ECE482 Fall 2024
Final Project Report
Group13

Ying-Ching Lee,

Jay Nathan,

Eric Tang,

Phuvit Thirasuntrakul,

Jie Yan

i. & ii. The overall design approach & transistor-level size, schematics, layout

In this project, we designed an image comparator that compares two images using CMOS technology, including schematic, layout and the test. We also designed a circuit to send the comparison result to the off chip, making the whole design more robust and realistic. Utilizing the testbench provided, we successfully verified the functionality of our design, and the clock frequency can go up to 2.5 GHz.

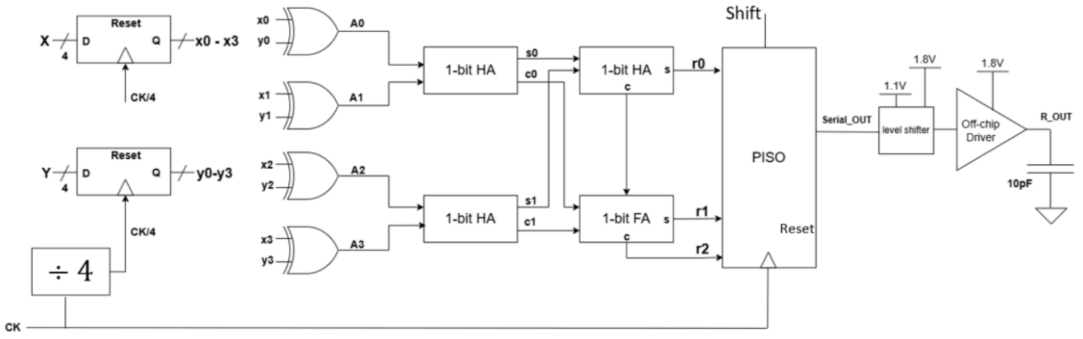


Figure 1 block diagram of our design

Following the general image comparator design block diagram, we designed each part separately and by adding instances of each part to the final design and connecting them together, we can complete the whole system. Below showed the schematic of our final circuit, which includes clock inverter, D register, frequency divider, XOR gate, half adder, full adder, PISO, level shifter and a supper buffer (off-chip driver)

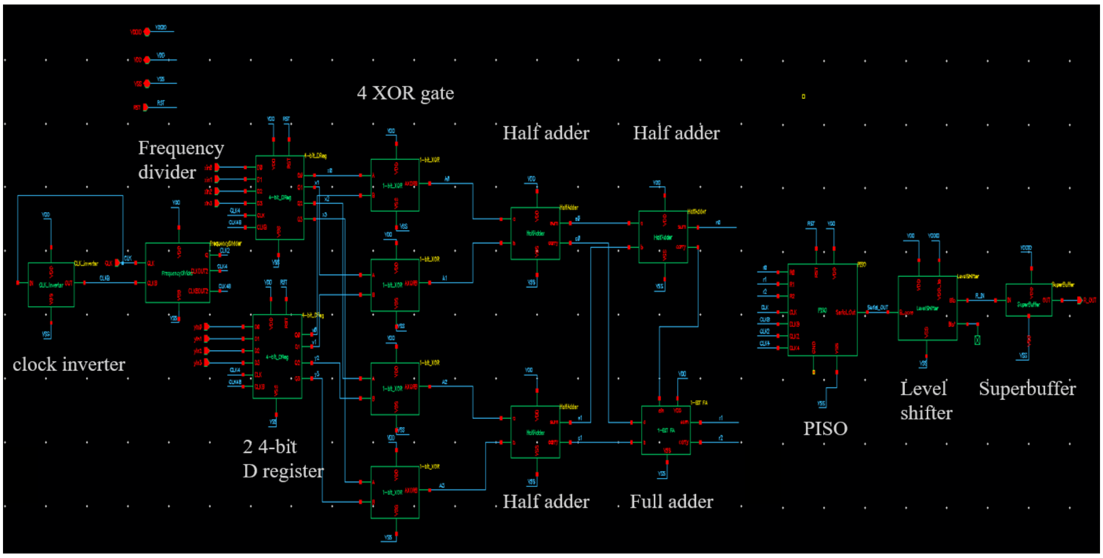


Figure 2. Cadence schematic of the image comparator circuit

We put on efforts to reduce the power, area and delay of the system. To reduce the number of transistors and thus reduce the size of the whole design, we use the design that with fewer numbers of transistors such as mirror-adder and engineer the truth table/logical operations to reduce the number of transistors. We also tried to share more diffusion regions. For example, when switching from 1-bit D registers to 4-bit D registers, instead of copying 1-bit D registers layout four times, we rebuild the layout and reconsider the body location. Considering the delay of the system, we sized our design properly to make sure the PMOS and NMOS have the same capabilities of drawing current (beta=2). Some designs are also favored to reduce the delay such as the mirror-adder. More details regarding efforts in improving the performance of the design can be found in the description of each sub design.

Clock inverter

The CLK inverter is needed to create the CLKB signal, which will need to be used in the register blocks. An FO3 sizing is chosen, since the CLK will be driving a few things within the circuit, like the frequency divider and D registers. Fig. 3 shows the Cadence schematic and the Cadence layout of the CLK inverter, respectively. Notice that the layout of the CLK inverter is designed such that the spacing is minimized, resulting in a compact and neat layout. Table 1 shows the transistor sizings used in the CLK inverter circuit.

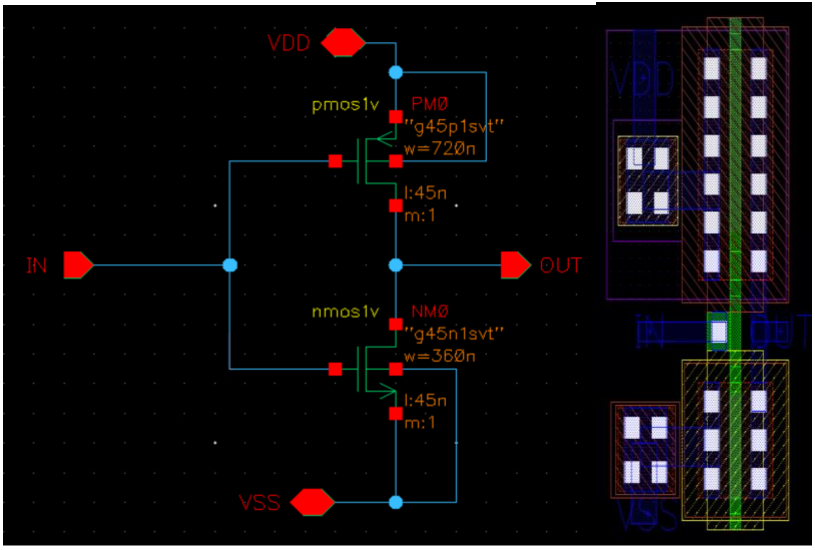


Figure 3. Cadence Schematic and layout of the CLK inverter circuit

Table 1. Transistor sizes of the CLK inverter circuit

Name (as labeled in schematic)	Length (nm)	Width (nm)
PM0	45	720
NM0	45	360

D register

Table 2: Transistor Sizes of the D register

Type	Length (nm)	Width (nm)
PMOS	45	240
NMOS	45	120

(1) 1-bit D register

We used C2mos design to implement D register. We first draw the schematic for the 1-bit D register which used the course note as the reference. After drawing the schematic, we used Euler's Path to share diffusion region and improve the layout's size. From the documentation from the class, we know that all of the devices should have a beta of 2. We first tested cases when we used nmos's width to be 240 nm (since this gives an equivalent width of 120 nm), and pmos of 480 nm. The test passed successfully; thus we tested nmos's width with 120 nm which also worked. Also, since the D register operates at CLK/4, and it's function is to store the value. This implies that we don't need the fastest design, thus a smaller sizing should work too.

To achieve the reset function, we utilized a pull-up network at node X and a pull-down network at node Q.

We had considered other designs for pull-up and pull-down networks which is by manipulating the input node at D and CLK. In short, we add two switches (transmission gate operated by RST) at the input of D and CLK, then we add one pull-down nmos that connects D and CLK which is operated by the signal RST. The advantage is there will be a less current draw if the reset stays high/when giving a longer RST signal, it can reduce the current flowing down inside the transistor. However, this design will be bigger than the original design (1 more transistor), and provide marginal gain to the power consumption since the RST signal is generally short. It is also very challenging to incorporate at the layout level, thus we didn't implement it.

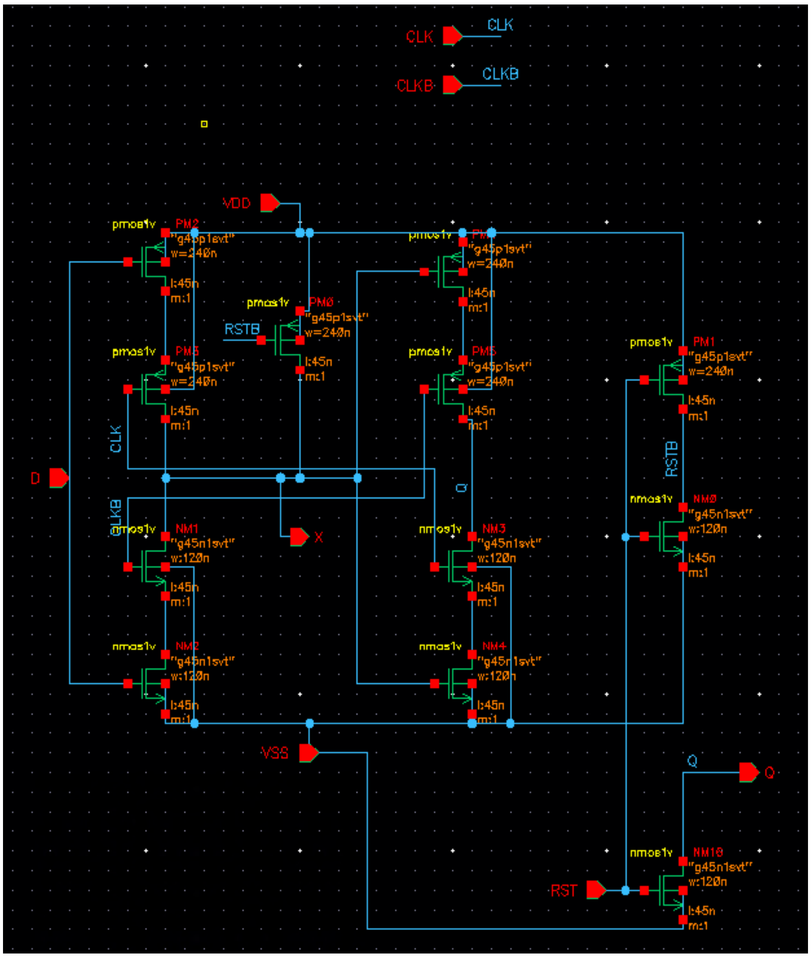


Figure 4. 1-bit D register schematic

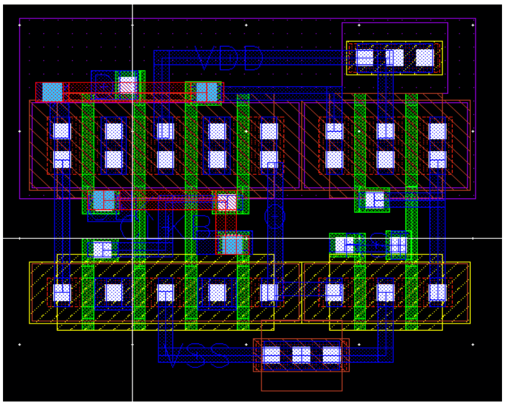


Figure 5. 1-bit D register layout

(2) 4-bit D register

For the 4-bit D register, the schematic is essentially the same, but since we can optimize the sizing since we only need one reset signal for the 4-bit register, which means we only need one inverter to generate the inversed reset signal. Due to the changes, more diffusion regions can be shared and we can reconsider the body location. We decided to redraw the layout and schematic.

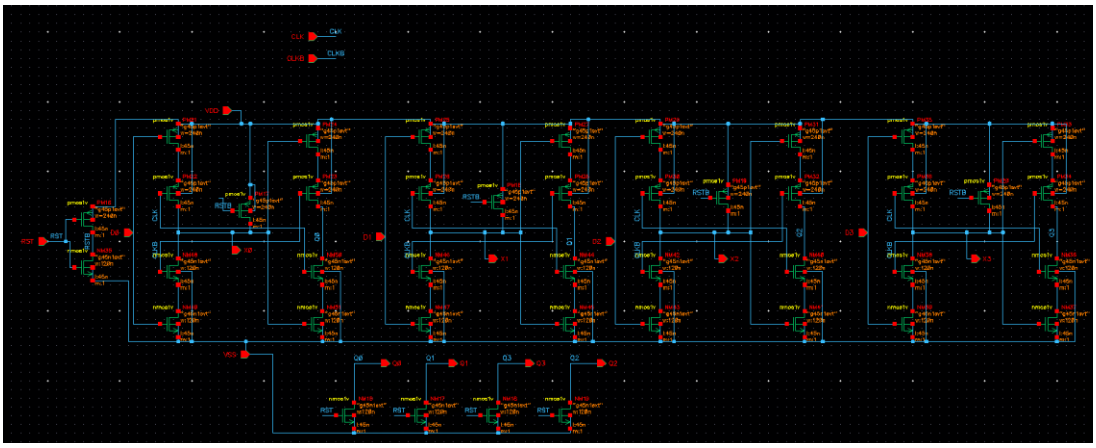


Figure 6. 4-bit D register schematic

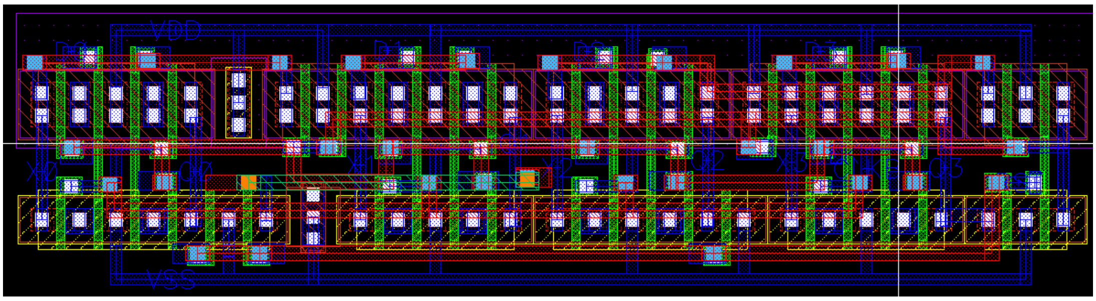


Figure 7. 4-bit D register layout

Frequency divider

We utilized two D registers to build the frequency divider. To achieve that, we connect the inversed output front of the registers to the input of a register, and we connect the CLK into the

CLK input of the register. One of the devices described above will provide a CLK/2 signal at the output. If we cascade two of them, we will have a CLK/4.

We spent some effort to maximize the circuit of the frequency divider. Since the component will keep operating, we removed the reset function from the circuit, which saves the area of the layout.

We also first had some issues with bigger clock skew and slow setup time for the first CLK cycle, as shown by the figure below of the simulated output. CLK is the input signal and CLKout1 is the output without any buffer.

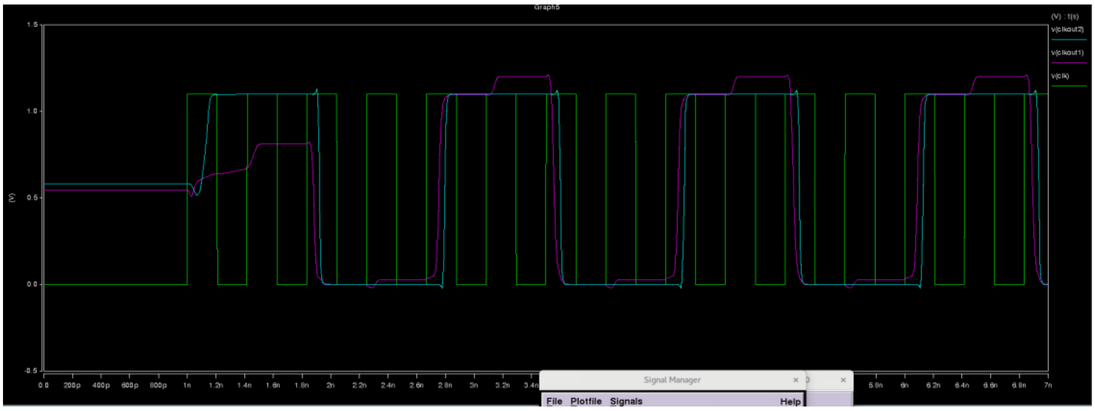


Figure 8. first frequency divider design with bigger clock skew and slow setup time

We suspected that it was caused by the second-stage sizing is too small. To solve this problem, we had two solutions, the first solution is to increase the sizing of the second D register of the circuit, and the second solution is to add some buffers. We tested both and had a better result with the one that added some buffer to it. It achieved a clock skew of 9.6ps and has relatively sharp edges, with the sacrifice of some extra delay compared to the original clock.

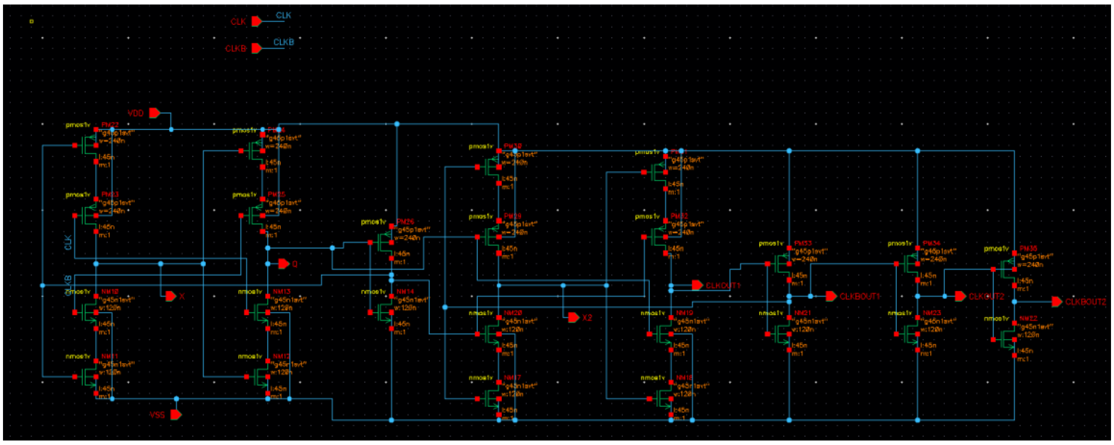


Figure 9. Frequency divider schematics

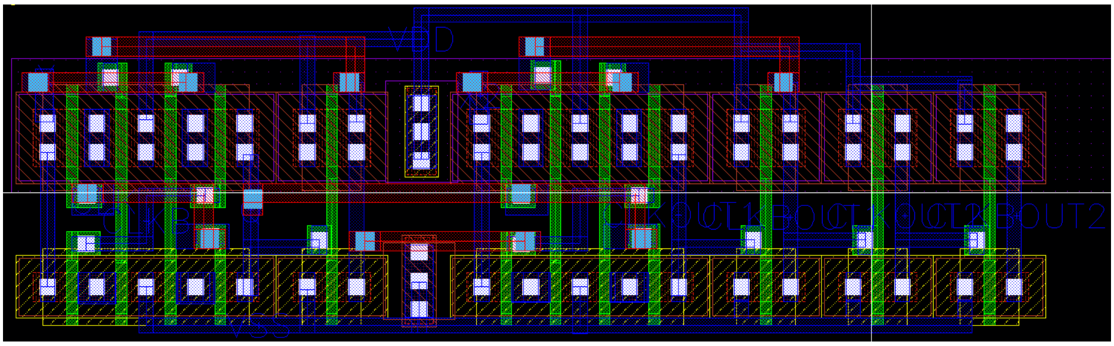


Figure 10. Frequency divider layout

Table 3: Transistor Sizes of the Frequency divider

Type	Length (nm)	Width (nm)
PMOS	45	240
NMOS	45	120

XOR gate

The design uses complementary static CMOS logic. With 6 PMOS and 6 NMOS, the gate uses two CMOS inverters to generate A’ and B’, then feed in to input ports of main XOR gate. The main XOR gate circuit is designed as a combination of pull-up and pull-down networks, connected to four types of inputs, A, B, A’ and B’, forming the required XOR logic function.

Boolean expression of XOR: A⊕B=(A·B’) +(A’·B)

Thus, for the pull-up network, A and B’ are connected in series, B and A’ are also connected in series, and finally, the two branches are connected in parallel. For the pull-down network, A and B are connected in series, B’ and A’ are also connected in series, and the two branches are connected in parallel.

The CMOS approach minimizes static power since there is no direct path between supplier and ground when the output is in a steady state. Using 8 transistors in the main circuit keeps the design compact with steady functionality. Properly sizing pull-up and pull-down networks can ensure reasonable small rise and fall time, reducing propagation delay.

We choose beta to be 2 for both the inverter and the XOR logic and S to be 2 for inverters and 1 for XOR gate.

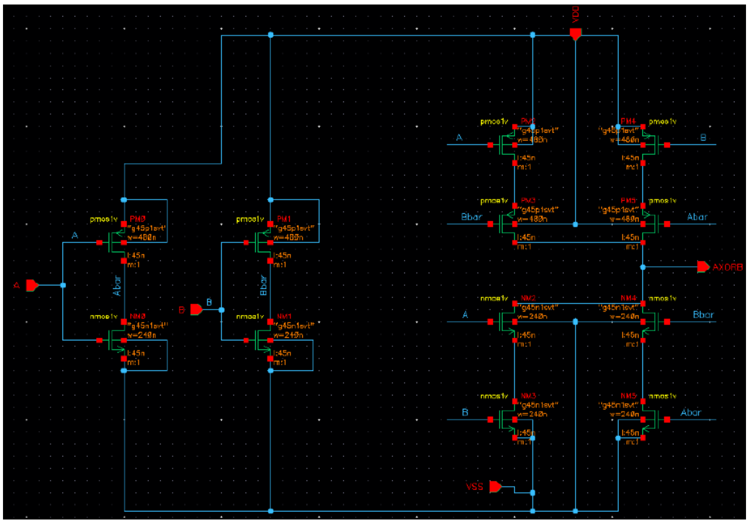
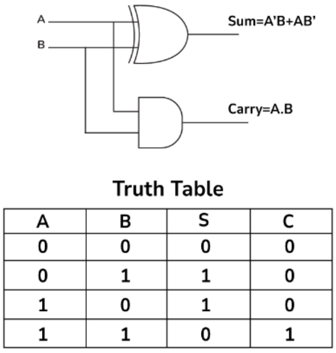


Figure 11 schematic of the XOR gate

Table 4: Transistor Sizes of the XOR gate

Type	Name(s)	Length (nm)	Width (nm)
PMOS	PM0, PM1, PM2, PM3, PM4, PM5	45	480
NMOS	NM0, NM1, NM2, NM3, NM4, NM5	45	240

Half adder



Truth Table

A	B	S	C
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

Figure 12. Half adder logic level implementation and truth table

Half Adder contains two input A and B and two output Sum and Carry. After writing down the truth table, we find that the sum bit can be implemented using an XOR gate and a Carry bit can be implemented using an AND gate. The final design contains a total of 14 transistors and is a complementary logic. To design an AND gate, we make a NAND gate plus and inverter. To design the XOR gate, the sum is written as $Sum = \overline{A + B + AB} = (A + B)(\bar{A} + \bar{B}) = A\bar{B} + \bar{A}B$, which reduced the number of transistors by 2 compared with using $A\bar{B} + \bar{A}B$

complementary logic to build the gate.

The circuit is sized to be S=1 Beta=2 considering the worst-case scenario.

To design the layout and save space, we use fingers that can share the diffusion region.

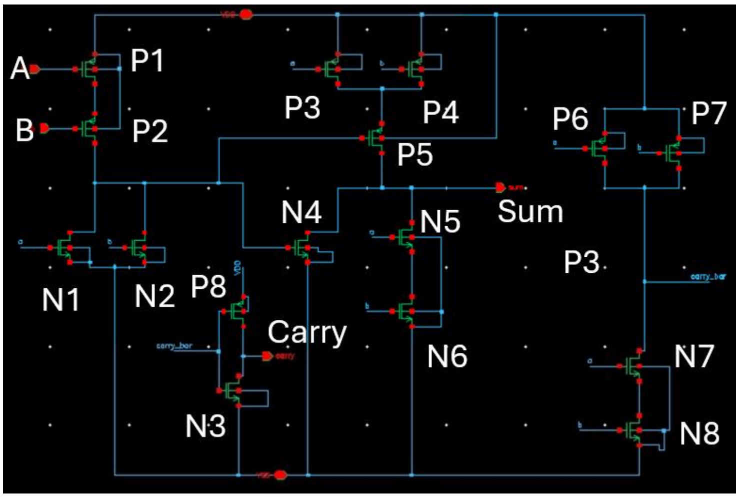


Figure 13. Schematic for the half adder with input and output pins indicated

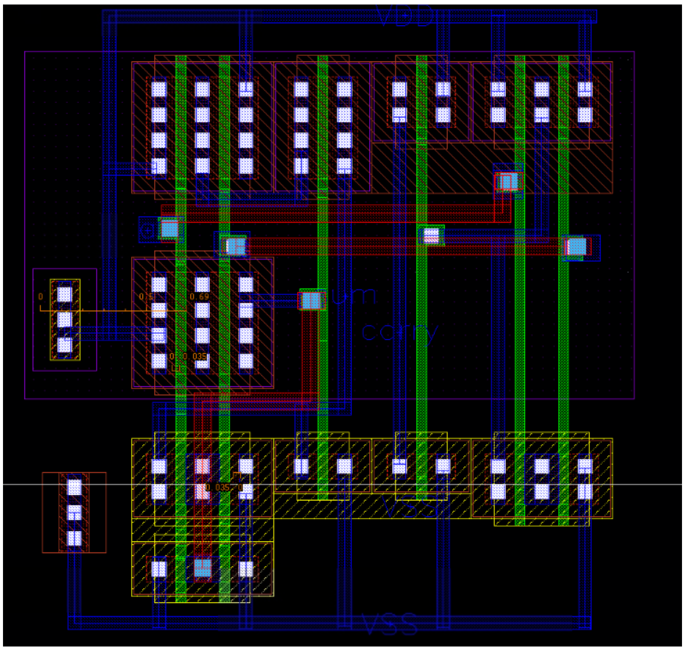


Figure 14. Layout of the half adder

Table 5: Transistor Sizes of the Half adder

Type	Name(s)	Length (nm)	Width (nm)
PMOS	P1, P2, P3, P4, P5	45	480
PMOS	P6, P7, P8	45	240

NMOS	N1, N2, N3, N4	45	120
NMOS	N5, N6, N7, N8	45	240

Full adder

A “Mirror Adder” design, as presented in the lecture, is used for the design of the full adder, due to its superiority over the complementary static CMOS design which requires a lot of transistors being connected in series, affecting its delay, area, and power. A Full Adder that uses the “bridge” design mentioned in [1] was also originally considered due to its superior delay compared to the “mirror” design of the Full Adder. However, due to the high transistor count of the “bridge” design [1], the Full Adder using the “mirror” design is chosen, to balance between delay, power, and area.

Fig. 15 shows the Cadence schematic of the Full Adder circuit used in this project. An FO1 inverter-equivalent sizing is used for each of the pull-up and pull-down networks, with β = 2. However, for the pull-up and pull-down networks that is used to calculate carry-out-not that has cin as input, an FO2 inverter-equivalent sizing is used, in order to minimize the delay of the critical path. This design approach is chosen because the input cin will most likely have the highest delay compared to other inputs (a and b), since the critical path will need to go through two half-adder circuits, compared to other inputs which go through only one half-adder circuit. Table 6 shows the sizes of the transistors used in the Full Adder circuit. Fig. 16 shows the Cadence layout of the Full Adder circuit. Metal 2 is used in some regions of the layout in order to create a neat and compact layout. Furthermore, the spacing between each part is minimized so that the compact is as compact and neat as possible. Lastly, multi-fingers design is used for some of the pull-up and pull-down networks in order to share the common diffusion region in order to reduce the required area even further, making the design incredibly compact and neat.

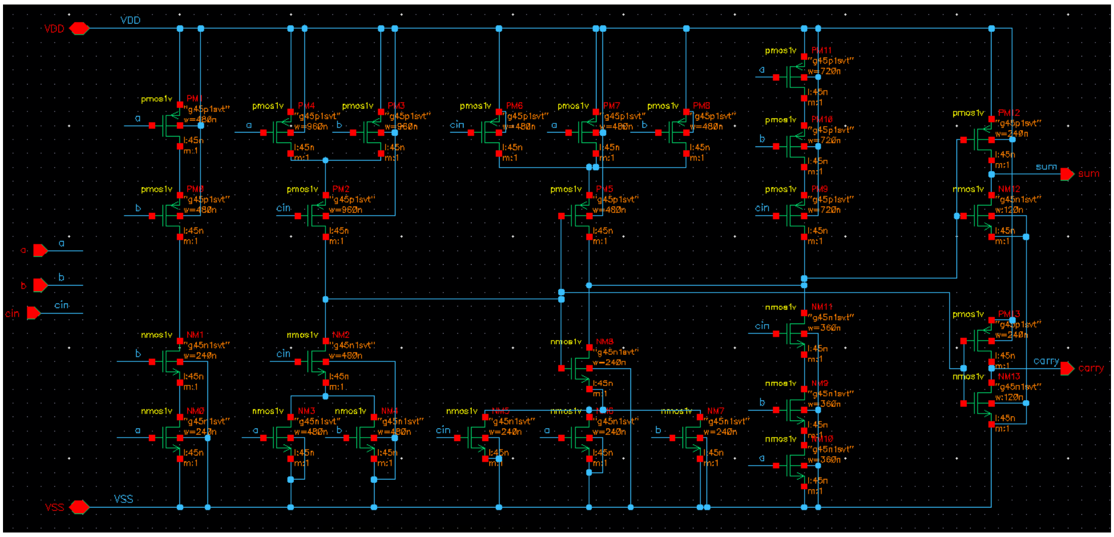


Figure 15. Cadence Schematic of the Full Adder

Table 6. Transistor Sizes of the Full Adder Circuit

Type	Name(s)	Length (nm)	Width (nm)
PMOS	PM12, PM13	45	240

	PM0, PM1, PM5, PM6, PM7, PM8	45	480
	PM9, PM10, PM11	45	720
	PM2, PM3, PM4	45	960
NMOS	NM12, NM13	45	120
	NM0, NM1, NM5, NM6, NM7, NM8	45	240
	NM9, NM10, NM11	45	360
	NM2, NM3, NM4	45	480

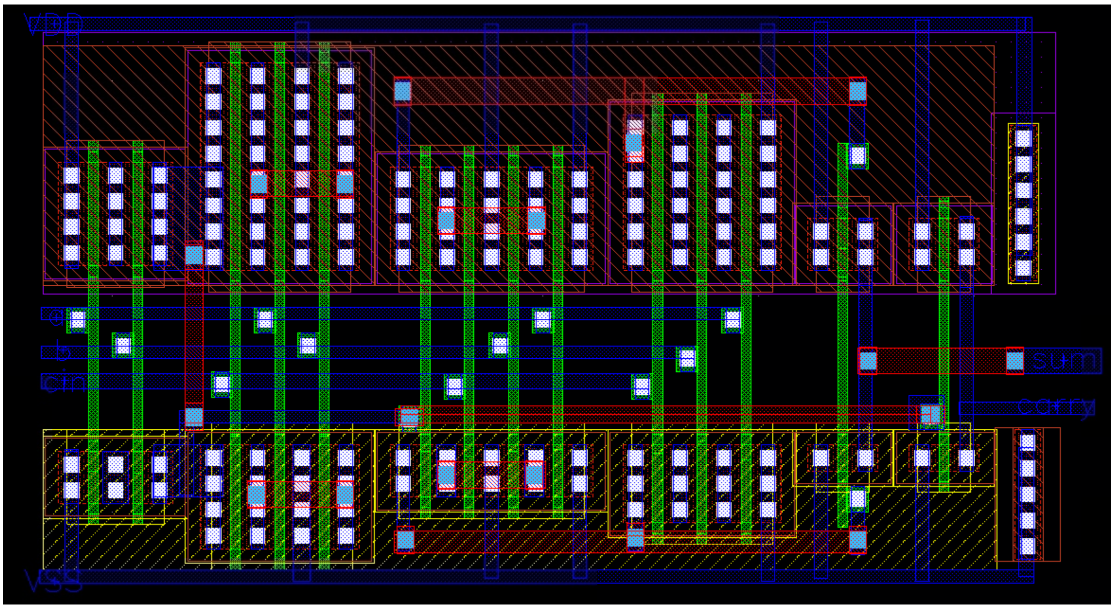


Figure 16. Cadence Schematic of the Full Adder

PISO

The Parallel-In Serial-Out (PISO) shift register is designed to convert parallel data into a serial data stream. This functionality is particularly useful when multiple parallel data bits need to be transmitted sequentially over a single line, reducing the number of data lines needed. The PISO shift register achieves this by using control signals and a synchronized clock to load and shift data through stages of 2input muxes and 1-bit registers.

Three 2-input multiplexers determine the data flow into the three 1-bit registers. Each multiplexer has two data inputs and a control input (Shift). When Shift is low (0), the multiplexer selects the parallel input data from the full & half adders (r2, r1, r0) and loads it into the corresponding register. When Shift is high (1), the multiplexer selects the data output from the previous register, enabling the shifting operation. This dual-mode operation allows the register to switch seamlessly between parallel loading and serial shifting.

The three 1-bit registers form the core storage unit of the PISO shift register. Each register saves & holds a single bit of data, which is then either shifted to the next register or output sequentially, depending on the state of the control signals. The registers are driven by a clock signal (CLK), ensuring that data is shifted or loaded in sync with the clock edges. For example, the data in the first register (I3) shifts into the second register (I4), and the data in D1 shifts into the third register (I5), exiting as a serial line of data (Serial_out).

For design choices, as seen in the schematic and layout, multiple instantiations of key components like multiplexers and D registers were used to simplify the top level of the PISO. This makes it easier to not only implement but debug issues that occur. In the layout, the decision to map the inputs and output to the top and bottom of the PISO helped with instantiating the shift register with the rest of the circuit. Additionally, by grouping the 3 multiplexers and D registers together, lateral space was saved, allowing for space between the 2 regions of the PISO to be easily routed.

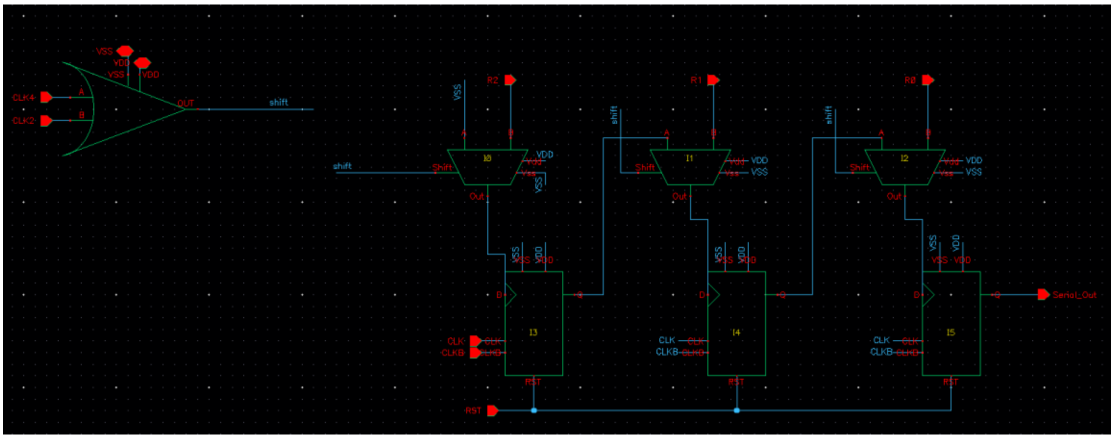


Figure 17 Cadence Schematic of the PISO Top Level

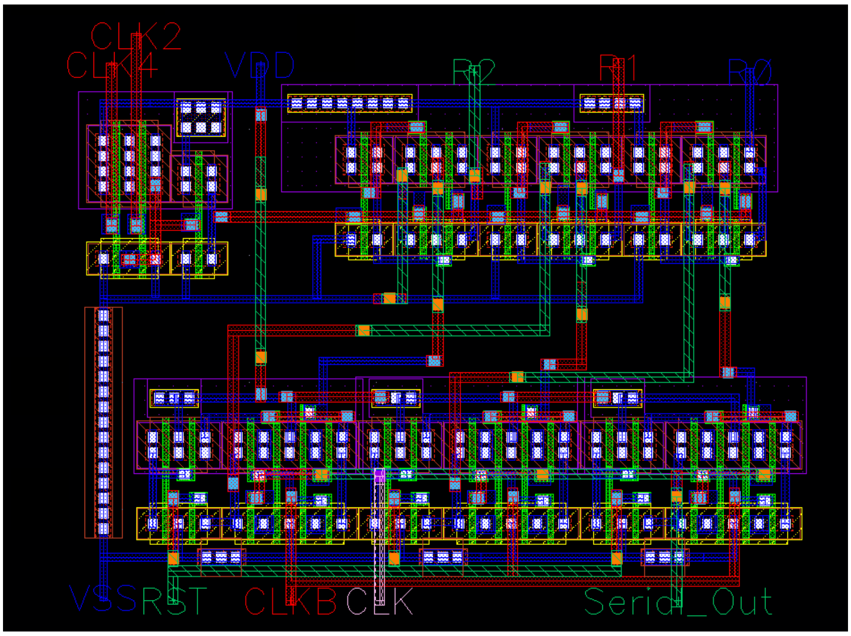


Figure 18. Cadence Layout of the PISO

(1) shift signal

As per the timing diagram, the select signal for the multiplexers is a 75% duty cycle. To generate this signal, from the Frequency Divider, an OR gate was constructed to combine both signals such that the Shift signal remails high for ¾ of the period and low for ¼ of the period. Additionally, the falling edge of clk/2, clk/4 remains high for half of the clk/4 period, extending the time that Shift is high. As for the OR gate, it is a NOR gate with its output inverted by a FO1 inverter with beta = 2. As for the NOR gate, the sizes of the pmos are 4 times larger than nmos to create enough drive strength. This is due to clk/2 and clk/4 having lower frequencies, meaning they need to drive higher capacitive loads.

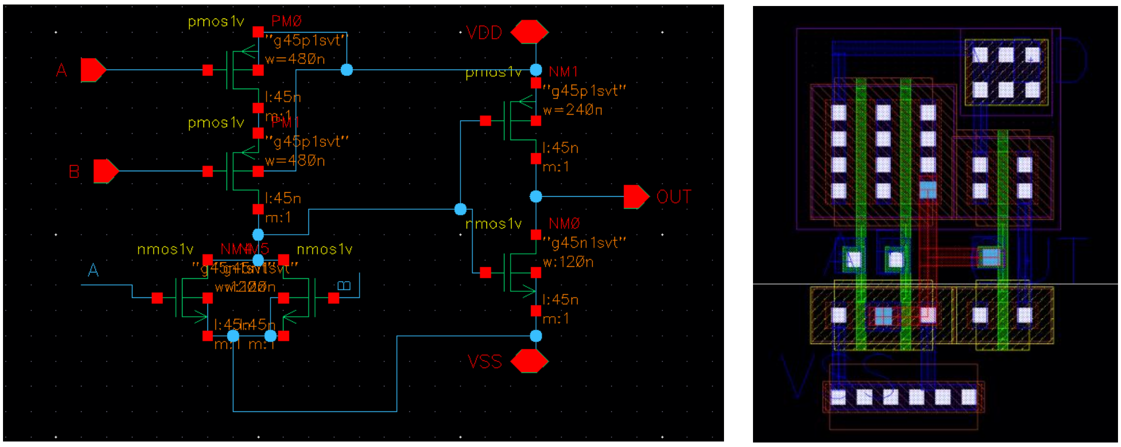


Figure 19. Cadence Schematic & Layout of the OR Gate

Table 7: Transistor Sizes Sorted by Width

Type	Name(s)	Length (nm)	Width (nm)
NMOS	NM1, NM2, NM3	45	120
PMOS	PM4	45	240
	PM0, PM5	45	480

(2) Multiplexer

As stated previously the 2 to 1 mux takes in two values and chooses between the two based on the Shift signal. This circuit follows static CMOS logic, comprised of two transmission gates, where the Shift signal is fed through a FO1 inverter with beta = 2. The idea of using transmission gates was a good idea, since they simplify the design and provide minimum delay when switching at high speeds. They also maintain signal strength preventing voltage degradation using rail-to-rail output. As for the layout, the use of metal2 allows us to bypass the metal1 connections, compacting the overall layout design as seen below.

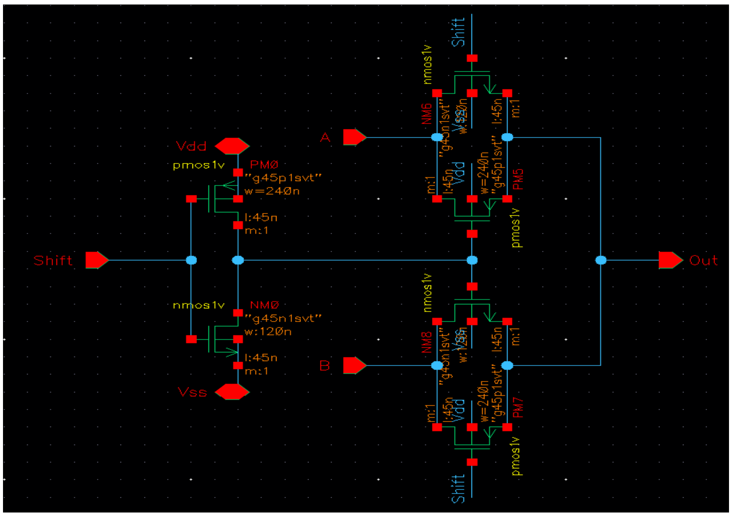


Figure 20. Cadence Schematic of the 2-to-1 Multiplexer

Table 8: Transistor Sizes Sorted by Width

Type	Name(s)	Length (nm)	Width (nm)
NMOS	NM0, NM6, NM8	45	120
PMOS	PM0, PM5, PM7	45	240

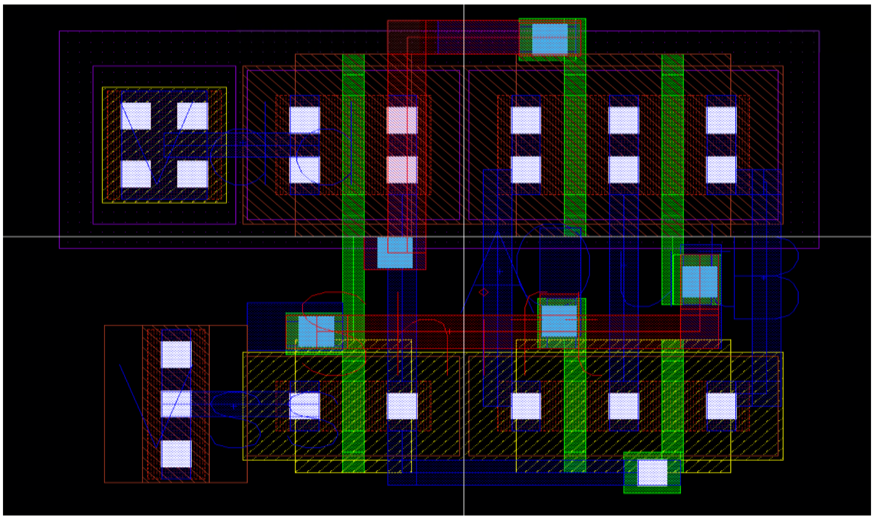


Figure 21. Cadence Layout of the 2-to-1 Multiplexer

Level shifter

The level shifter in the image comparator is used to bridge the voltages between the 1.1V logic (core circuit) and the 1.8V I/O interface (off chip driver). Its main function is to ensure compatibility between different components. The level shifter takes the 1.1V output from the PISO Shift Register (Serial_out) and drives it to 1.8V. The design of the circuit follows a static level shifter. This design is tailored to translate voltage signals from a low domain (1.1V) to a higher domain (1.8V).

In the top-level layout of the Image Comparator, 2 level shifters are used, to handle communication both ways between 1.1V and 1.8V domains. The first level shifter instantiated converts signals from the 1.1V core domain to the 1.8V I/O domain for driving external circuits. The second level shifter converts signals from the 1.8V I/0 to 1.1V core. This means that the sizes of the transistors within each level shifter are different, where the 1.1V to 1.8V level shifter has a ratio of 640nm:2320nm regarding PMOS/NMOS size. Since the I/O domain typically drives larger capacitive loads, larger transistor widths provide stronger drive currents overcoming capacitive loads and have a faster switching speed.

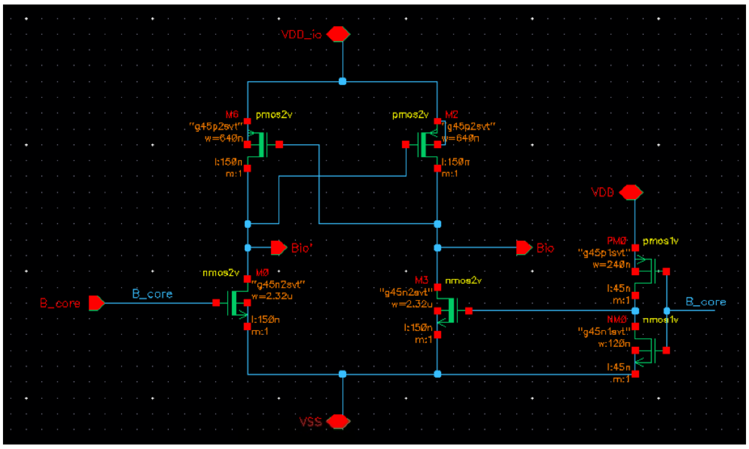


Figure 22. Cadence Schematic of the Level Shifter

Table 9: Transistor Sizes Sorted by Width

Type	Name(s)	Length (nm)	Width
NMOS	NM0	45	120 nm
PMOS	PM0	45	240 nm
PMOS	M2, M6	150	640 nm
NMOS	M0, M3	150	2.32 µm

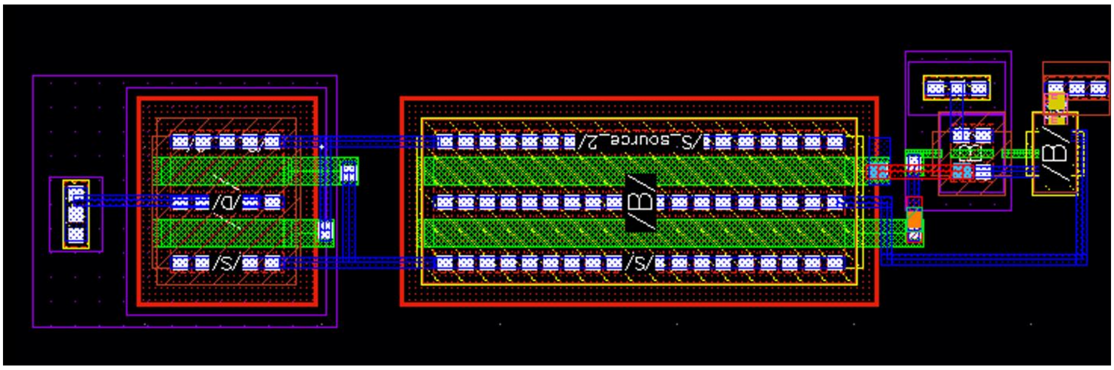


Figure 23. Cadence Layout of the Level Shifter

Supper Buffer (off-chip driver)

According to the project manual, the off-chip driver must be able to function properly such that the output is able to swing from 0.1*VDDIO to 0.9*VDDIO within a single clock period and also that the input and the output of the off-chip driver should not be inverted. Since the off-chip driver will be loaded with load capacitance of 10 pF, which is a considerably large capacitance, a Superbuffer is used as the off-chip driver.

Firstly, since the parameters of the 1.8-V transistors is different from that of 1.1-V transistors, the tpo, Cgmin, and γ parameters were extracted, so a manual calculation of the optimal Superbuffer sizing can be performed. The methods of extraction are similar to the methods taught in the homework. The extracted parameters are: tpo = 5.1ps, Cgmin = 0.7268 fF and γ = 2.43. Then, the optimal number of inverter stages and sizings are found. The delay of the design of several cases that have parameters close to the optimal values are then calculated to compare the delay and area of each case. A table showing the comparison of all the cases is shown in Table 10. Even though design case 1 has the lowest theoretical propagation delay, it would require 66% more area compared to design case 2, even though design case 2 is only about 6% slower than design case 1. Therefore, design case 2 is chosen as our design for the Superbuffer.

Table 10. Comparison of the different designs of the Superbuffer from manual calculation

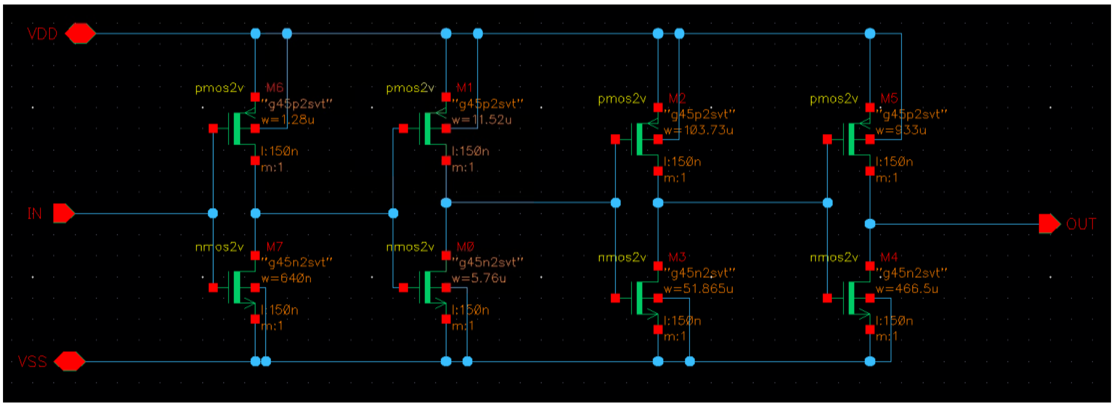
Design Case	Design Parameters			Performance	
	Size of the first inverter	Number of stages	Sizing ratio between each stage (f)	Propagation delay (ps)	Rough estimate of area (in terms of FOX)
1	2	6	4	87	2730
2	2	4	9	92	1640
3	1	6	5	92	3906
4	1	6	4	101	1365

Table 10 shows the transistor sizes, and the approximate size of each inverter used in our design of the Superbuffer circuit. Notice that multi-fingers design is used to minimize the required area of the whole layout, while also conforming with the gpdk045 design rules. The Cadence schematic and layout are shown in Fig.24 and Fig. 25, respectively. Notice that the layout is designed such that the Superbuffer is as compact and neat as possible.

Table 11. Transistor sizes of the Superbuffer circuit

Name (as labeled in schematic)	Length (µm)	Finger Width (µm)	Number of Fingers	Total Width (µm)	Size of that inverter (in terms of FOX, rounded to nearest integer)
M6	0.15	1.28	1	1.28	2
M7	0.15	0.64	1	0.64	

M1	0.15	5.76	2	11.52	18
M0	0.15	2.88	2	5.76	
M2	0.15	9.43	11	103.73	162
M3	0.15	4.715	11	51.865	
M5	0.15	9.33	100	933	1458
M4	0.15	4.665	100	466.5	



VDD
IN
OUT
VSS
pmos2v M6 "g45p2svt" w=1.28u l:150n m:1
pmos2v M1 "g45p2svt" w=11.52u l:150n m:1
pmos2v M2 "g45p2svt" w=103.73u l:150n m:1
pmos2v M5 "g45p2svt" w=933u l:150n m:1
nmos2v M7 "g45n2svt" w=640n l:150n m:1
nmos2v M0 "g45n2svt" w=5.76u l:150n m:1
nmos2v M3 "g45n2svt" w=51.865u l:150n m:1
nmos2v M4 "g45n2svt" w=466.5u l:150n m:1

Figure 24. Cadence Schematic of the Superbuffer circuit

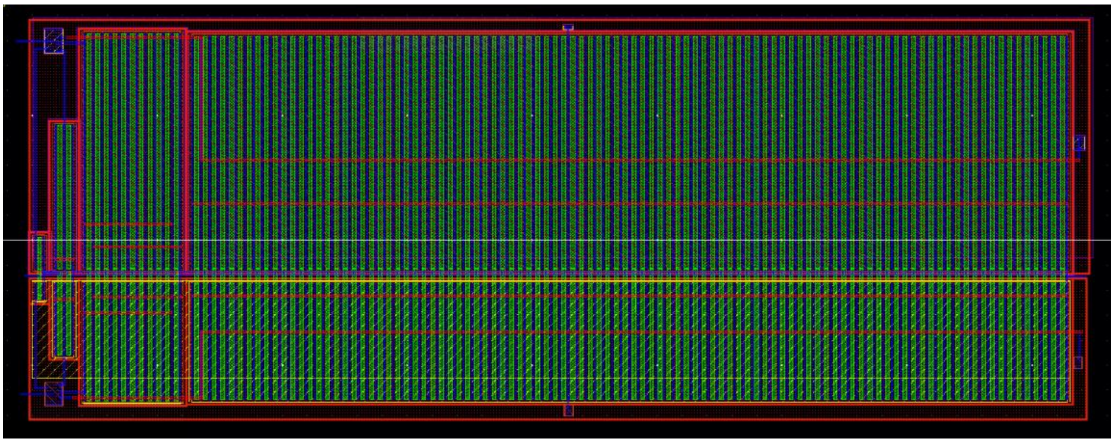


Figure 25. Cadence Layout of the Superbuffer circuit

After the design of both schematic and layout of the Superbuffer circuit on Cadence, the netlist is extracted and the functionality of the Superbuffer when loaded with 10 pF of capacitance is tested using HSpice. The simulated waveform is shown in Fig. 26. Notice that the Superbuffer is functional and the output is not inverted. The time it takes to swing between 0.1*VDDIO to 0.9*VDDIO is approximately 370 ps, which is within one clock period of 416.7 ps. This demonstrates that the Superbuffer can function at the highest frequency of approximately 2.7 GHz.

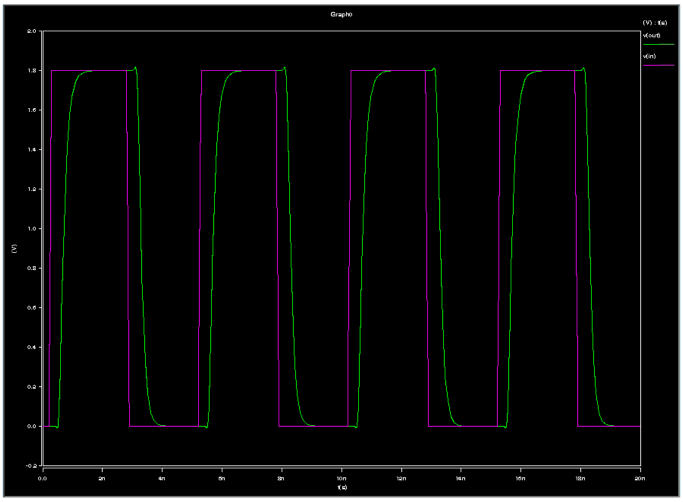


Figure 26. HSpice simulation of 10 pF loaded Superbuffer

iii. Performance table

Note that all metrics were extracted from the Hspice simulation of the extracted top-level netlists. The functionality of the image comparator circuit at the maximum clock frequency (f_{CK}) will be demonstrated in the "Demonstration of circuit functionality" part of this report. Table 12 shows the performance and efficiency metrics.

Table 12. Performance and efficiency metrics.

1	Maximum f_{CK} for correct functionality at all process corners	2.5 GHz
2	Average power drawn from VDD at MIN(f_{CK} listed in row 1, 2.4 GHz)	115.3969 μW
3	Core supply figure-of-merit: Power/frequency (same power and frequency as in row 2)	0.048082 pJ
4	Average power drawn from VDDIO at MIN(f_{CK} listed in row 1, 2.4 GHz)	15.9878 mW
5	IO supply figure-of-merit: Power/frequency (same power and frequency as in row 4)	6.66158 pJ
6	Area of Top-level layout (μm ╳ μm)	71.32 μm ╳ 16.3 μm = 1162.516 μm^2
7	Latency (measured in number of CK cycles)	(measured when there is a change in inputs, and when each signal reaches half of its maximum voltage, i.e. 0.55 V for 1.1V-

		devices and 0.9 V for 1.8V-devices) From: ● Rising edge of CK to Serial_out = 0.76 CK cycles ● Serial_out to R_OUT = 1.63 CK cycles ● Total latency = 2.39 CK cycles

iv. Demonstration of circuit functionality

At 2.4 GHz: Fig. 27-31 shows the functionality of the top-level image comparator circuit at a clock frequency of 2.4 GHz for the TT, SS, SF, FS, and FF process corners, respectively. All the signals (except from D2:D0, since they are internal signals) given in the timing diagram for the system given in the Project Manual were plotted as a waveform for all the simulations, so the functionality of the top-level circuit can easily be checked, with each bit of input Y and X plotted in the same box, so each input bits can easily be compared, when checking the functionality of the image comparator circuit. The results will be explained and analyzed in the "Discussion of results" section of this report. Since, from our simulations, the top-level image comparator circuit still has the correct functionality up to a clock frequency of 2.5 GHz, the functionality of the top-level image comparator circuit for the SS process corner at a clock frequency of 2.5 GHz is shown in Fig.30. Even though the top-level image comparator circuit is functional for all the process corners at a clock frequency of 2.5 GHz, only the functionality for the SS process corner case is shown, to avoid cramming all the screenshots of the waveforms in this report. The SS process corner is chosen, since it seems to give the worst-case results, out of all the process corners, even though the results of each process corner is not that different, and the top-level circuit is fully functional for all process corners. Even though for some cases, R_OUT wasn't able to reach and stabilize at 1.8 V, it still passes 0.9 V by quite a margin, which means that the top-level circuit is functional, even when loaded with a 10 pF capacitive load.

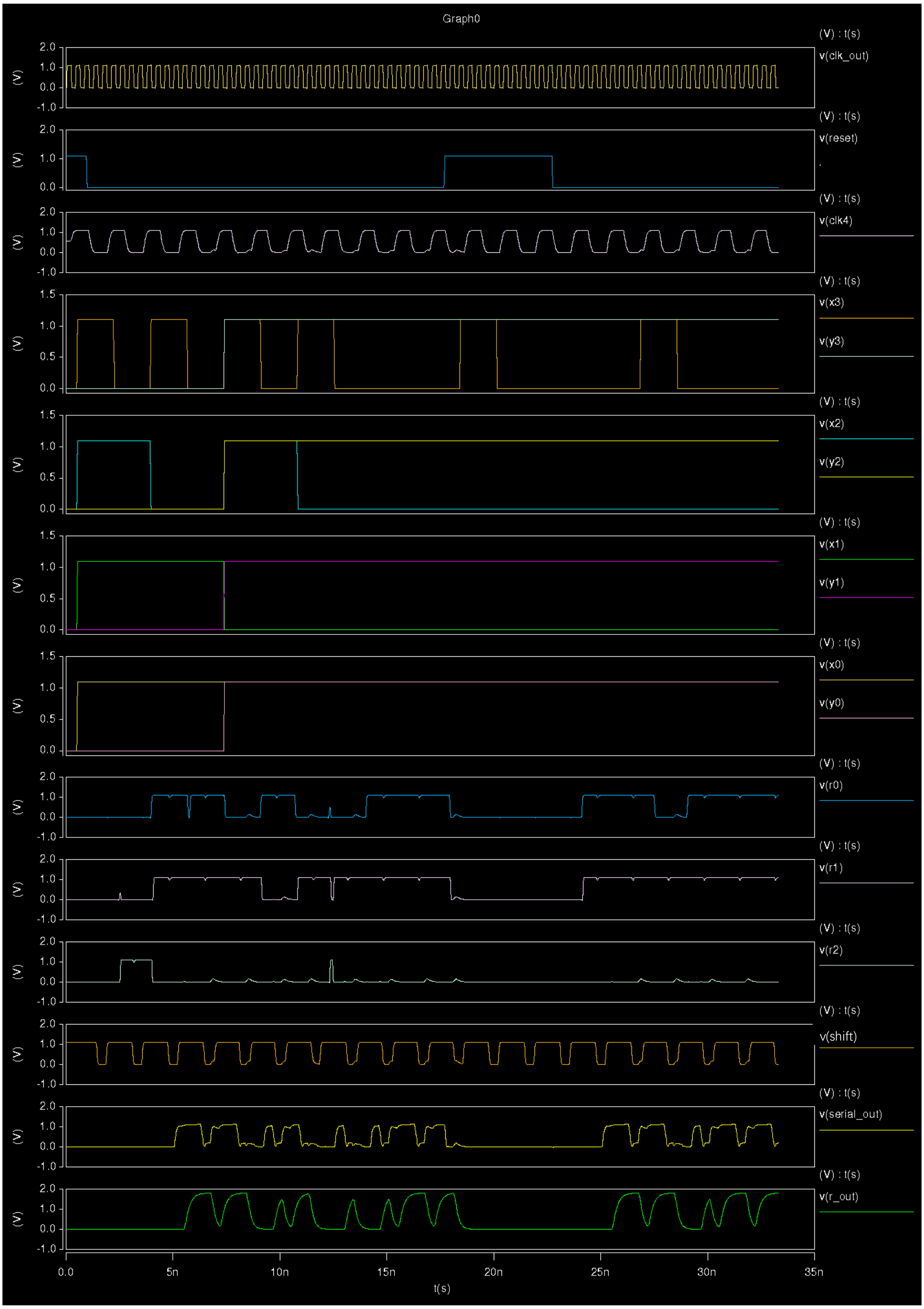


Figure 27. Demonstration of circuit functionality for the TT process corner at 2.4 GHz

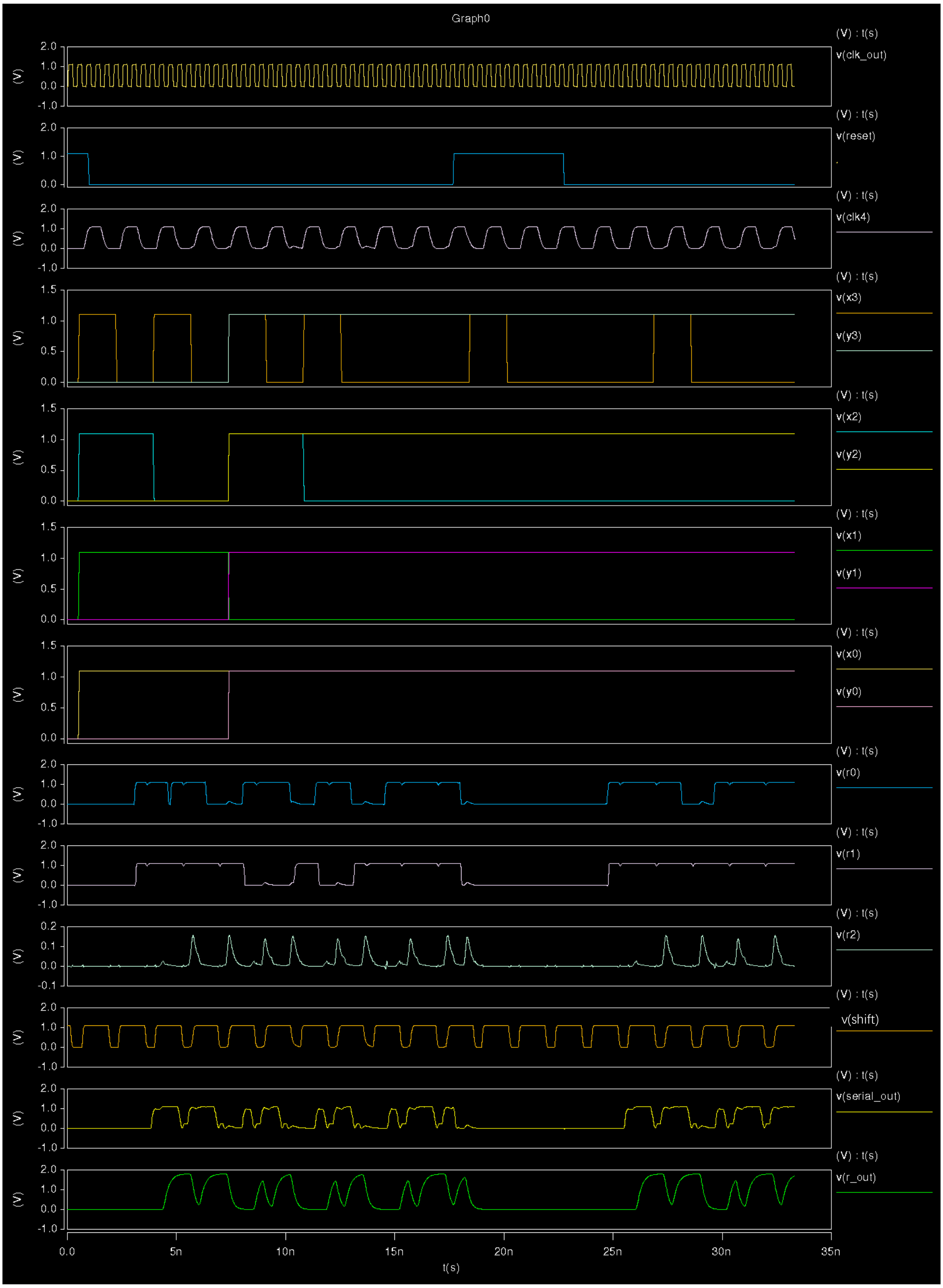


Figure 28. Demonstration of circuit functionality for the SS process corner at 2.4 GHz

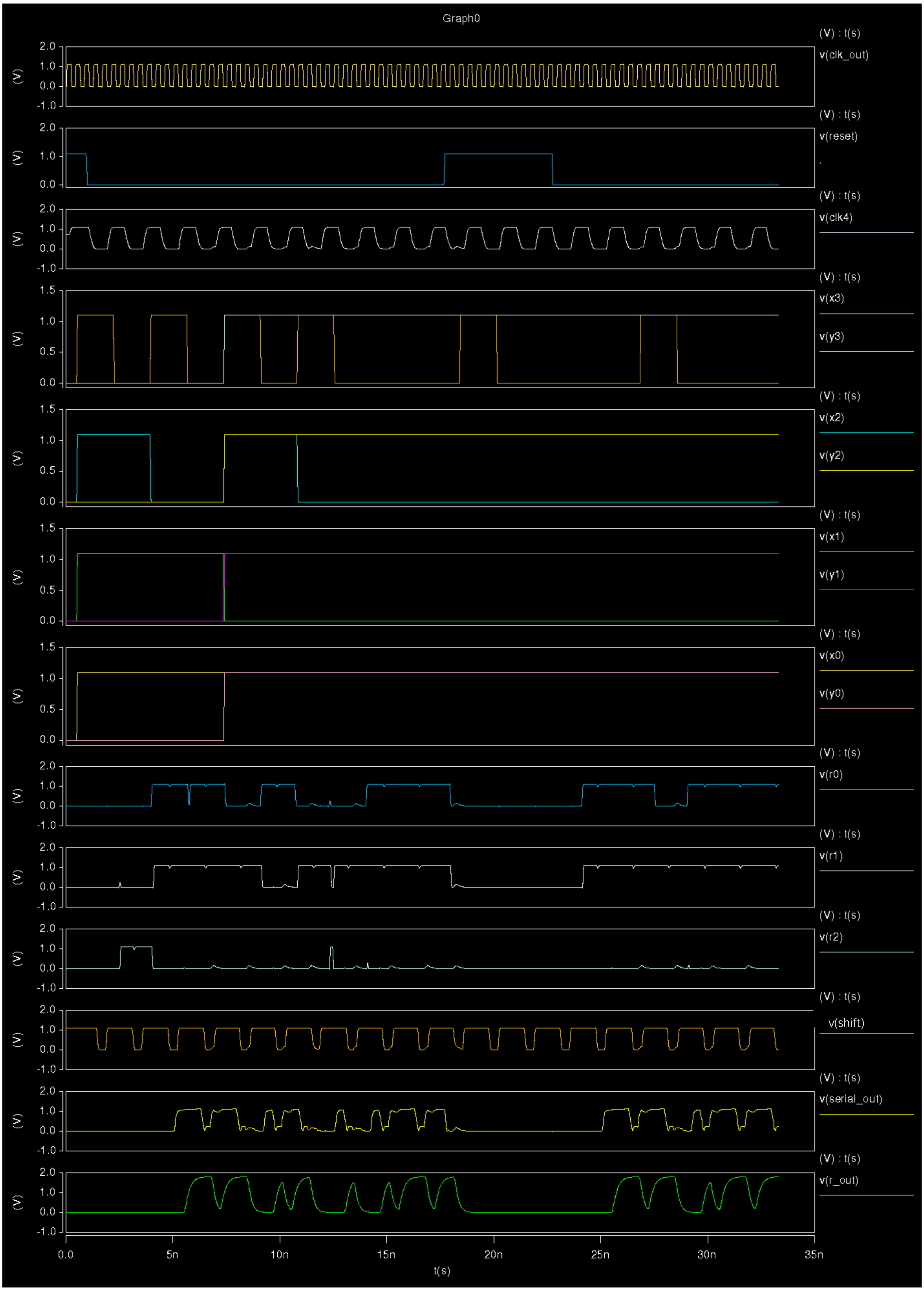


Figure 29. Demonstration of circuit functionality for the SF process corner at 2.4 GHz

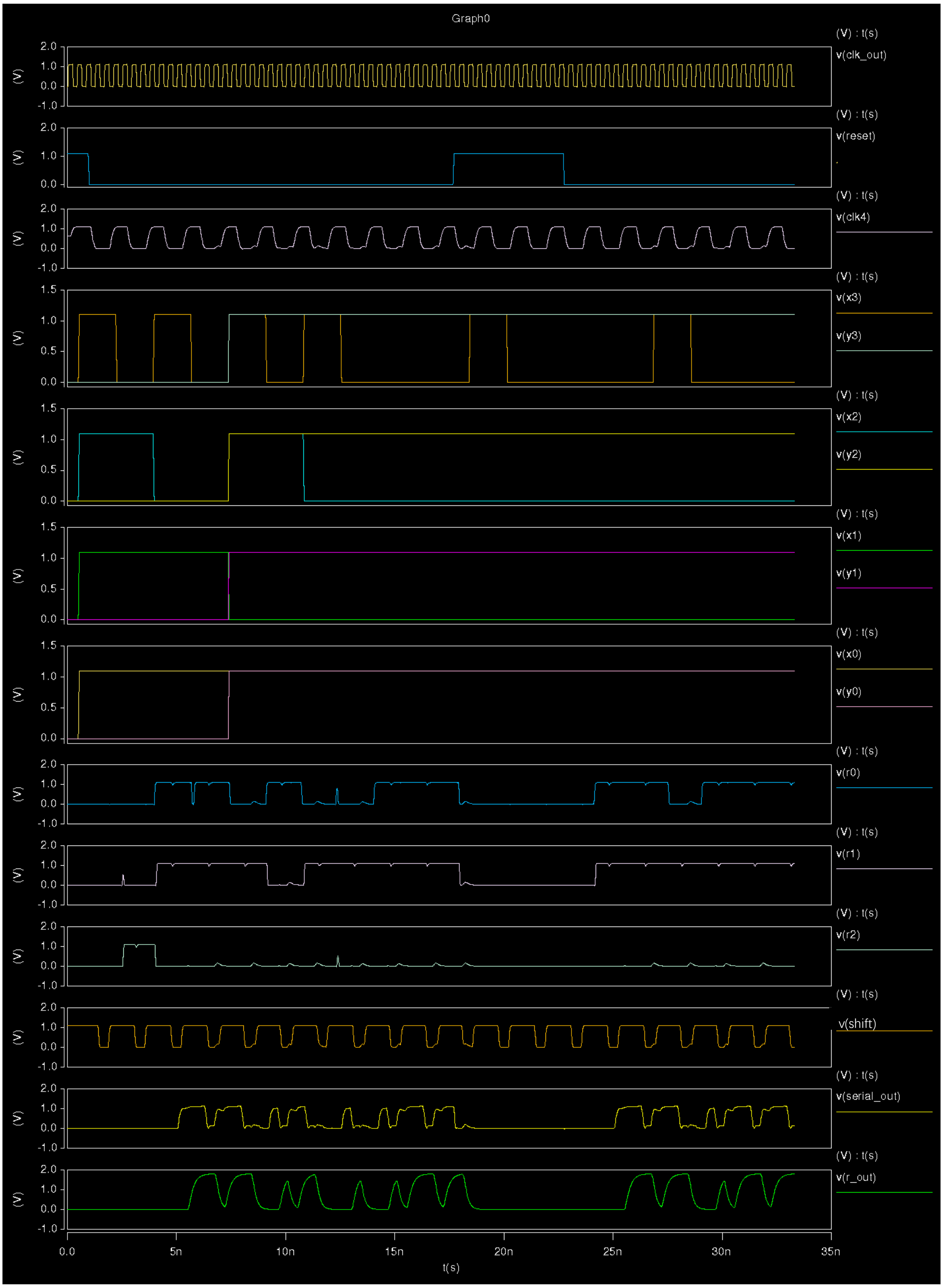


Figure 30. Demonstration of circuit functionality for the FS process corner at 2.4 GHz

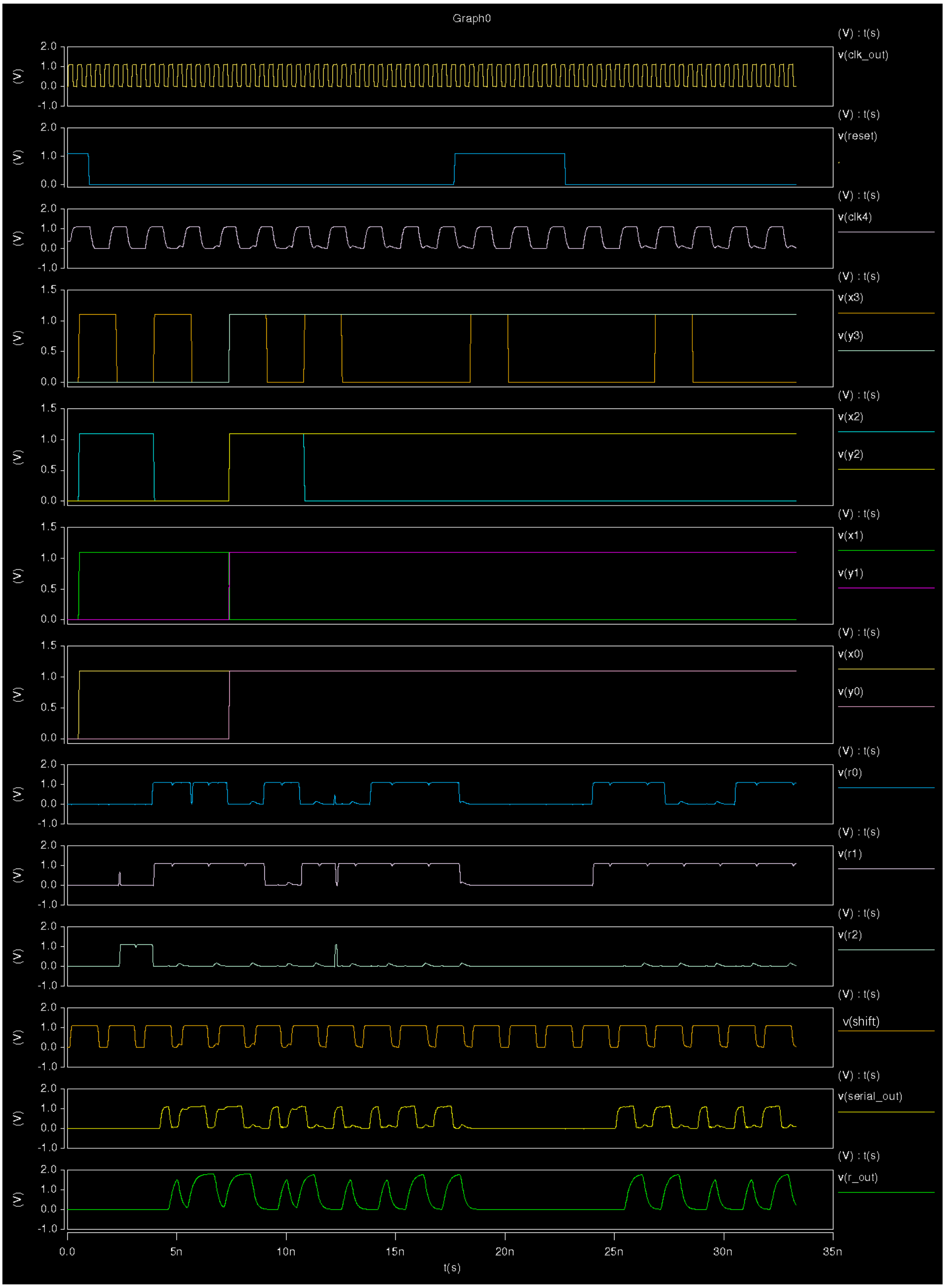


Figure 31. Demonstration of circuit functionality for the FF process corner at 2.4 GHz

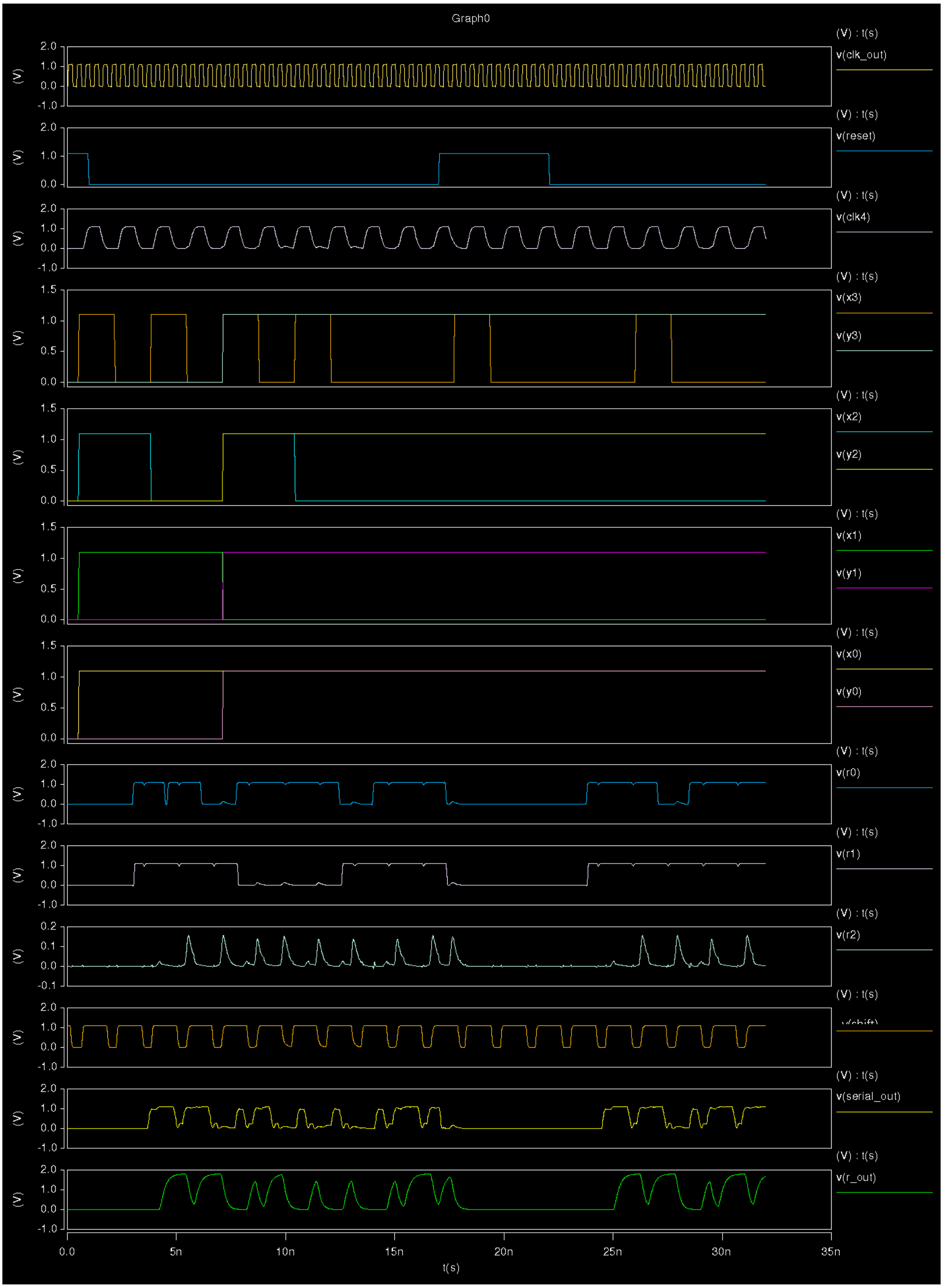


Figure 32. Demonstration of circuit functionality for the SS process corner at 2.5 GHz

v. Discussion of results

As demonstrated in section iv ("Demonstration of circuit functionality"), our top-level image comparator circuit is functional not only at 2.4 GHz, but also up to a frequency of 2.5 GHz. To explain the result, firstly, it should be appreciated that the frequency divider is functional, judging from the clock divided by four signals (clk4) in the waveforms in Fig. 27-31. As expected, every time the "reset" signal is high, r0 - r2, Serial_out and R_OUT are logic 0, demonstrating that all registers in the design are resettable, and that the circuit is in a standby state whenever the reset signal is set to high. Secondly, the r0 - r2 signals are correct, representing the 3-bit binary count of the number of bits of X and Y that are different. This shows that the combinational part of our design (XOR gates, half adders, and full adder) are functioning correctly, even at a frequency of 2.5 GHz as shown in Fig. 32. Furthermore, it should be appreciated that the generation of the internal "shift" signal functions properly as a 75% duty cycle signal, operating at the frequency of the clock, representing the PISO loading new data when "shift" = 0, and the PISO shifting the data through the shift register when "shift" = 1. Next, the "Serial_out" signal also shows the correct serial output data, according to r0 - r2, shift, and CK signals. This demonstrates that our PISO block is functional, even at a frequency of 2.5 GHz, according to the block diagram of the PISO shift register given in the Project Manual. Lastly, the level shifter and off-chip driver demonstrate their functionality, since the R_OUT signal is a scaled version of the Serial_out signal to 1.8 V when the logic is high, with some delay from Serial_out to R_OUT, as expected since R_OUT is driving a large capacitive load of 10 pF.

To determine the maximum frequency at which the circuit can operate with the correct functionality, the clock period (TCK) was gradually decreased in steps of 0.01 ns, starting from an initial value of 0.42 ns, using the HSpice simulation. Our top-level circuit functions correctly up to a clock period of 0.4 ns, which corresponds to a frequency of 2.5 GHz. As previously mentioned in the description of the design Superbuffer block of our circuit, the "theoretical" highest frequency in which the Superbuffer can function is approximately 2.7 GHz, which is close to the maximum frequency of 2.5 GHz we obtained. In the "Performance and Efficiency Metrics" table (Table 12), it is demonstrated that the part of the circuit that is responsible for most of the delay is between Serial_out and the output R_OUT. Since the Superbuffer needs to drive a very large capacitive load of 10 pF, the Superbuffer is most likely responsible for most of the latency of the top-level circuit. Therefore, the maximum frequency of 2.5 GHz of our top-level circuit most likely has the Superbuffer as the bottleneck of the frequency, limiting the circuit from being able to function at an even higher frequency. This can also be shown in Fig. 30 where there are some cases in which R_OUT was not able to stabilize at 1.8 V when the logic is high. However, this should not affect the functionality of the logic since R_OUT was able to pass 0.9 V by a large margin every time it swings high. It is important to note that the large delay of the Superbuffer is expected, since the Superbuffer is driving a very large capacitive load, so it would require a longer time to swing from logic low to logic high, compared to other blocks in the circuit.

vi. Top view of the layout

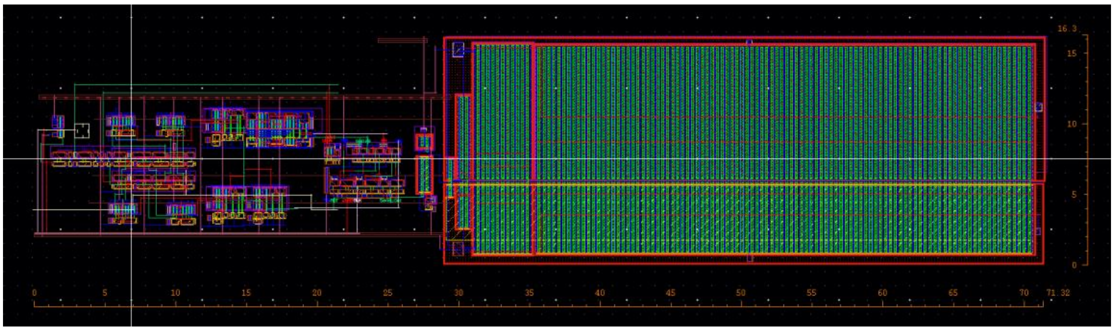


Figure 33. Cadence layout of the image comparator circuit

To implement the Image Comparator circuit, all of our block designs were combined. Fig. 2 (in the first section) shows the Cadence schematic and Fig. 33 shows the Cadence layout of the image comparator circuit. The main goal of the layout design is functionality, while also ensuring a compact and neat layout. As expected, the area of the Superbuffer layout is the majority of the top-level layout design, due to its large sizings to drive the large capacitive load. The total area of the top-level layout is approximately 71.32 µm X 16.3 µm. All the instructions given in the Project Manual were followed, including the use of Metal 9 as power (VDD and VDDIO) and ground (VSS) horizontal buses and the use of Metal 8 as vertical buses, the connection of the CLK signal with a 1 µm X 1 µm square of Metal 5 at a single location (at the left edge). The top-level circuit passed both DRC and LVS, the netlists were successfully extracted, and RC simulations were performed, which will be discussed in the "Demonstration of circuit functionality" section of this report.

To implement a compact and neat layout, Metal 8 and Metal 9 are only used for power and ground connections as vertical and horizontal buses, respectively. Other Metal layers were also used, since the different Metal layers can be overlapped in the layout without actually making electrical connection, making it possible to create a compact layout. We also added labels internally for certain important signals like r0-r2, making it easier to debug our connections to pass LVS, while also making it possible to check for delays at a certain location of the top-level circuit. Since all of our individual blocks were already optimized in terms of design and compactness, it was not hard for us to optimize the design and compactness of our top-level layout.

vii. Workload distribution

Ying-Ching Lee: XOR gate design, layout and test. Hspice netlist testing and construction. Super buffer and part of the whole circuit design. Organizing team meeting and group progress.

Jay Nathan: PISO, level shifter, select signal design layout and testing. MUX layout.

Eric Tang: D register and frequency divider design, layout and test. Debug the lvs problem for group members.

Phuvit Thirasuntrakul: Full adder, super buffer, clk inverter design, layout and test. Hspice netlist testing and construction.

Jie Yan: Half adder design, layout and test. Power connection. Sort up the report.

References

[1] K. Navi, O. Kavehei, M. Rouholamini, A. Sahafi, S. Mehrabi, and N. Dadkhahi, "Low-power and high-performance 1-bit CMOS full-adder cell," *Journal of Computers*, vol. 3, no. 2, Feb. 2008. doi:10.4304/jcp.3.2.48-54