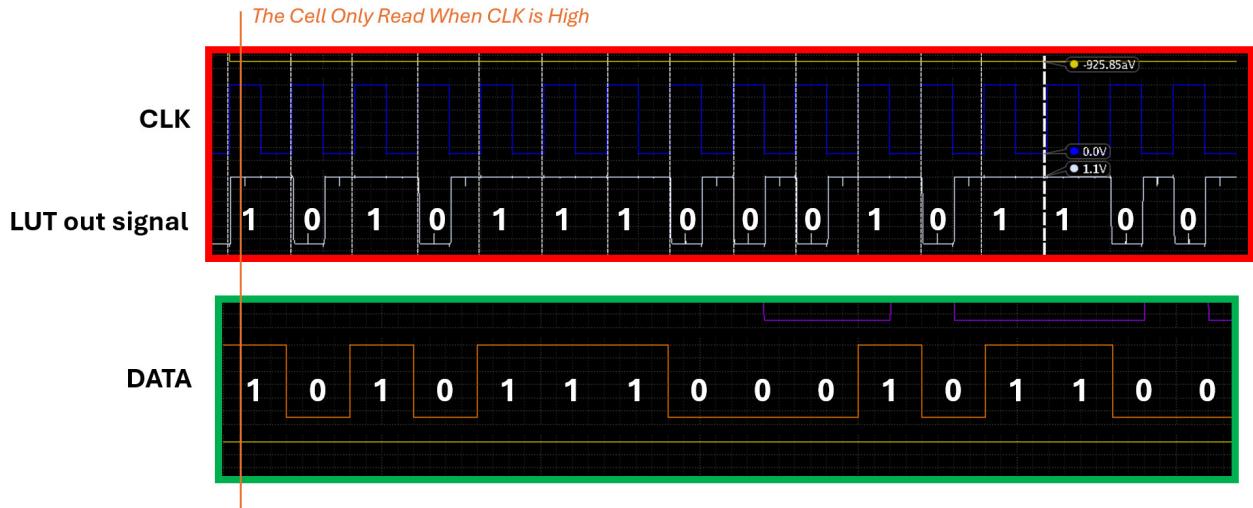


Figure 24. Pre-charge signal distortion pattern (with label)

As the output of LUT16:1 verified, the next stage of signal goes into D flip flop to erase the distortion caused by the pre-charge signal (pull BL high before read).

4. Timing and Control Logic

Basically, in this part we explain why we have two flipflop at the last stage. Due to the delay caused by our CLB circuit, the output signal of LUT will be slower than the CLK signal, which results of the rising edge catching mismatch of the final D Flip Flop. During the rising edge of the CLK signal, the input signal D from the LUT does not stabilize until the end of the rising edge. As a result, the final output Q may fail to capture the correct value, leading to data errors, as the Figure 25 shown below.

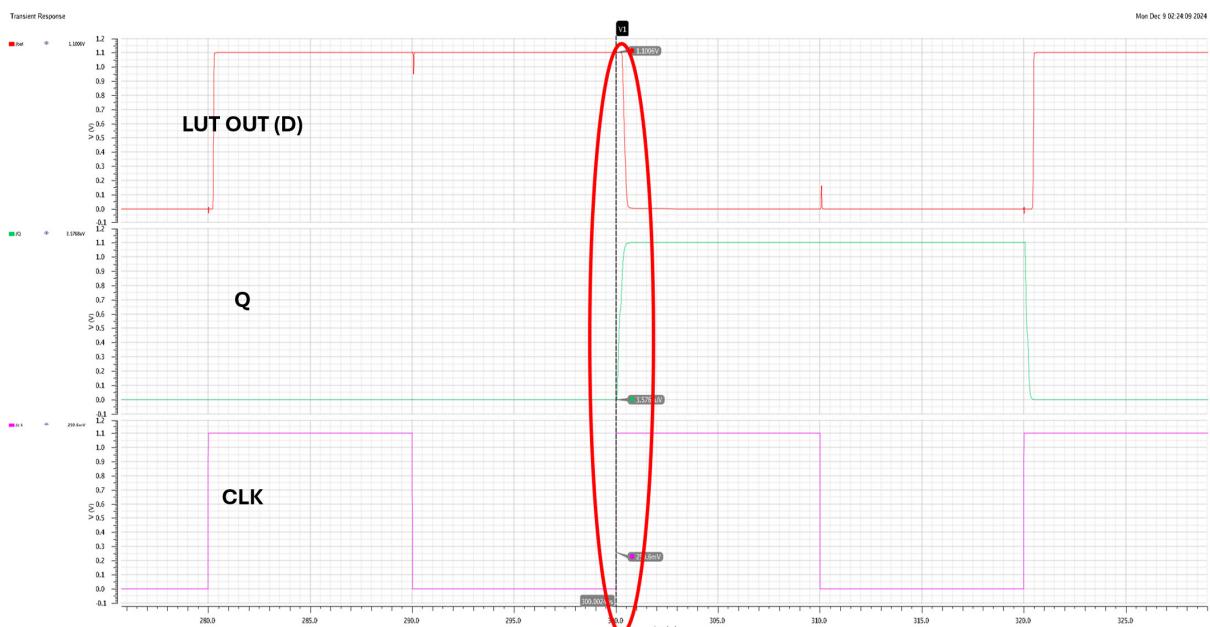


Figure25. D-FlipFlop Fail to capture LUT output at rising level

One of the methods to resolve this issue is to use an additional D Flip Flop with a higher CLK frequency to set **two rising edges within one DATA cycle**. The first rising edge could be ignored by the flip flop, but the second rising edge occurs at the middle of one DATA input period, which is unlikely to be mismatched. In Figure 26, when CLK rise, the output has not been set, which cause the capture failure. However, CLK1, which is double the frequency of the CLK, successfully catch the signal at the its second rising edge/end of CLK period & falling edge, also the end of the data bit. This method solved mismatch problem the caused by missing the rising edge of lower CLK (with frequency equals to the data bit)

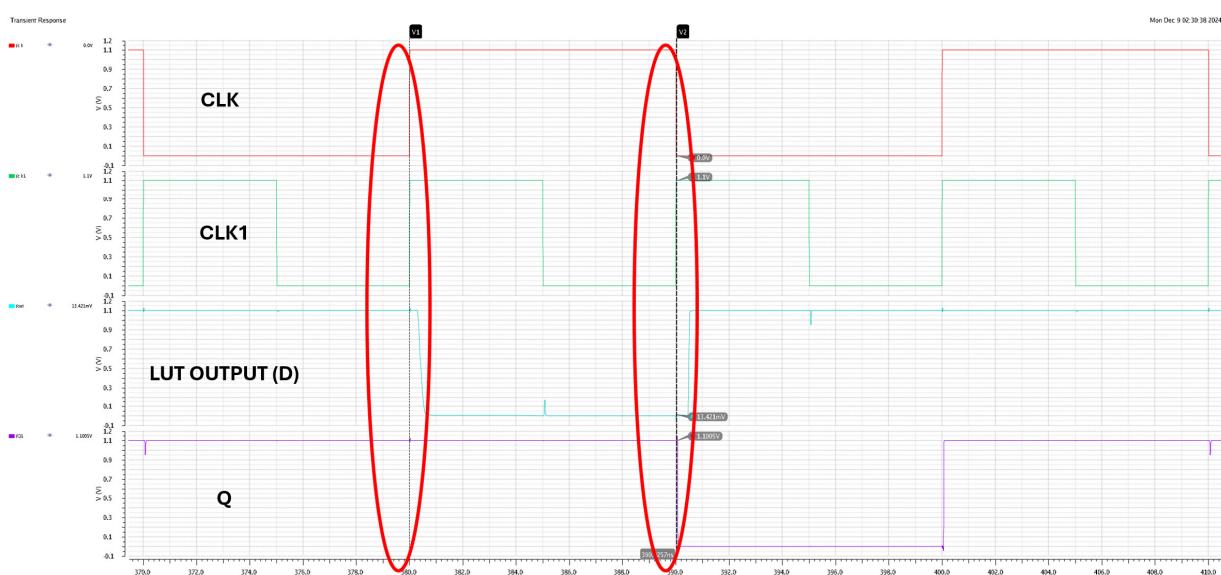


Figure26. Using 2 rising edges to catch signal

The first D Flip-Flop, clocked by CLK1, ensures the LUT output is correctly latched without delay. However, this introduces a half-phase shift in the signal relative to CLK. The second D Flip-Flop, operating at the regular CLK frequency, restores proper phase alignment for the final output signal. This dual Flip-Flop configuration ensures reliable data capture and synchronization, mitigating timing issues caused by circuit delays. (In actual cadence simulation, we take input clock as CLK1 to feed first flipflop and use a clock frequency divider to provide a half frequency clock for all other clock, such as SIPO, SRAM cell and second flipflop)

III. Design Validation/Verification

1. LUT Validation

1.1 2:1 iMUX Validation

Test Schematic:

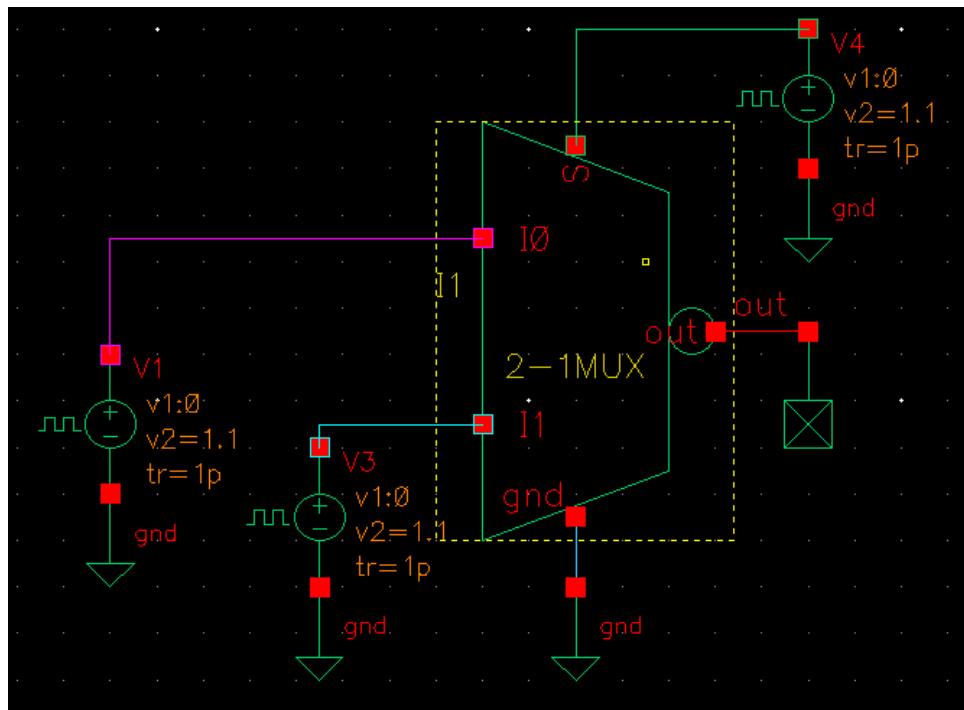


Figure27. Test Schematic of 2:1 iMUX

Table1. Parameters of Voltage Source for 2-1MUX Test Schematic

Parameters for V1 (I0)		Parameters for V3 (I1)		Parameters for V4 (S)	
<i>Voltage 1</i>	0 V	<i>Voltage 1</i>	0 V	<i>Voltage 1</i>	0 V
<i>Voltage 2</i>	1.1 V	<i>Voltage 2</i>	1.1 V	<i>Voltage 2</i>	1.1 V
<i>Period</i>	120n s	<i>Period</i>	240n s	<i>Period</i>	1u s
<i>Delay time</i>		<i>Delay time</i>		<i>Delay time</i>	
<i>Rise time</i>	1p s	<i>Rise time</i>	1p s	<i>Rise time</i>	1p s
<i>Fall time</i>	1p s	<i>Fall time</i>	1p s	<i>Fall time</i>	1p s
<i>Pulse width</i>	60n s	<i>Pulse width</i>	120n s	<i>Pulse width</i>	500.0n s

Simulation result:

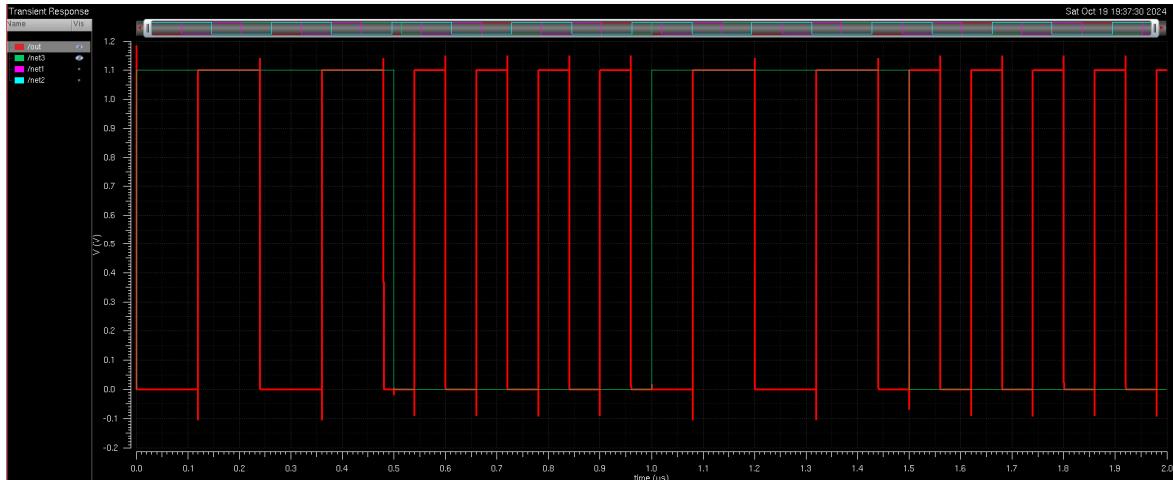


Figure28. Test Result of 2:1 iMUX

As it shown in the result above, the output will be I0 when S (select) is low, and the output will be I1 when S (select) is high. In this case, this 2-1MUX is functionally correct.

This test set is using a nanoseconds level simulation as the designed 2-1MUX has a longer propagation delay time than the NAND4 or NOR4 gate.

1.2 16:1 LUT Validation

Schematic:

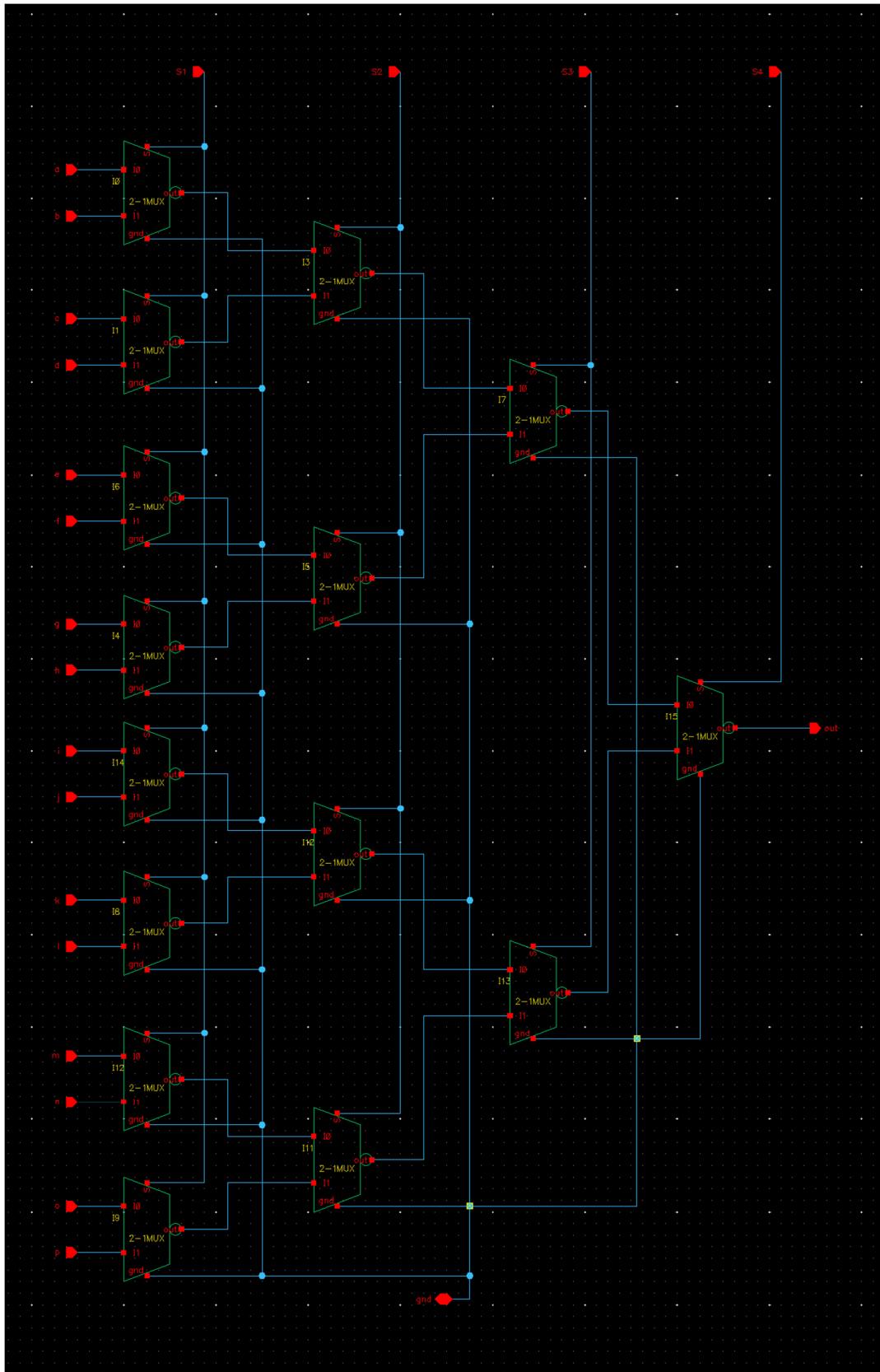


Figure 29. Schematic of 16:1 LUT

To generate the real inputs driven by minimum size inverters, one '16 Real Inputs' circuit and symbol has been created:

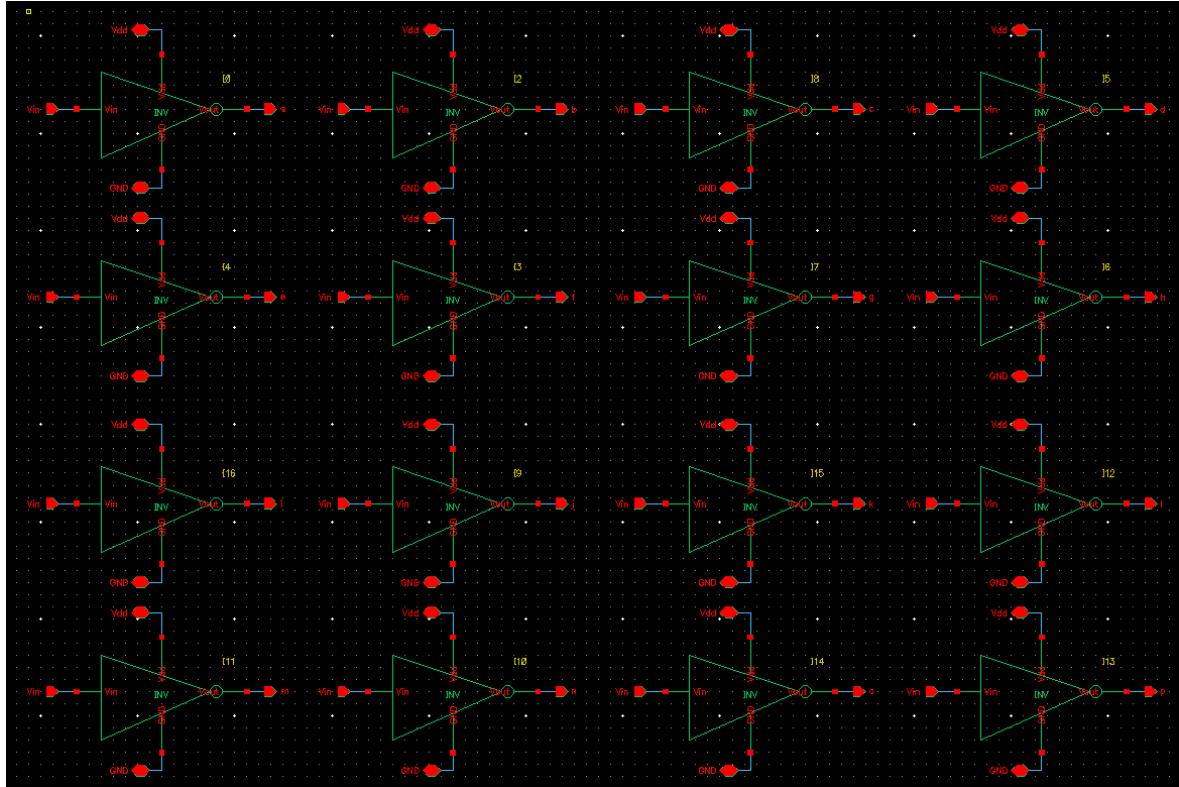


Figure30. Schematic of 16 Real Inputs

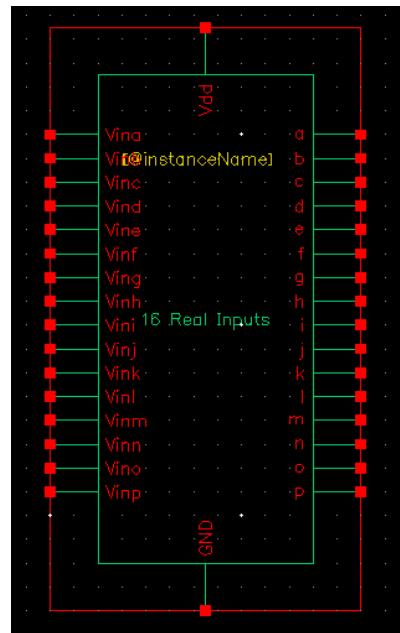


Figure31. Shematic of 16 Real Inputs

Combine the 16 LUT symbol and the 16 read inputs symbol together to draw the final testing circuit shown as below. Driven the address input with INV.

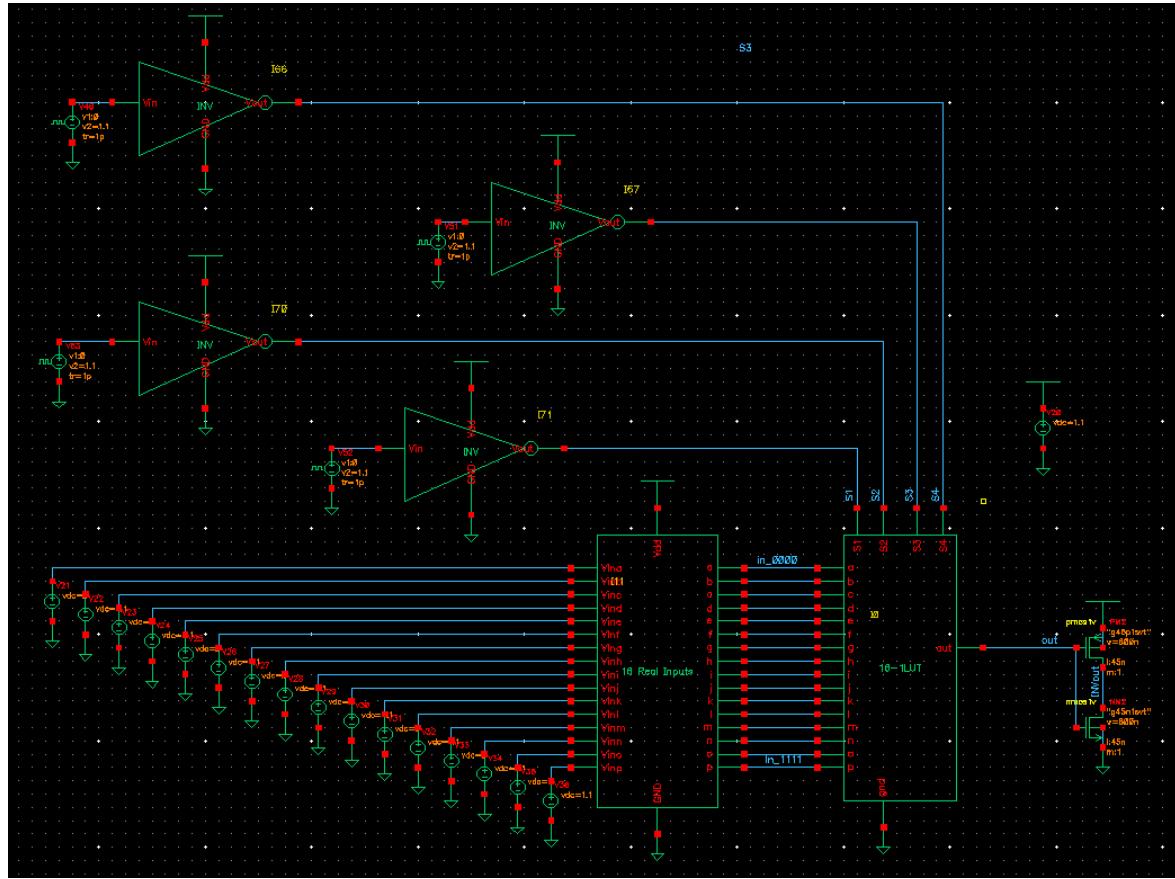


Figure32. Test Schematic of 16 LUT with Real Inputs

Vin are driven by the inputs from 'a' to 'p' as shown in Table2. below:

Table 2. LUT Test Data

Vin	a	b	c	d	e	f	g	h
Voltage	1	0	1	0	1	0	1	0
Code	0000	1000	0100	1100	0010	1010	0110	1110

Continue:

Vin	i	j	k	l	m	n	o	p
Voltage	1	0	1	0	1	0	1	0
Code	0001	1001	0101	1101	0011	1011	0111	1111

Set the select voltage for S1-S4 varies, run the simulation. Result shown below:

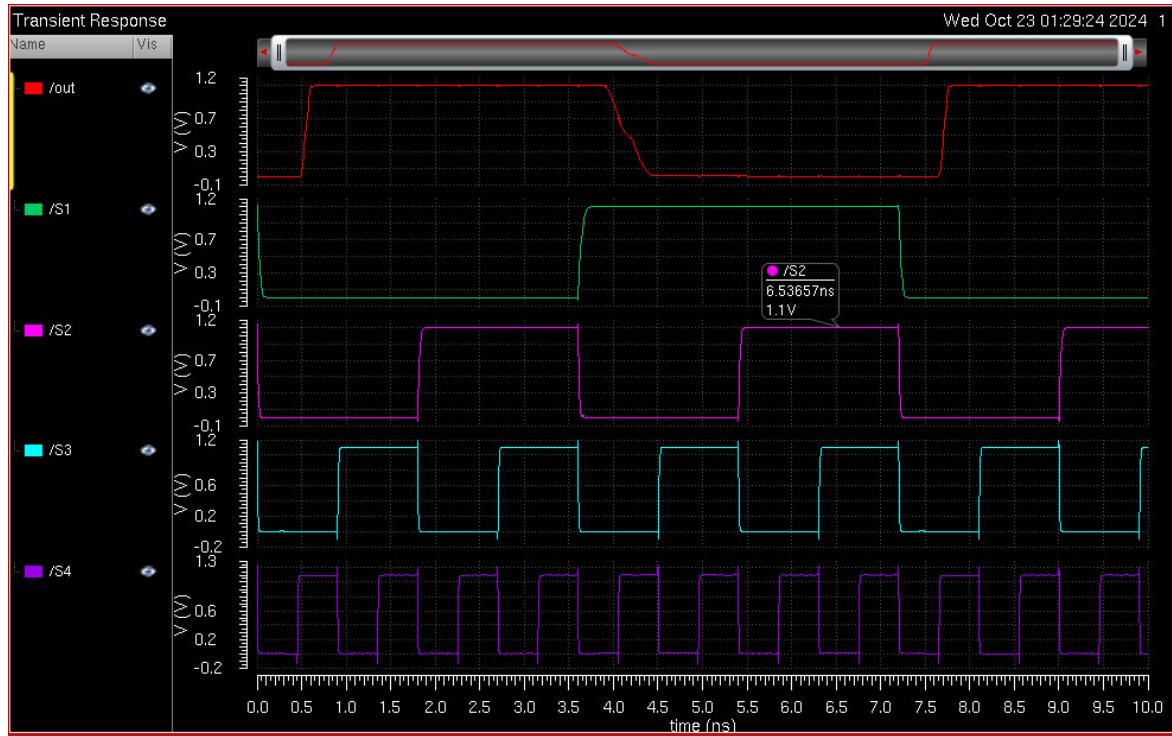


Figure33. Test Schematic of 16 LUT with Real Inputs

S1, S2, S3, S4 inputs vary from 0000 to 1111, and the outputs followed what expected in Table2. Therefore, the 16-1 design is functioning correct.

2. D Flip Flop Validation

Schematic:

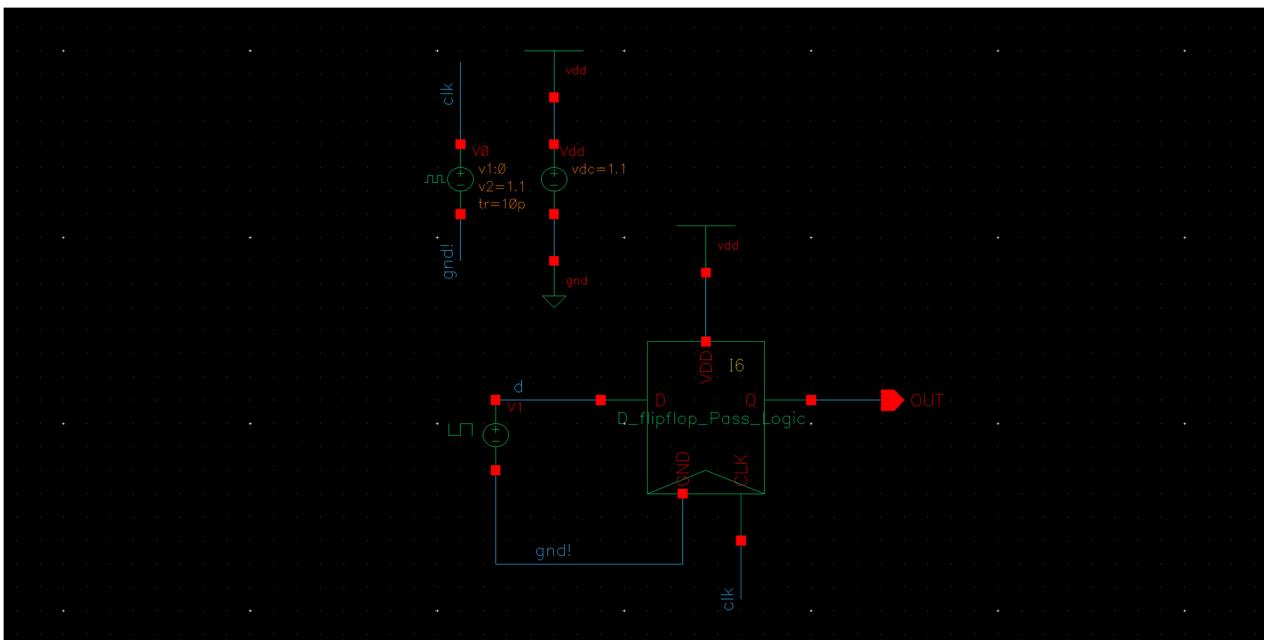


Figure34. Test Schematic of D-FlipFlop

Test Cases:

- Data input vbit: 0110 repeat 4 times, 5 ns period with rise/fall time 10ps
- CLK input vclk: 10ns period, 5ns pulse width clock signal with rise/fall time 10ps

Simulation Result:

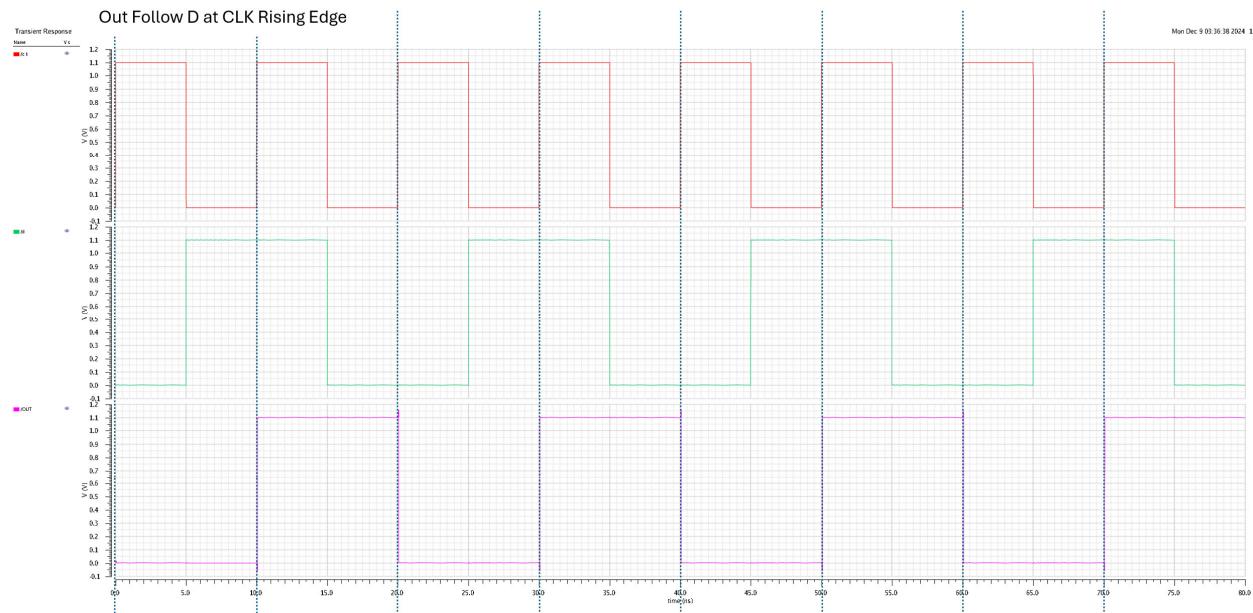


Figure35. Test result of D-FlipFlop

When clock is at rising edge, output will follow the input d. When clock is at other situation, the output will not change. The logic of this design is correct.

3. Serial In Parallel Out (SIPO) Shift Register Validation

Schematic:

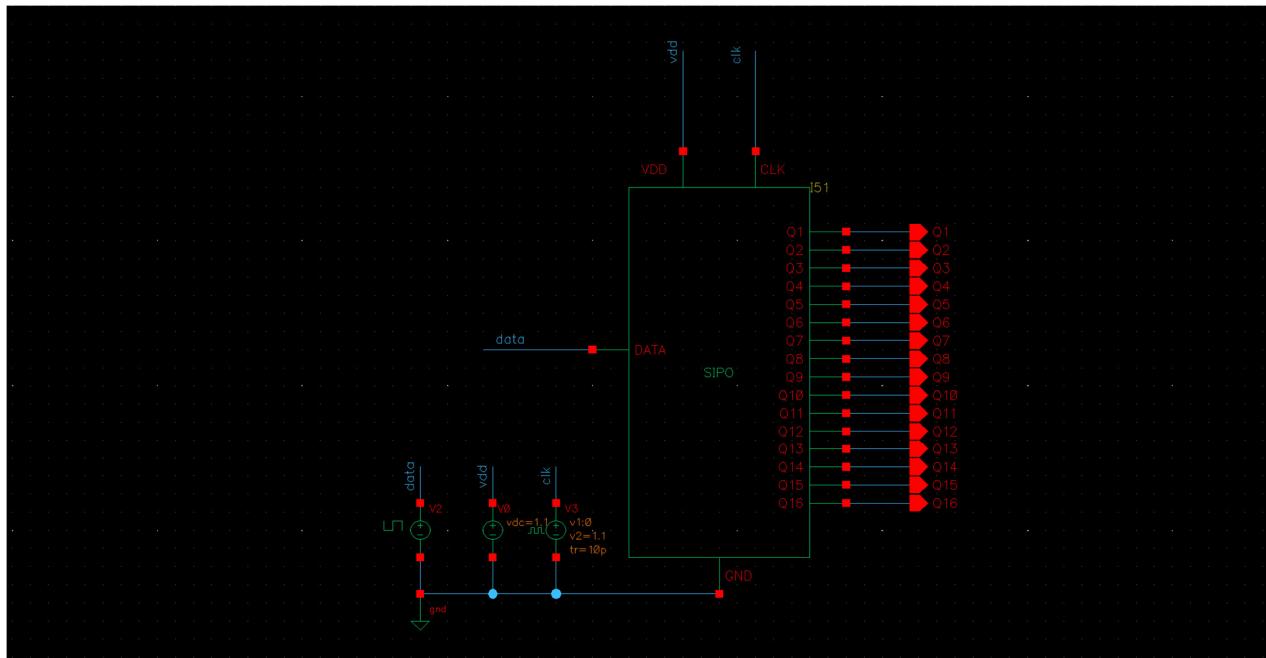


Figure36. Test Schematic of SIPO

Test Cases:

- Data input: 16bits (1010 1110 0010 1100)
- CLK input: 10ns period, 5ns pulse width clock signal with rise/fall time 10ps

Simulation Result:

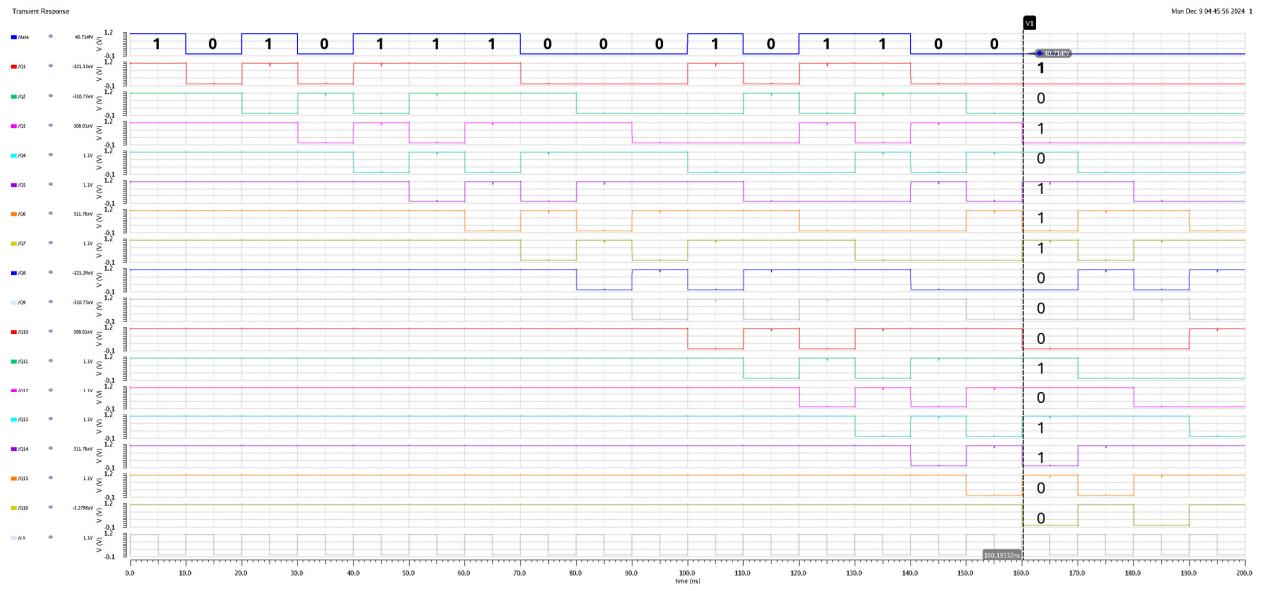


Figure37. Test result of SIPO

The simulation result clearly shows how the input data is load and shift bit by bit. After 16 data bit's period, all the SRAM cells are load.

4. 6T-SRAM Validation

Schematic:

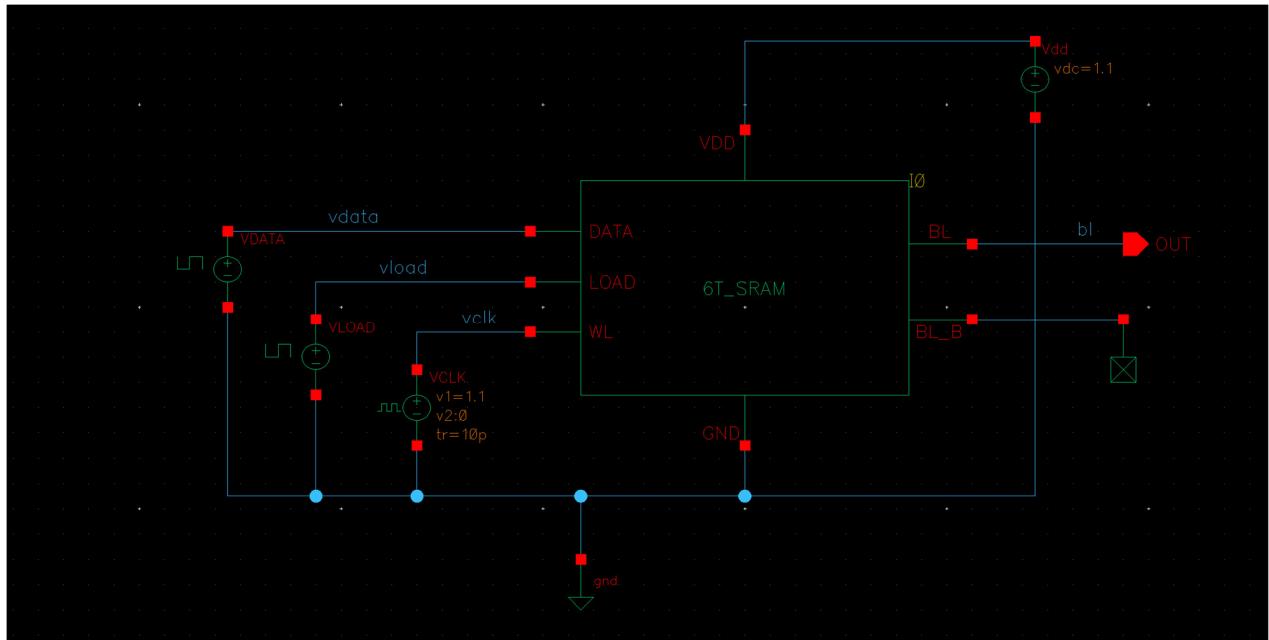


Figure38. Test Schematic of 6T-SRAM

Test Cases:

To validate the performance of the 6T SRAM, all the read and write cases of 6T-SRAM need to be considered, there are:

- A. Write 1 to 0
- B. Write 1 to 1
- C. Write 0 to 0
- D. Write 0 to 1
- E. Continuous Read
- F. Continuous Write

Test inputs setup for case A, E, F (W0 R0 R0 W1 R1 R1) have shown in Table 3.

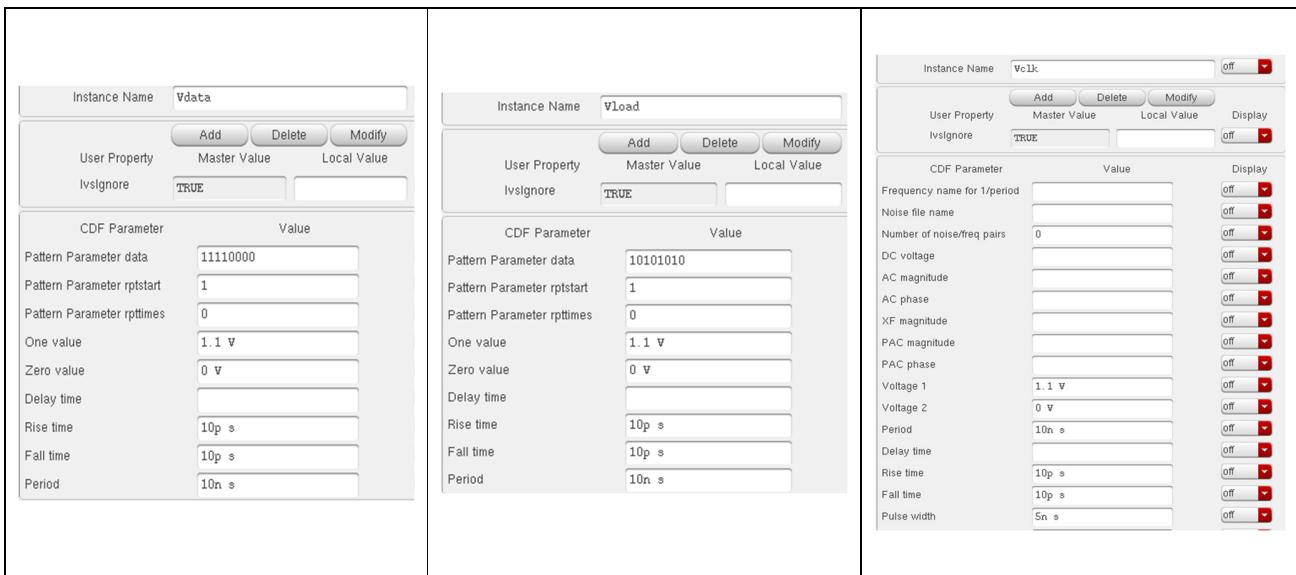
Table3. Test Inputs Setup for A, E, F

Data input	Load control	Clock

Test inputs setup for case B, C, D (which means W1 R1 W1 R1 W0 R0 W0 R0)

Table4. Test Inputs Setup for B, C, D

Data input	Load control	Clock
------------	--------------	-------



Simulation Result:

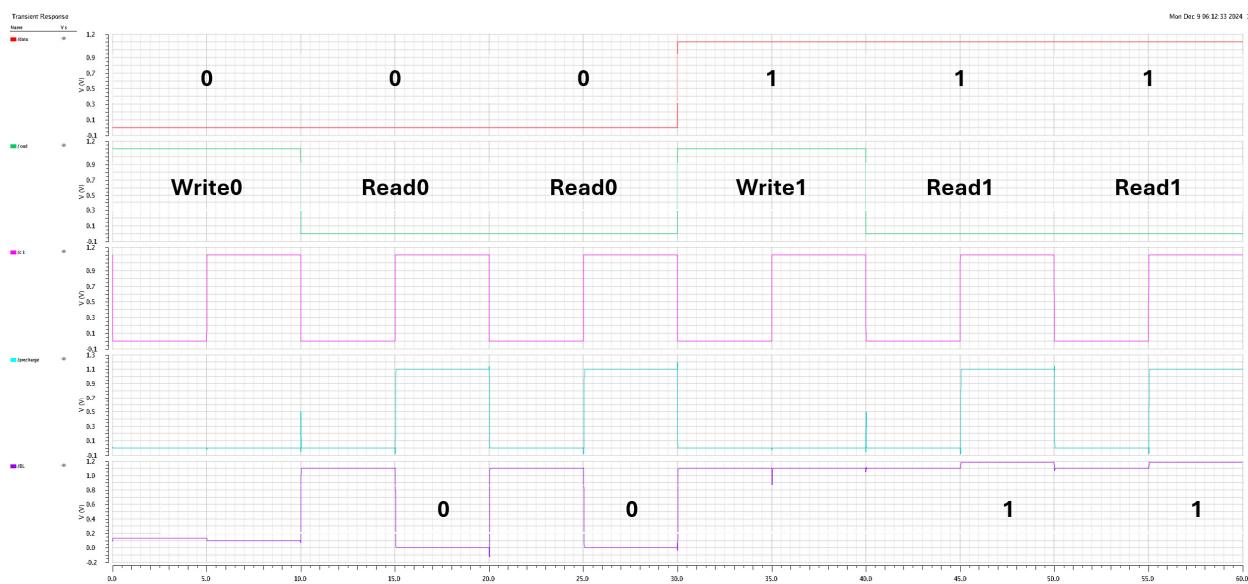


Figure 39. Validation Result of 6T-SRAM case A, E, F

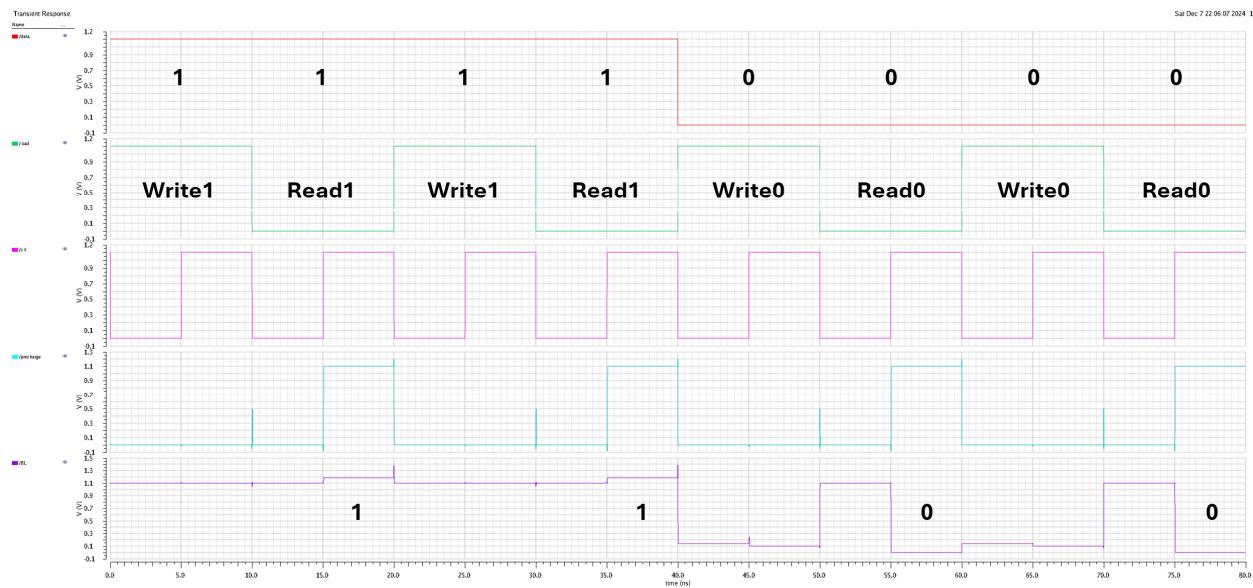


Figure40. Validation Result of 6T-SRAM case B, C, D

5. SRAM Array Validation (SIFO in to SRAM Array out)

Schematic:

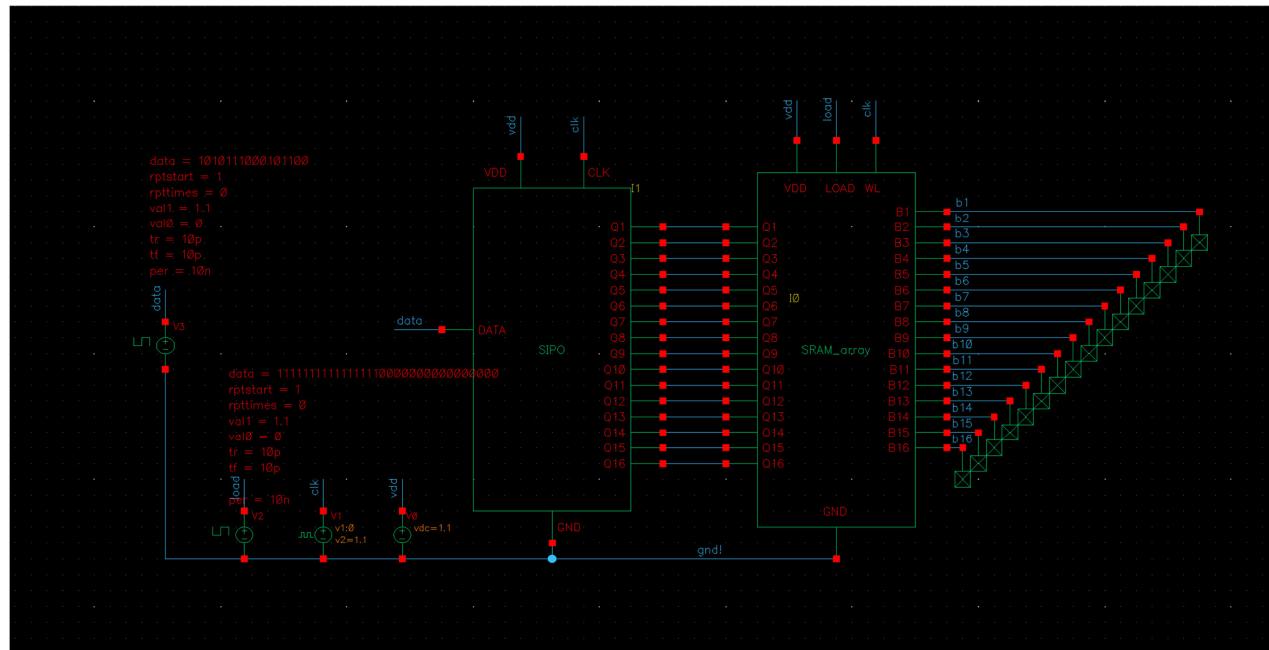


Figure41. Test Schematic for SRAM Array

Test Cases:

To validate the logic correctness of the clock frequency divider, we use a clock signal as input and check whether the output signal is a half frequency clock signal compared to input.

- Data input: 16bits (1010 1110 0010 1100)
- CLK input: 10ns period, 5ns pulse width clock signal with rise/fall time 10ps
- CLK input: 160ns period, 80ns pulse width clock signal with rise/fall time 10ps

Simulation Result:

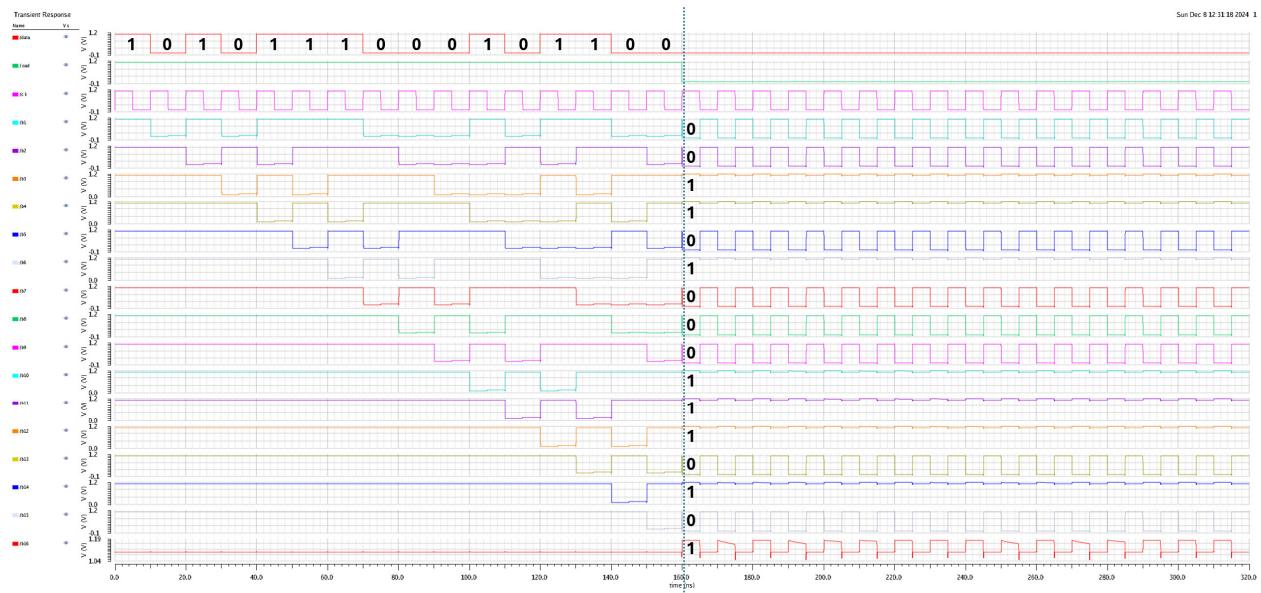


Figure42. Simulation Result for SARM Array

6. Clock Frequency Divider Validation

Schematic:

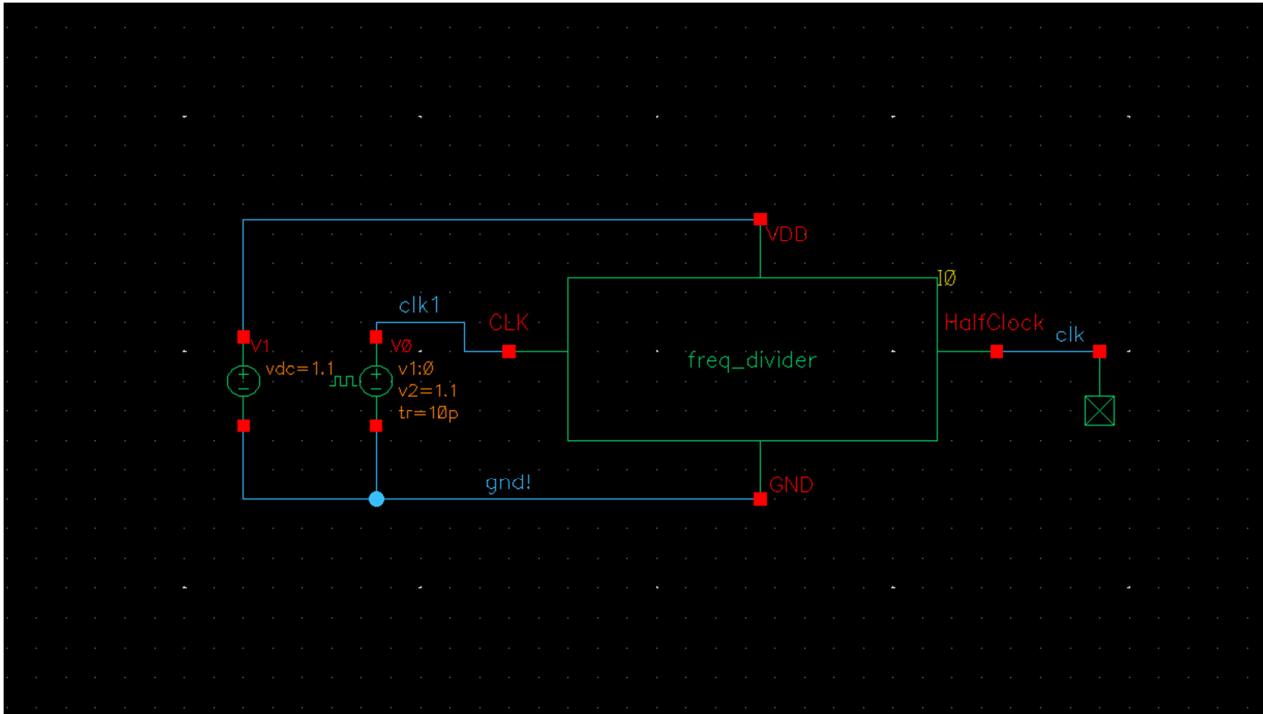


Figure43. Test Schematic for Frequency Divider

Test Cases:

To validate the logic correctness of the clock frequency divider, we use a clock signal as input and check whether the output signal is a half frequency clock signal compared to input.

- CLK1 input: 20ns period, 10ns pulse width clock signal with rise/fall time 10ps

Expected output signal:

- CLK output: 10ns period, 5ns pulse width clock signal with rise/fall time 10ps

Simulation Result:

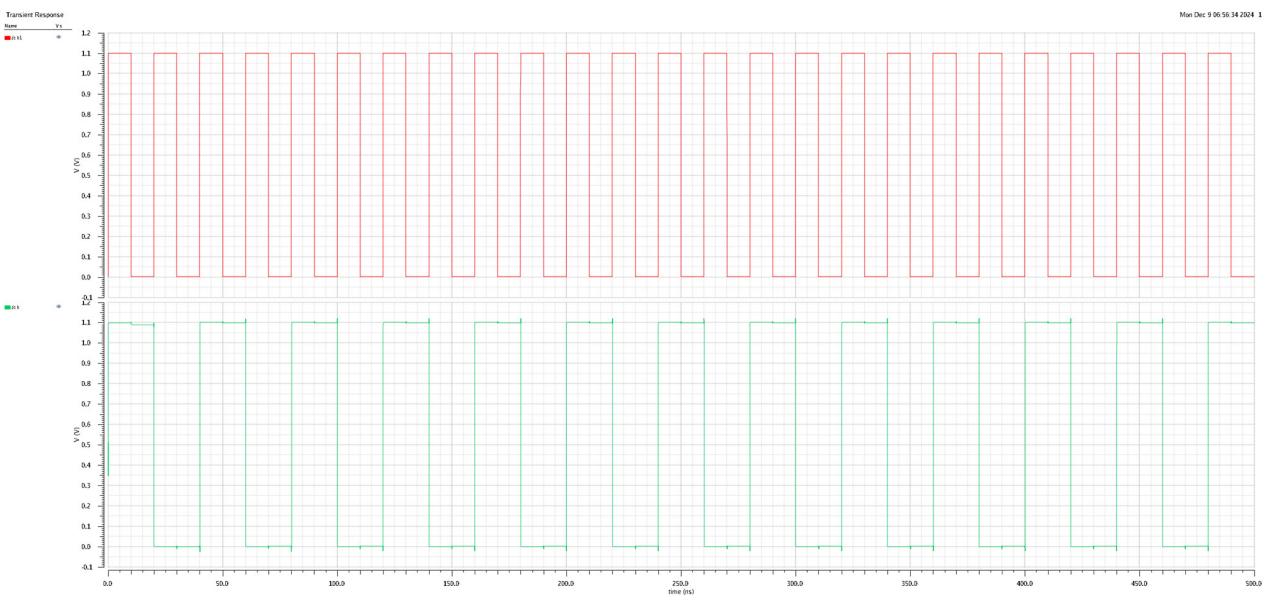


Figure44. Simulation Result for Frequency Divider

IV. Design Metric Test Cases

1. Maximum operating frequency

Maximum operating frequency represents the largest clock frequency (smallest clock period) of CLB to ensure the correct operation of the CLB cell logic. Therefore, we set the CLK time as variable T_{clk} and assign all other input's period as integer times of T_{clk} .

Notify that our clock input is the CLK of the SRAM, we set the high pulse width of DATA signal (T_{data}) and high pulse width of LOAD signal (T_{load}) to be T_{clk} , four address line I_3 to I_0 as $2T_{clk}$, $4T_{clk}$, $8T_{clk}$, $16T_{clk}$ respectively to include the every cases including the worst case delay to testing the max T_{clk} . Then we sweep the variable T_{clk} to find the least T_{clk} value while the output is not distorted.

Test Cases:

- T_{clk} sweep from 0.1ns to 20ns with the total number of steps of 20
- Reduce the sweep range to find the optimal operating frequency
- Four address line for LUT: I_3 to I_0 set as $2T_{clk}$, $4T_{clk}$, $8T_{clk}$, $16T_{clk}$ respectively

2. Average Energy

We have provided the average energy calculation equation:

$$\text{Average energy} = 0.2 \cdot \text{loading energy} + 0.8 \cdot \text{active energy}$$

Test Cases:

- The test transient simulation should last for at least 32 bits long time.
- At the first 16 bits long time, we should write 16 bits data to the CLB and calculate the maximum loading energy (which is supposed to be write all 16 ones to the CLB).
- In the second half, we should set read from the CLB while set the address line from 0000 to 1111 to iterate over all cases.

3. Area

We should calculate area with the data from the sized design using given formula.

$$area_{cell} = \sum_{i \in Tx} W_i$$

$$area_{2:1\text{ mux}} = \sum_{j \in Tj} W_j$$

$$area = 15 \cdot area_{2:1\text{ mux}} + 16 \cdot area_{cell}$$

Test Cases:

- Sized transistor should maintain correct performance of CLB circuit.

4. FOM

By definition:

$$FOM = \text{area} \cdot \text{averageEnergy} \cdot \frac{1}{\text{maxFrequency}}$$

We should calculate FOM using the design parameter and test results based on the test cases above.

V. Design Metrics

1. Maximum Switching Frequency

Parameter analysis:

We sweep T_{clk} from 20ns to 2ns and find that even at 2ns, the output is still correct. Therefore, we keep trying lower T_{clk} until reach the limit of 990ps. This is the lowest T_{clk} the CLB can accept while ensuring the output data is same as the input data with one-bit period shift.

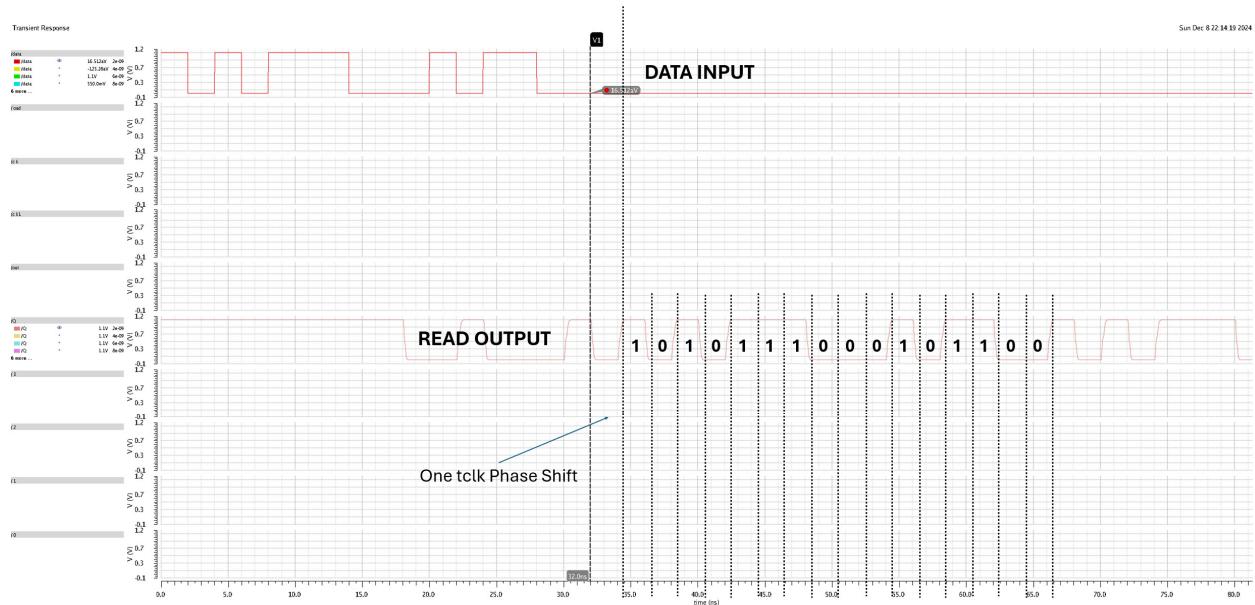


Figure45. Parameter Analysis Result of T_{clk}

Final Simulation result:

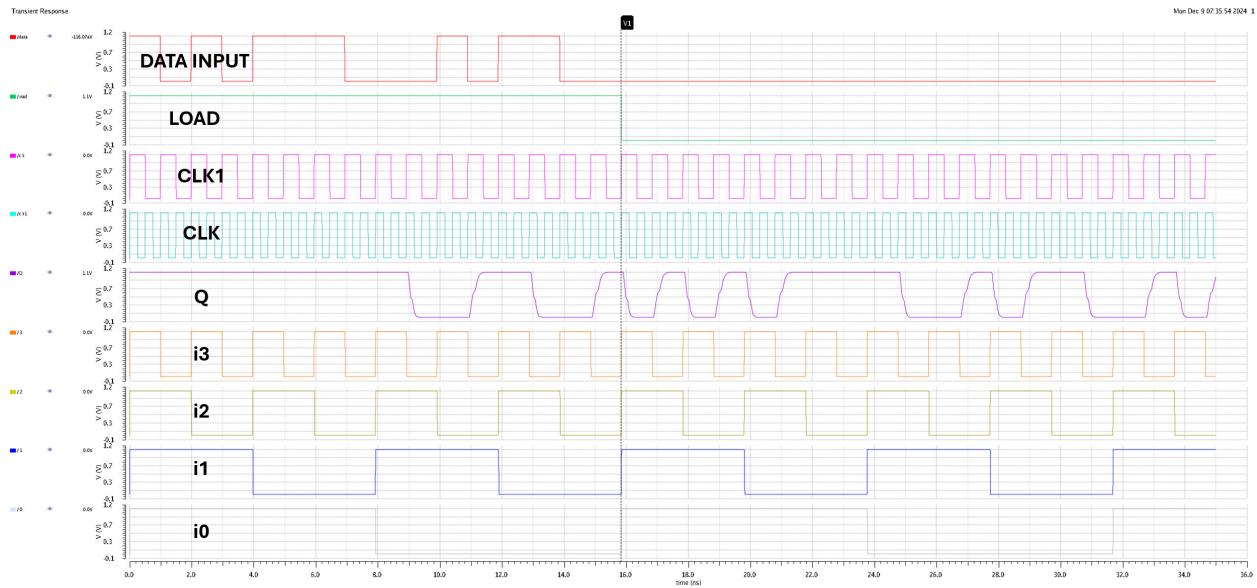


Figure46. Output pattern when $T_{clk} = 990\text{ps}$

Therefore,

$$\min T_{clk} = 990 \text{ ps}$$

$$\max Freq = 1.01 \text{ GHz}$$

2. Average Energy

Simulation result:

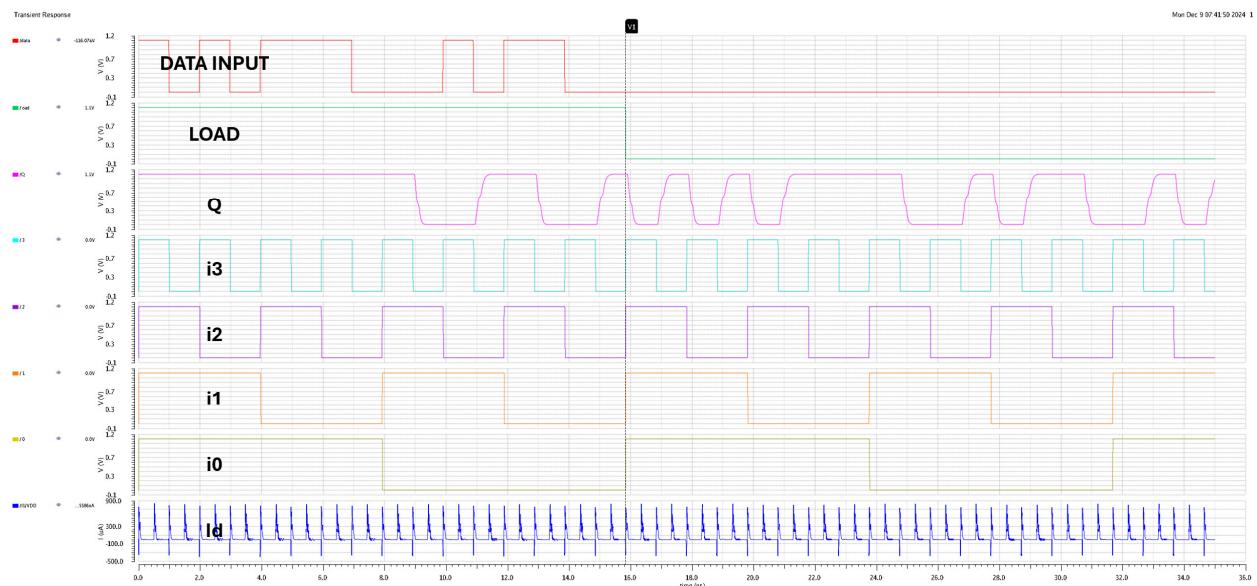


Figure47. Output pattern at maximum switching frequency

Calculation result:

Virtuoso (R) Visualization & Analysis XL Table	
File Edit View Tools Help	
Expression	Value
<code>1.1 * (0.2 * integ(i("/I1/VDD" ?result "tran") On 16n) + 0.8 * integ(i("/I1/VDD" ?result "tran") 16n 33n))</code>	881.2E-15

Figure48. Average Energy calculation result

$$\begin{aligned} \text{Average energy} &= 0.2 \cdot \text{loading energy} + 0.8 \cdot \text{active energy} \\ &= 881.2 \times 10^{-15} \text{ J} \end{aligned}$$

3. Area

Sized SRAM cell:

$$area_{cell} = 14 \times 120\text{nm} + 4 \times 360\text{nm} = 3.12 \mu\text{m}^2$$

Sized LUT cell (all minimum size):

$$area_{2:1mux} = 6 \times 120\text{nm} = 0.72 \mu\text{m}^2$$

Total area:

$$\begin{aligned} area &= 15 area_{2:1mux} + 16 area_{cell} \\ &= 15 \times 0.72 \mu\text{m}^2 + 16 \times 3.12 \mu\text{m}^2 \\ &= 60.72 \mu\text{m}^2 \end{aligned}$$

4. FOM

$$\begin{aligned} FOM &= area \cdot averageEnergy \cdot \frac{1}{maxFrequency} \\ &= \frac{60.72 \mu\text{m} \times 881.2 \times 10^{-15} \text{ J}}{1.01 \text{ GHz}} \\ &= 5.2977 \times 10^{-19} \mu\text{m}^2 \cdot \frac{\text{J}}{\text{Hz}} \end{aligned}$$

$$= 5.2977 \times 10^{-25} m^2 \cdot J/Hz$$

5. Summary table

Table5. Summary Table for CLB Performance

Max Freq	Average Energy	Area	FOM
1.01 GHz	$8.812 \times 10^{-13} J$	$60.72 \mu m^2$	$5.2977 \times 10^{-25} m \cdot J/Hz$

VI. Design Process Writeup

1. SRAM Optimization

1.1 Stability

First of all, the most important optimization is sizing the write circuit in SRAM cell. The reason is that it will affect the writing function of SRAM. To be specific, pre-charging circuit could be on when loading. If we are loading 1, it's ok because we need upper 2 PMOS on to pull the bit line high. However, when we are loading 0, Both of the pull-up and pull-down network is on, which will cause bit line to be a floating point, as shown in Figure 49. In order to solve this, we size the NMOS in writing circuit as 3Wmin to increase its pull-down ability. Therefore, we solve this problem without adding any other topology, which may cost extra area & energy.

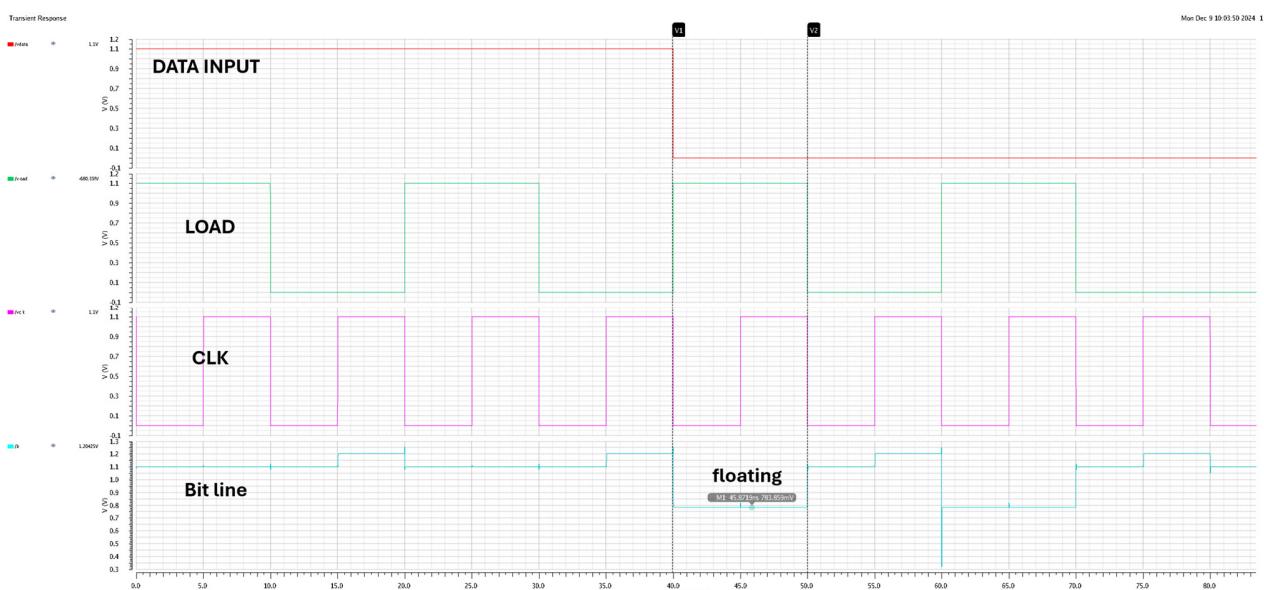


Figure49. Output by low pull-down ability NMOS

1.2 Delay/switching speed

The read/write speed of SRAM largely depends on the size of 2 pass logic NMOS controlled by word line (WL/CLK). If we use low width NMOS, it will bring lower V_{th} , which makes it open faster (the 2 NMOS are activated and allow Q to be read & write at lower voltage) Through cadence parameter analysis, we get the same trend. Therefore, the size of 2 pass logic NMOS should be minimum size. To increase the read stability, we can also size the 2 other NMOS in 6T NMOS to be larger, which can also pull the signal quicker. However, according to simulation, benefits of increasing width is lower than remain min size from the angle of total FOM calculation (trade off between frequency & area). Therefore, all the MOS in 6T-SRAM, except for 4 NMOS in writing circuit is min size.

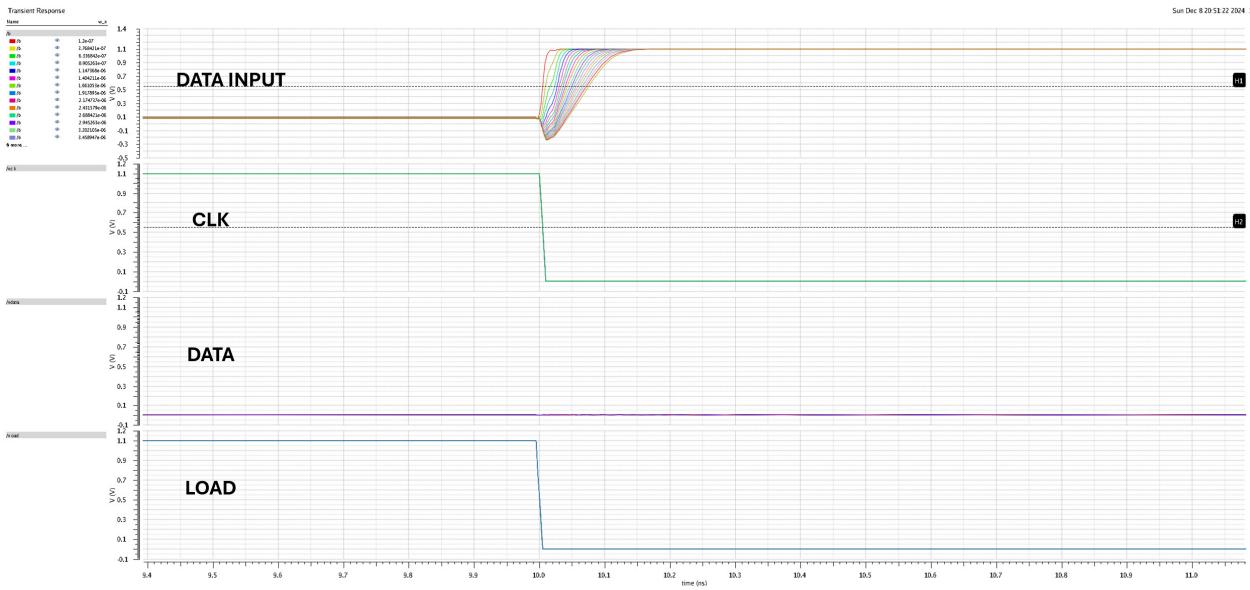


Figure 50. Sizing pass logic NMOS in 6T SRAM

2. LUT Optimization

In general, the delay of LUT will affect the maximum switching frequency. When switching speed of address line is too fast, the selection bit will change before the output is correctly set. To reduce the delay of LUT, size pass logic NMOS's width is the most efficient way. Because of the El'More delay model, the pass logic NMOS is on the delay's critical path. The sweep results are shown in Figure 51. We can find that at around 3.5 um, delay is almost fixed. If we consider single iMUX/LUT, we should choose 3.5 um width. After calculating the whole CLB, min size is still the optimal choice.

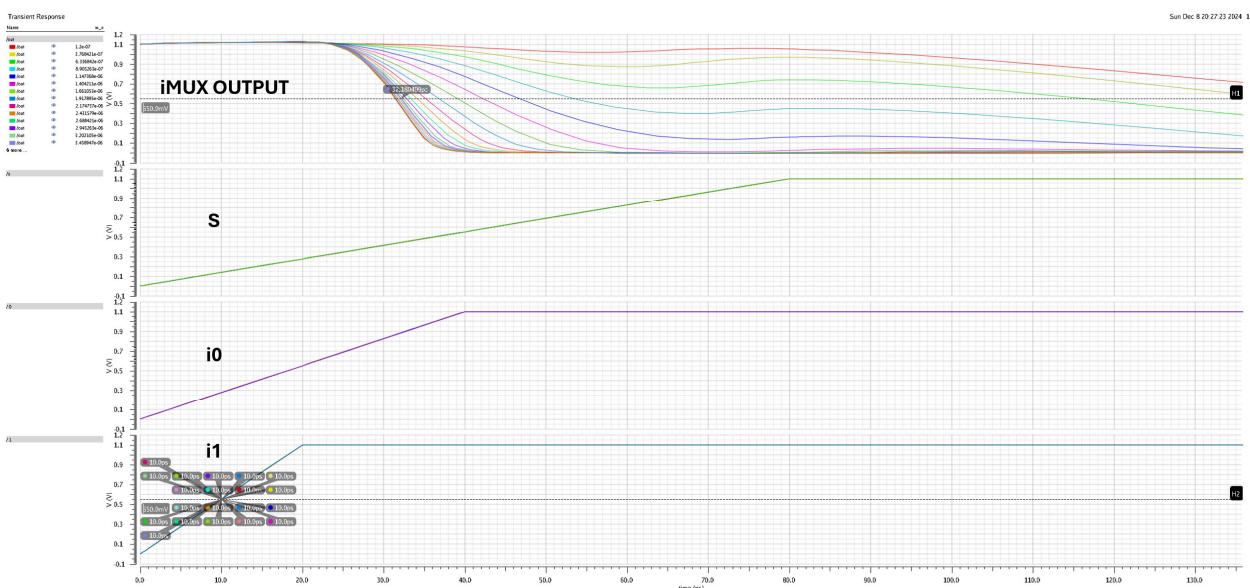


Figure 51. Sizing Pass logic NMOS size in iMUX

3. Pre-charge signal generation circuit

Shared pre-charge signal generation circuit could be used to decrease total area in our design. The reason is that our CLB is designed to control 1 word 16 bits, which means we only have a single word line thus every bit write/read at the same time. However, if the load of the pre-charge circuit is too large, we need to size the MOS in this circuit to make sure it can drive. After testing, if all 16 SRAM cells share one min size pre-charge circuit, the maximum frequency is around 600MHz. By calculating FOM, use separate pre-charging circuit will bring better FOM performance because pre-charge circuit's area is relatively small compared to other parts.

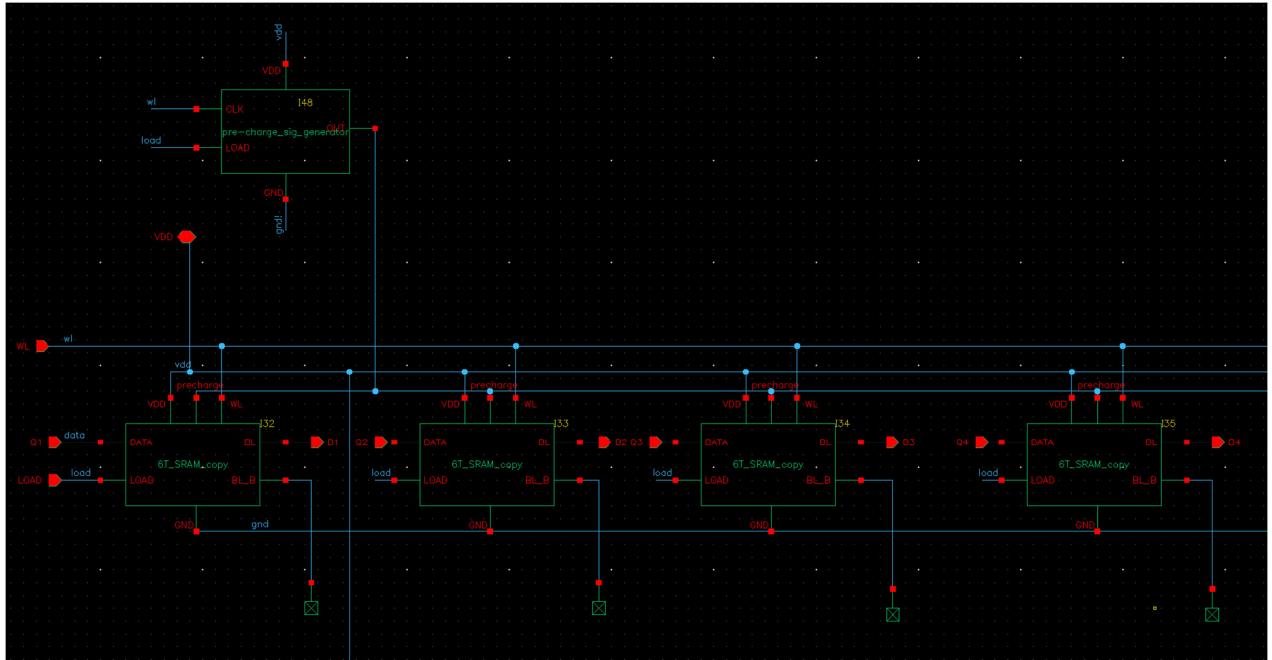


Figure 52. Shared pre-charge circuit

4. Double flipflop to regulate output timing (perfectly one-bit shift)

D-flipflop in the last stage perform as output control. It will configure the CLB output timing based on system clock. However, because output of the LUT has delay, we use a double frequency clock instead to detect. The benefit of this is that we could get a perfectly one-bit shift output, as shown in Figure 53. When using the output of CLB as input of another logic block in the same system, it is easy to figure out the timing.

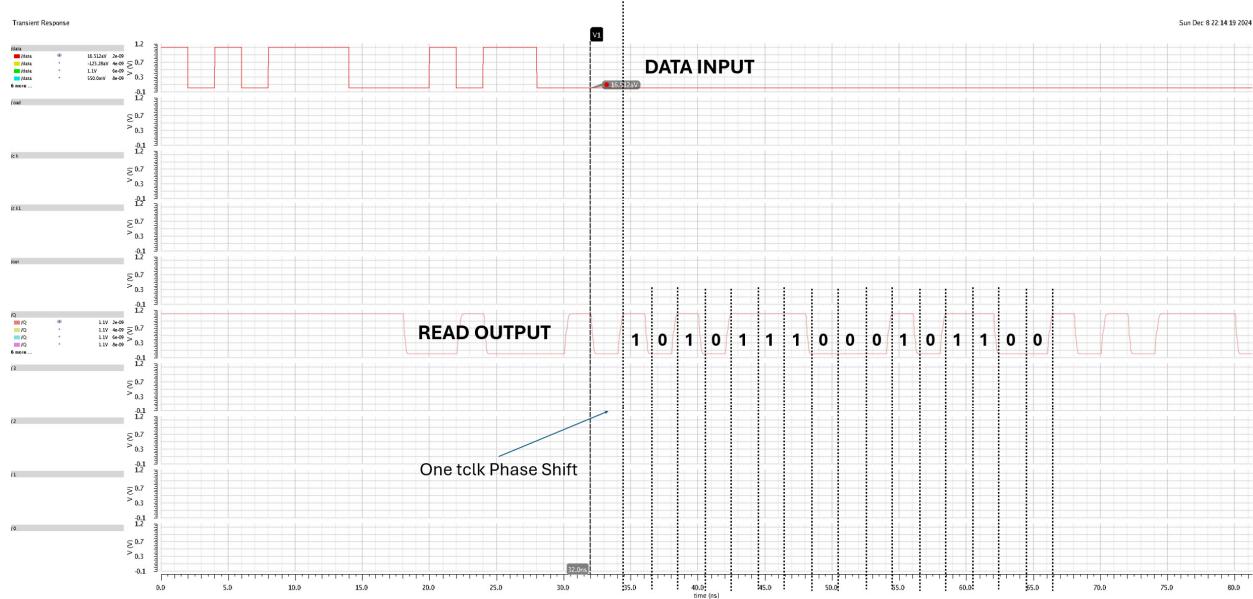


Figure 53. One bit shift pattern

However, it does require a little or even no phase shift clock with double/half frequency. Any large Clock shift will cause detection failure at rising edge. Our system using a simple D-Flipflop based frequency divider, which could bring some phase shift. To further improve, Phase Lock Loop (PLL) could be used to provide double/half frequency clock with no phase shift, which will increase the system's robustness.