Project #2 Report Cover Sheet

Submit on Blackboard by Friday 6:00 PM, 4/20/2024

PROJECT TITLE: 4 Bit Binary Synchronous Up Counter

Student's Name: **Mahir Asef** .

	Topics ***DO NOT alter the order and shape of this table***	GRADES
Required	Section1: Executive Summary (1/2 page to 1 page)	
Required	Section 2: Introduction and Background	/5
Required	Section 3: Electric Circuit Schematics (only transistors)	/10
Required	Section 4: Detailed Electric Layouts (*The entire layout should be on single page landscape form.*) Provide detailed layouts of subcircuits with simulations.	/30
Required	Section 5: LTSPICE code and parasitic extractions with calculation analysis. Put only samples of code. Do Not provide multiple pages of LTSPICE code. (-2 points for more than 2 pages of code)	/15
Required	Section 6: IRSIM Logic Simulations and Measurements for Layout and Schematic (must provide comparisons between the two using tables.)	/15
Required	Section 7: Measurements in LTSPICE for delays for Layout and Schematic (must provide comparisons between the two using tables)	/15
Required	Section 8: Measurements of power, delay, chip area, timing, number of transistors for the layout. (If you are using TG, static or dynamic CMOS, compare here using tables.)	/5
Required	Section 9: Pathwave ADS simulations for RC circuits	/5
Required	Section 10:Conclusion and References	
	Late Report (-5pts per day for 2 days max.)	
Penalty	Does the project work? (-20pts for not functioning design) Yes or No	YES
	TOTAL	**/100**

Five points will be deducted for not following the directions. Do Not modify the table.

Table of Contents

Topic: **Page:**

Section 1: Executive Summary:

The objective of this experiment is to create a 4-bit synchronous up counter with D flip flops. The 4-bit synchronous up counter designed using D flip-flops is a digital circuit that counts in binary from 0000 to 1111, representing decimal numbers from 0 to 15. This counter operates synchronously which means its outputs change only on clock edges, ensuring proper timing and reliable functionality.

This circuit utilizes four D flip-flops connected in a cascading manner to create a 4-bit counter. The D flip-flops' inputs are connected to the previous flip-flop's output which builds a synchronous chain. Only one clock is used to drive all the flip-flops to ensure and implement synchronicity. This counter works by incrementing by one at each rising edge of the clock following the binary counting sequence. When the count reaches its maximum value (1111), it resets to 0000 creating a cycle. The counter provides parallel output of the current count in binary form by showing a logic high or low bit for each output bit from Q3 to Q0 where Q3 is the most significant bit and q0 is the least significant bit.

The synchronous up counter is great for reliable and predictable counting behavior due to synchronous design. It also caused minimal propagation delay between clock input and output changes ensuring fast and timely execution. It's also scalable to higher bits.

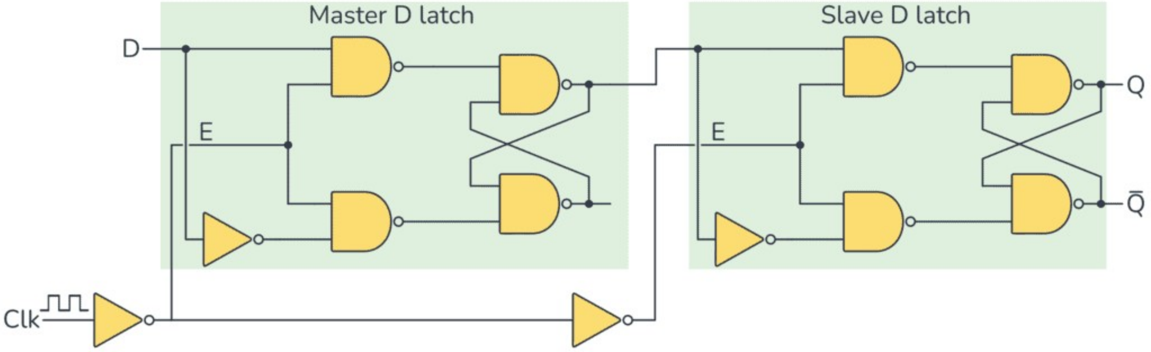
The up counter is commonly used in digital systems for tasks such as event counting, address generation, and sequence control. It's also very suitable for educational purposes to demonstrate synchronous counter design principles since it implements a very basic counter that requires minimal gates and a very commonly used D flip flop. It can be integrated into larger digital systems for various counting and control applications.

Overall, the 4-bit synchronous up counter implemented through D flip-flops combines simplicity, reliability, and versatility, making it a fundamental component in digital circuit design and education.

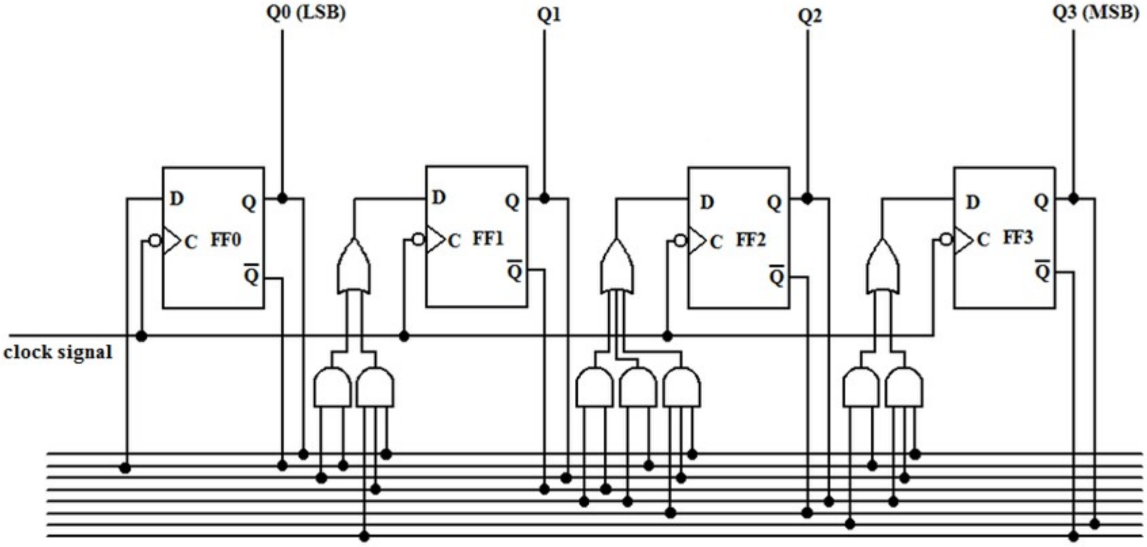
Section 2: Introduction and Background

In the realm of digital electronics, counters play a pivotal role in various applications, from simple event counting to complex sequence generation. Among the different types of counters, the synchronous up counter using D flip-flops stands out as a fundamental and widely used design. This introduction sets the stage for understanding the intricacies and significance of a 4-bit synchronous up counter, delving into its design principles and operational functionality.

The following 4-bit synchronous up counter uses 4 D flip flop to count from 0000 (0) to 1111 (15) and then resets to 0000 after. is a digital circuit capable of sequentially counting in binary from 0000 to 1111. D flip flops are the fundamental blocks of building this counter. D flip flop stores and transfer the input data with a delay by using clock signals. In our case, we will use rising edge to store the outputs from present state to be shown at a delay in the next stage. We will use the master slave design implementation to create our D flip flop circuit. A master slave D flip flop basically consists of a leading D latch that's built from a SR latch as well as a follower D latch. Since we are changing the output at the rising edge of the clock signal, we will be using two inverters in between the D latches to make the D flip flop function properly. Each D flip flop has a D input with a clock input that produces a Q output which is 1 bit. For the 4-bit counter we will need 4 of such flip flops.



To build the whole counter using D flip flops, we will need to connect the output of each D flip flop as inputs to the next D flip flop, so it shows the right pulse at next stage. To cascade and connect our flip flops, we will use a 3 input AND gate connected with a XOR gate, a 3 input AND gate with XOR gate and another XOR gate by itself. The design of the full 4-bit synchronous D flip flop up-counter will implement a simplified structure of this:



Once we successfully design the 4-bit up counter, the output of the circuit should follow the following truth logic table where we can see as Q3-Q0 written in 4 bit binary where the next state of Q3-Q0 is 1 increment if present state in binary.

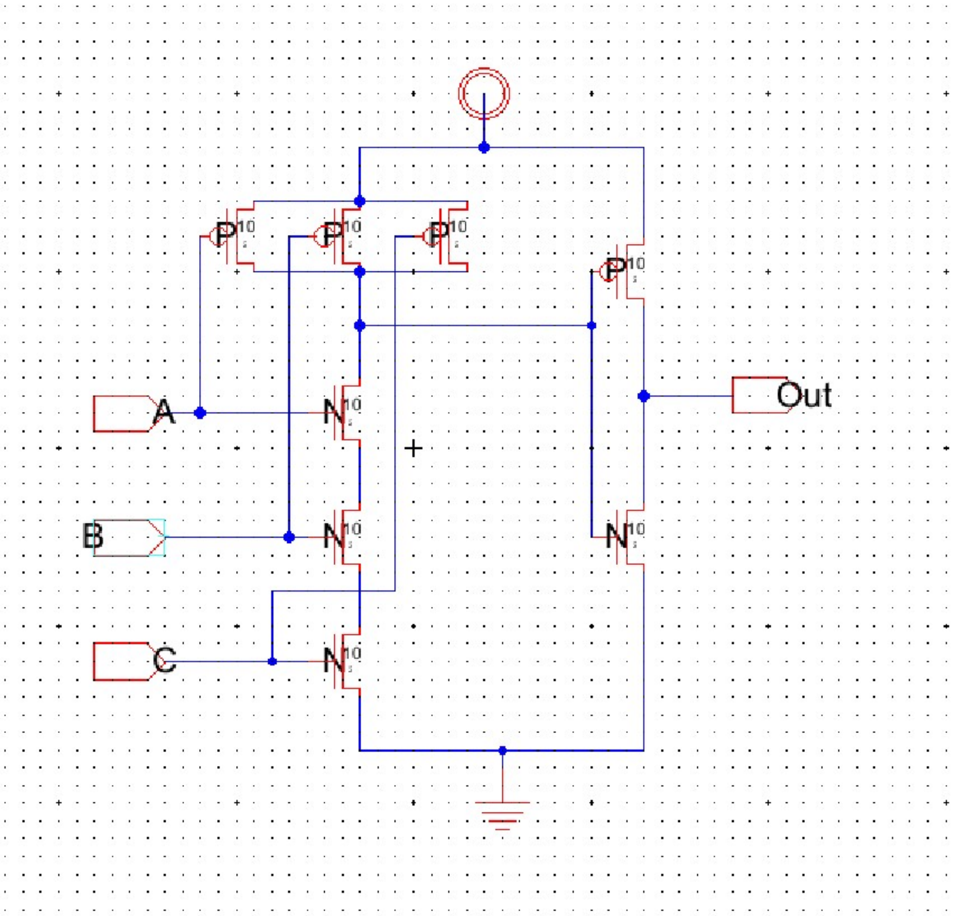
Truth Table:

Present State				Next State				DFF Inputs			
Q3	**Q2**	**Q1**	**Q0**	**Q3**	**Q2**	**Q1**	**Q0**	**D3**	**D2**	**D1**	**D0**
0	0	0	0	0	0	0	1	0	0	0	1
0	0	0	1	0	0	1	0	0	0	1	0
0	0	1	0	0	0	1	1	0	0	1	1

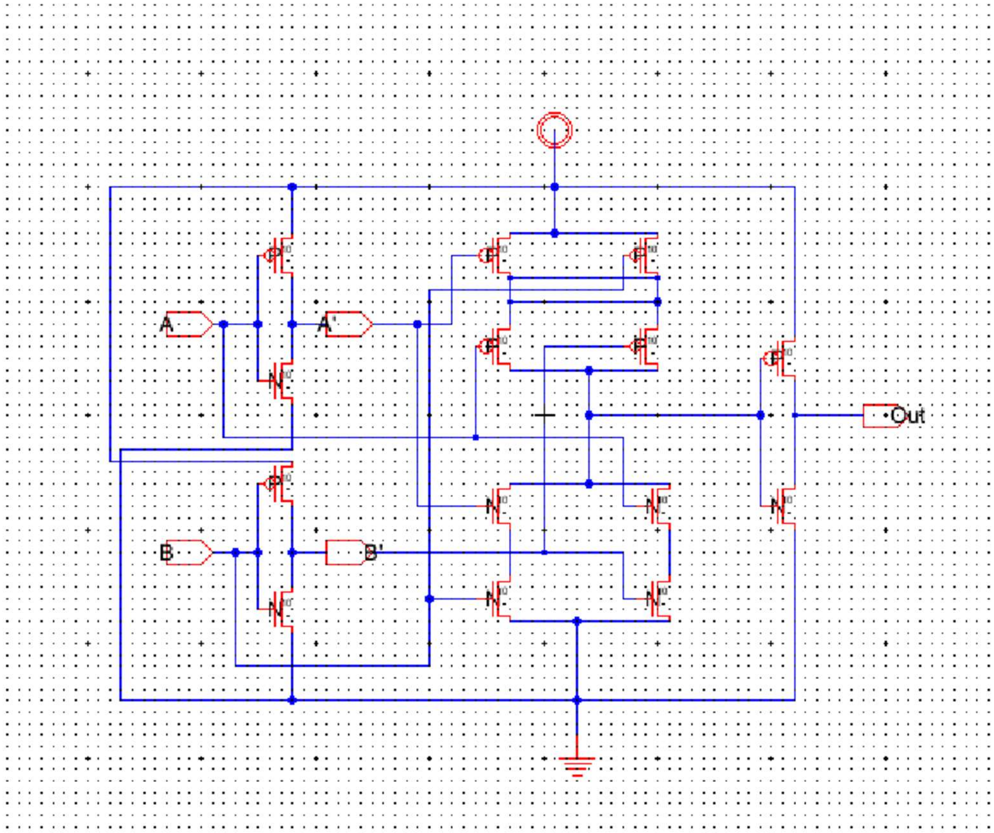
0	0	1	1	0	1	0	0	0	1	0	0
0	1	0	0	0	1	0	1	0	1	0	1
0	1	0	1	0	1	1	0	0	1	1	0
0	1	1	0	0	1	1	1	0	1	1	1
0	1	1	1	1	0	0	0	1	0	0	0
1	0	0	0	1	0	0	1	1	0	0	1
1	0	0	1	1	0	1	0	1	0	1	0
1	0	1	0	1	0	1	1	1	0	1	1
1	0	1	1	1	1	0	0	1	1	0	0
1	1	0	0	1	1	0	1	1	1	0	1
1	1	0	1	1	1	1	0	1	1	1	0
1	1	1	0	1	1	1	1	1	1	1	1
1	1	1	1	0	0	0	0	0	0	0	0

Section 3: Electric Circuit Schematics:

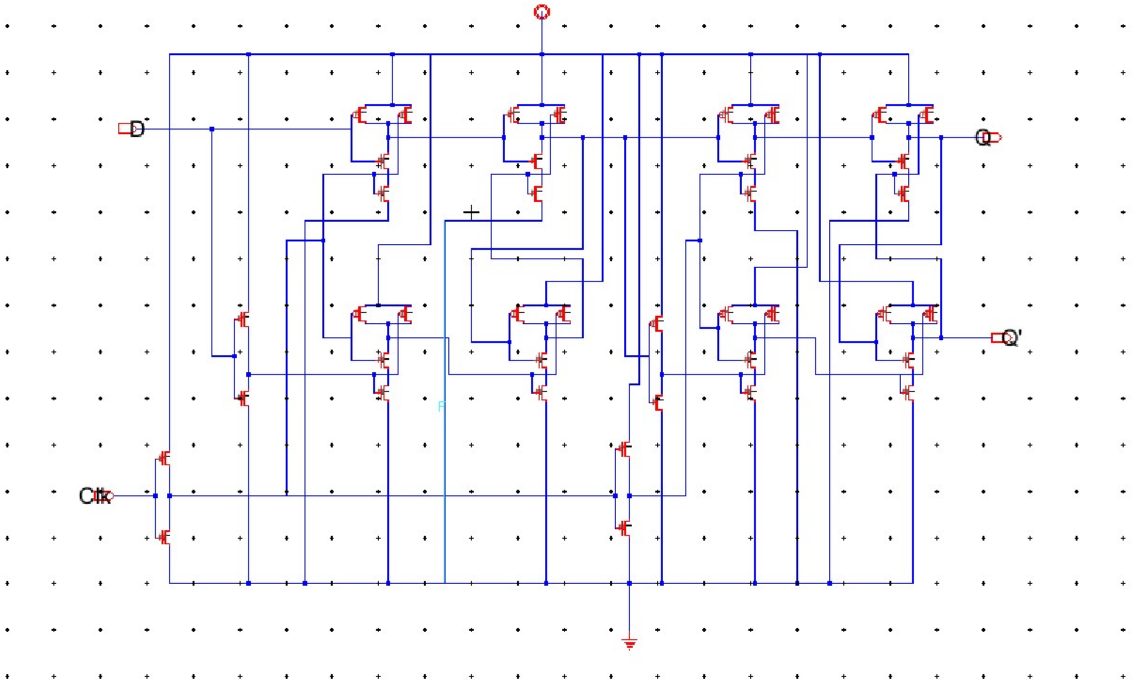
3_input AND gate:



XOR Gate:

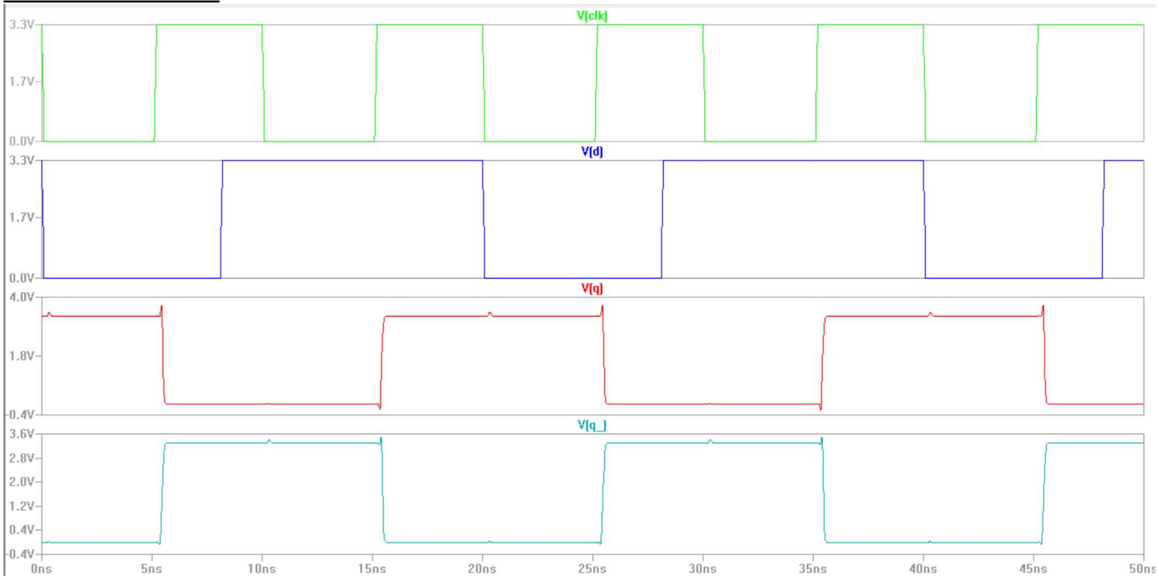


D Flip Flop:

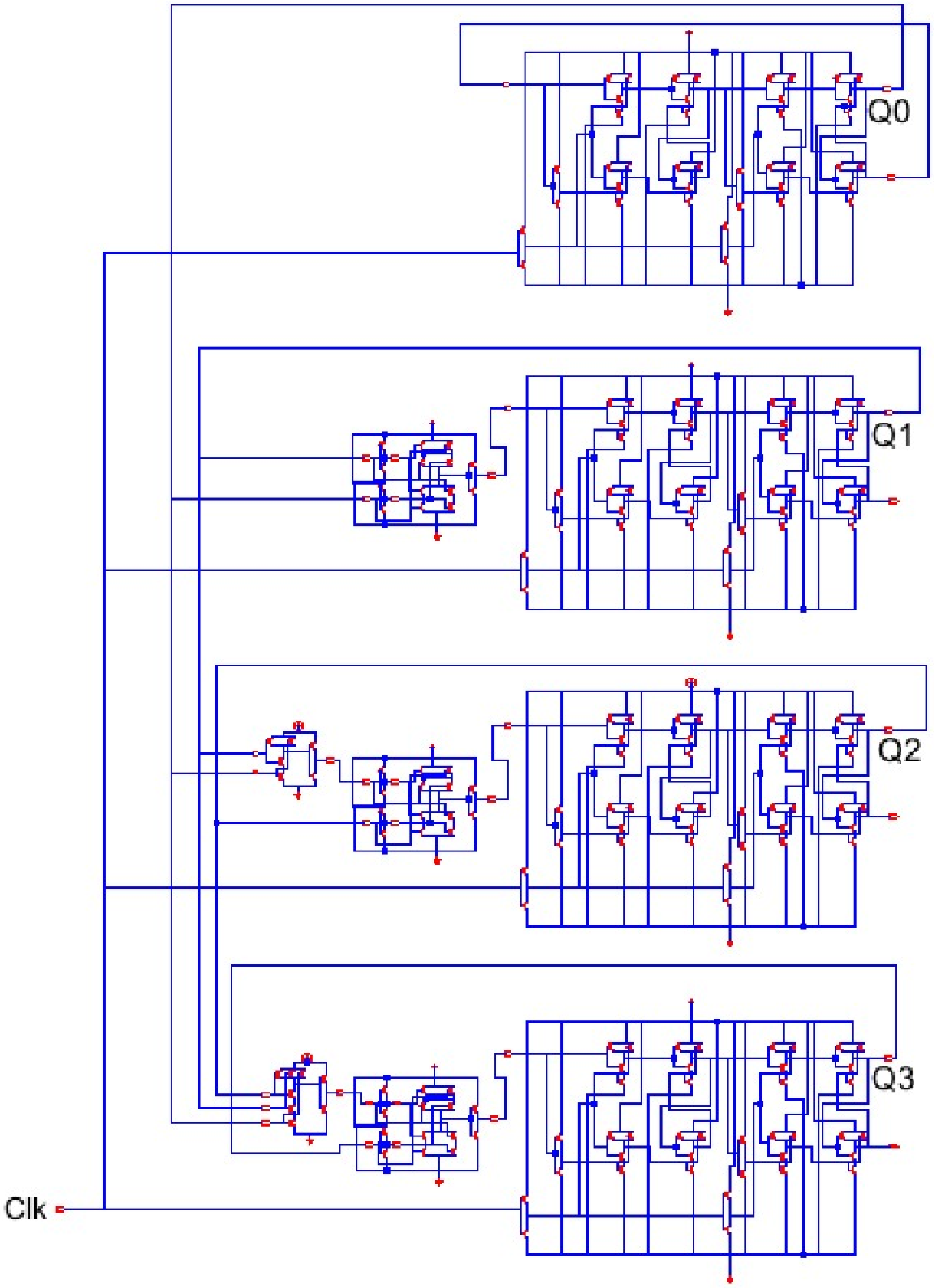


Here the D flip fop is made with digital transistors that consist of two D latches. Each D latches are made with 4 nand gates and an inverter. The latches are then cascased together with the D input and clock signal with a two inverters which allows the flip flop to flop to function on rising edge of the clock. The flip flop updates the value of Q based on the D input at each rsing edge as shown below.

DFF Simulation:



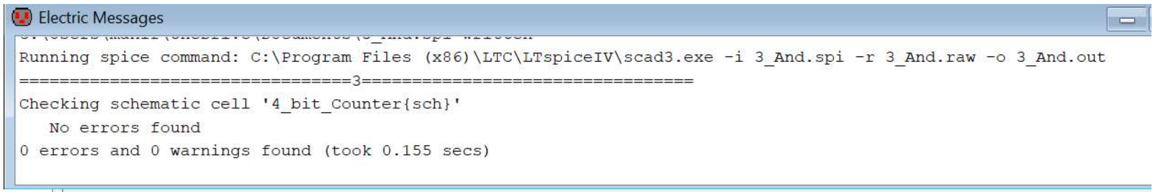
4-Bit Synchronous Up Counter:



Note: The Grid is ON. Circuit is too zoomed out for full view.

This is the full transistor design on electric schematic for the 4-bit synchronous up counter using D flip flops. Here, D3 takes an input of Q0.Q1.Q2 through an XOR gate as well as cascade back the Q3 output back into the same XOR gate. The output of XOR gate is the D input. Q3 output is the most significant bit of the 4-bit binary digit. D2 takes the input Q0.Q1 through an XOR gate paired with Q2 which is a cascaded output feedback back into input. The output of XOR gate is the D input. D1 takes Q0 and Q1 as the input of another XOR gate and then the output of XOR connects to D input. Finally, the D0 only cascades back Q0' as the D input while producing Q0 output which is the east significant bit of the 4-bit number. Each flip flop is also connected through ne common clock at the enable input for synchronization.

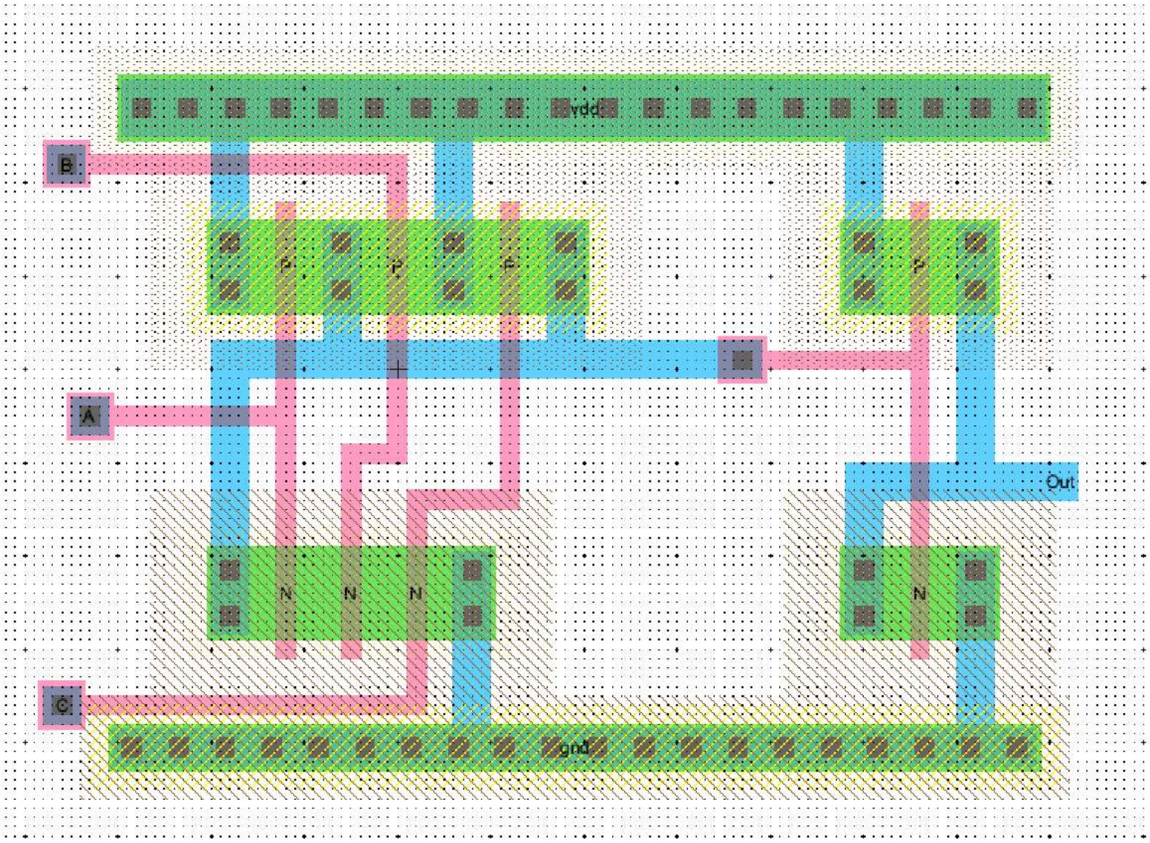
DRC Check:



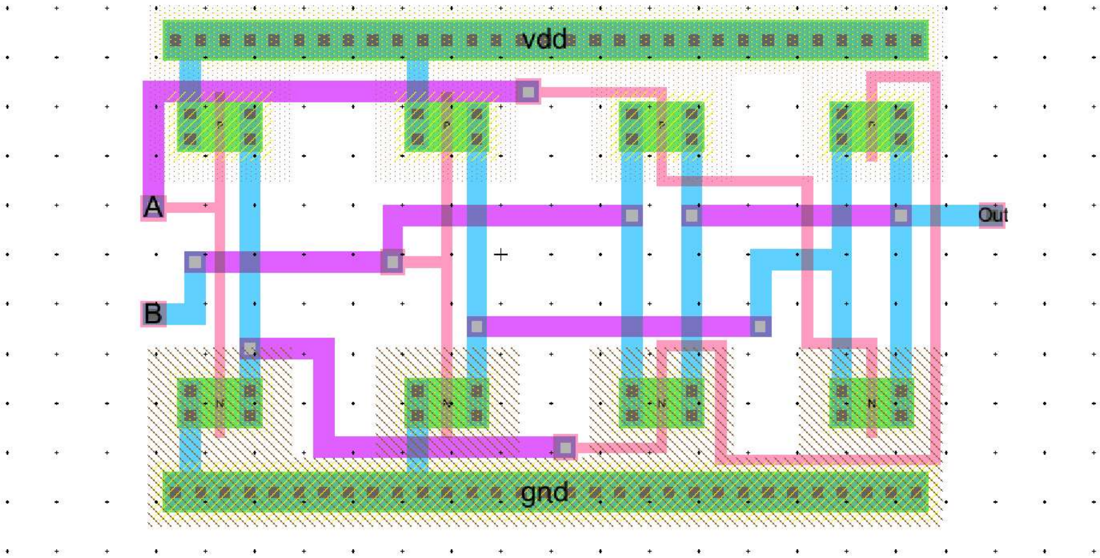
```
Running spice command: C:\Program Files (x86)\LTC\LTspiceIV\scad3.exe -i 3_And.spi -r 3_And.raw -o 3_And.out
=================================3=================================
Checking schematic cell '4_bit_Counter{sch}'
   No errors found
0 errors and 0 warnings found (took 0.155 secs)
```

Section 4: Detailed Electric Layouts:

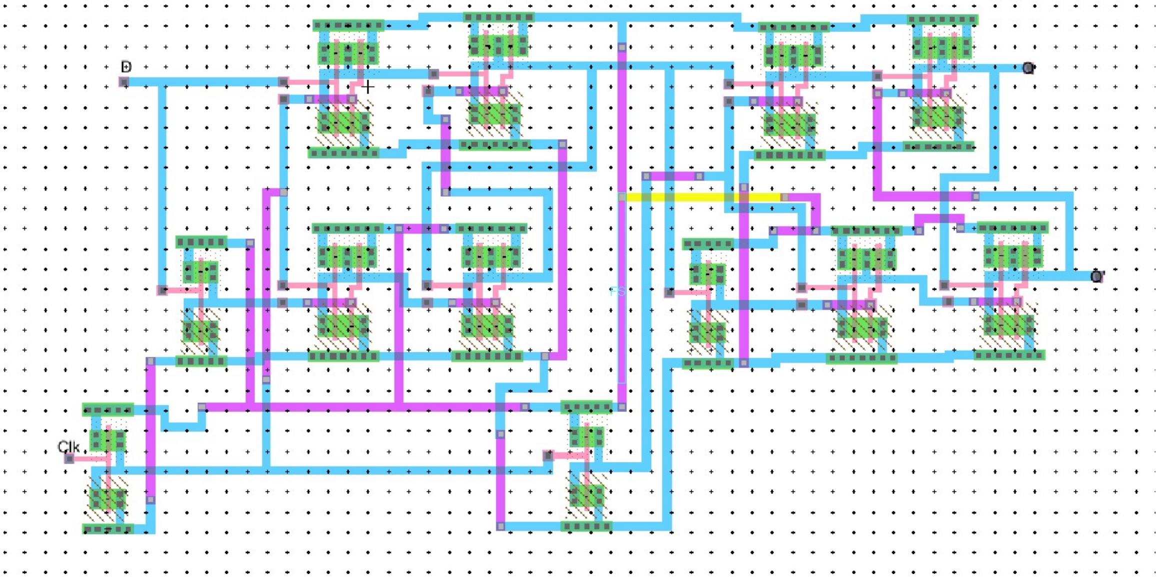
3_input AND:



XOR:

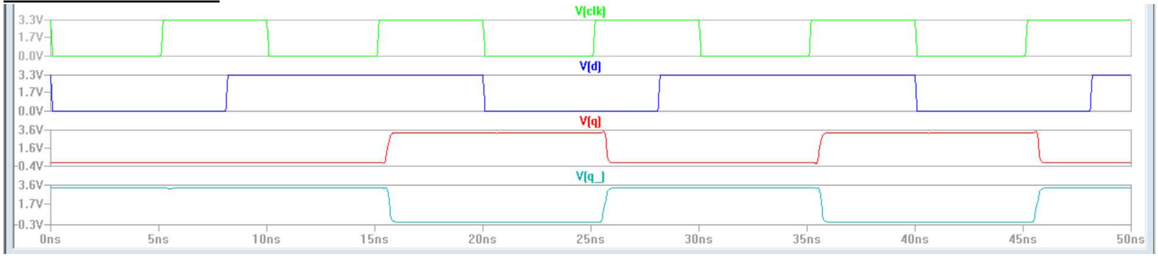


D Flip Flop:

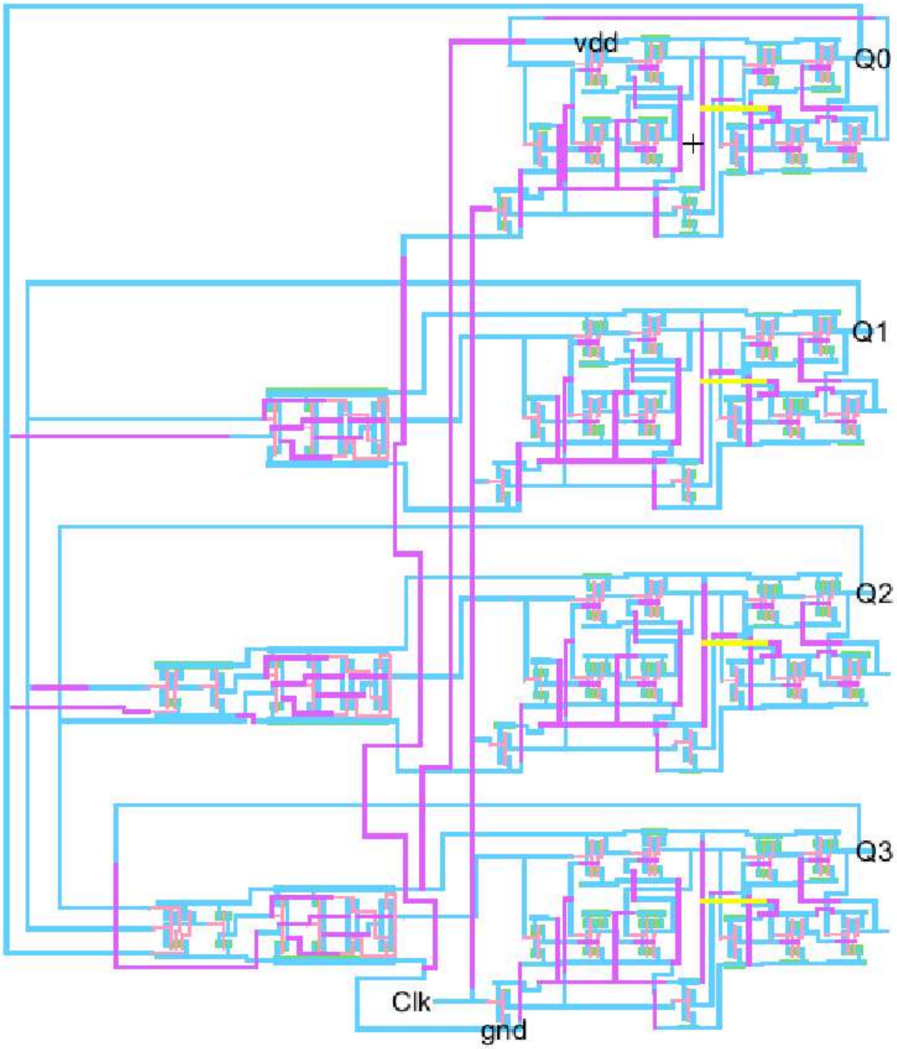


Here the DFF is made in layout with PMOS and NMOS which are interconnected with Metal conatctas and wires. The fip flop is made of D latches which are made of 4 NAND gates and an inverter. The latches are connected together and fed with D input as well as clock input by two inveters which allows the flip flop to read the Q bits at rising edge of the clock which is shown below in the simulation.

DFF Simulation:



4-Bit Synchronous Up Counter:

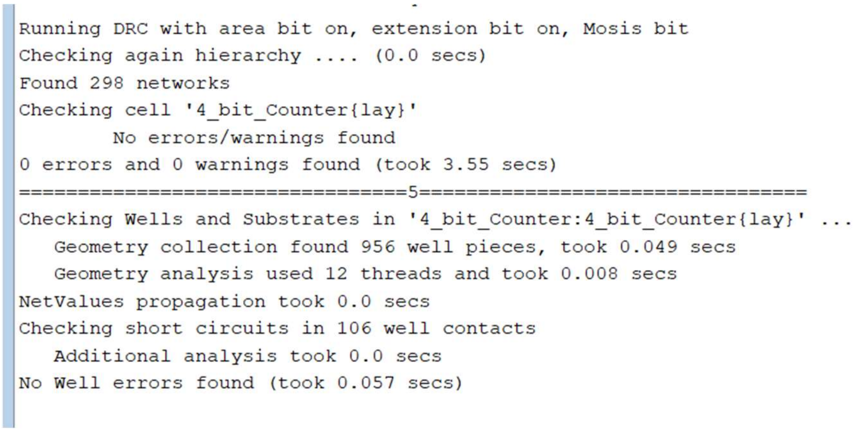


Note: The Grid is ON. Circuit is too zoomed out for full view.

This is the entire layout for the 4-bit synchronous up counter using D flip flops. Here, D3 (the bottom DFF) takes an input of Q0.Q1.Q2 through a XOR gate as well as cascade back the Q3 output back into the same XOR gate. The output of XOR gate is the D input.

D2 takes the input Q0.Q1 through an XOR gate paired with Q2 which is a cascaded output feedback back into input. The output of XOR gate is the D input. D1 takes Q0 and Q1 as the input of another XOR gate and then the output of XOR connects to D input. Finally, the D0 only cascades back Q0’ as the D input. Each flip flop is also connected through a new common clock at the enable input for synchronization. Q0 is the least significant bit whereas Q3 is the most significant bit.

DRC and ERC Check:

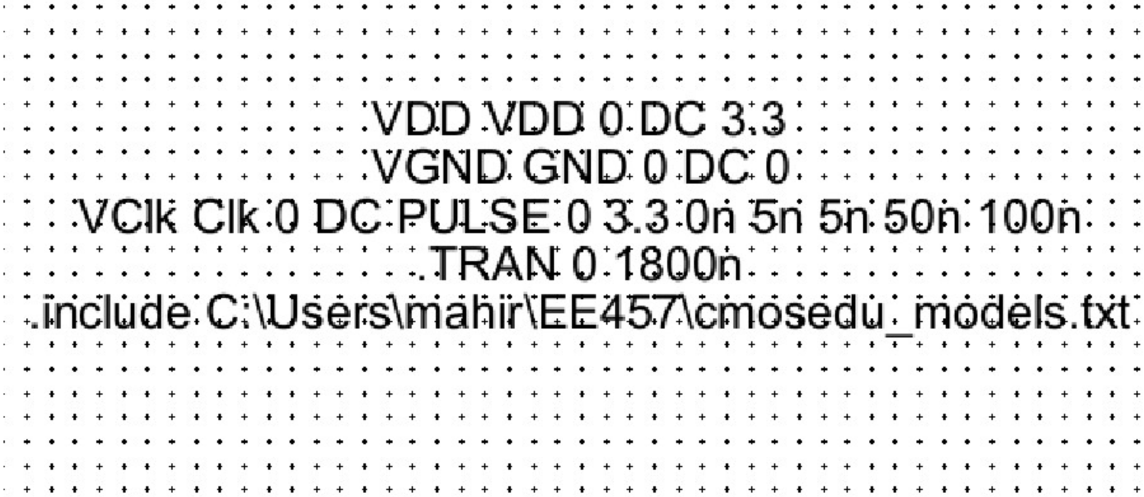


```
Running DRC with area bit on, extension bit on, Mosis bit
Checking again hierarchy .... (0.0 secs)
Found 298 networks
Checking cell '4_bit_Counter{lay}'
        No errors/warnings found
0 errors and 0 warnings found (took 3.55 secs)
=================================5=================================
Checking Wells and Substrates in '4_bit_Counter:4_bit_Counter{lay}' ...
   Geometry collection found 956 well pieces, took 0.049 secs
   Geometry analysis used 12 threads and took 0.008 secs
NetValues propagation took 0.0 secs
Checking short circuits in 106 well contacts
   Additional analysis took 0.0 secs
No Well errors found (took 0.057 secs)
```

Section 5: LTSPICE code and parasitic extractions with calculation analysis:

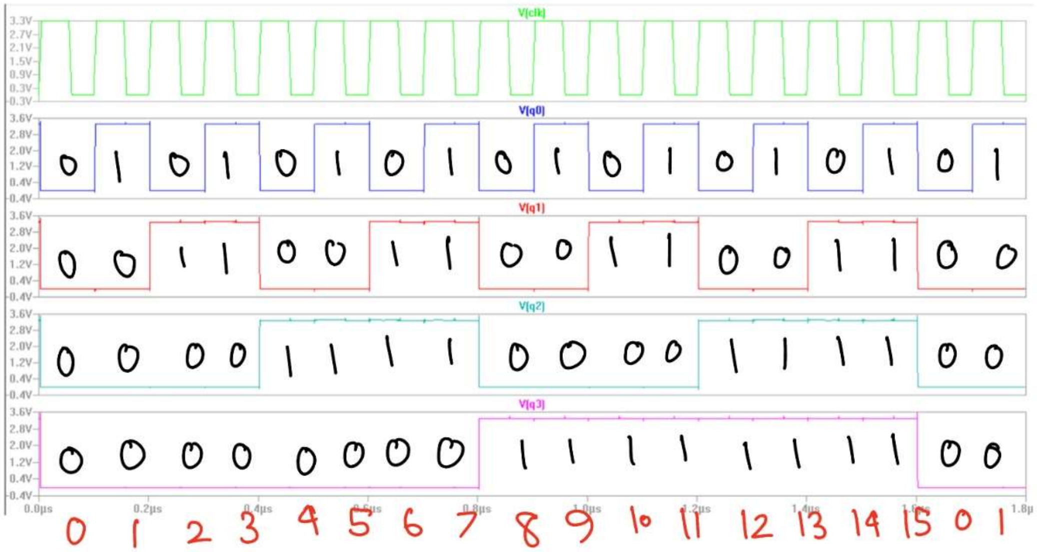
Schematic:

Spice Code:



```
VDD VDD 0 DC 3.3
VGND GND 0 DC 0
VClk Clk 0 DC PULSE 0 3.3 0n 5n 5n 50n 100n
.TRAN 0 1800n
.include C:\Users\mahir\EE457\cmosedu_models.txt
```

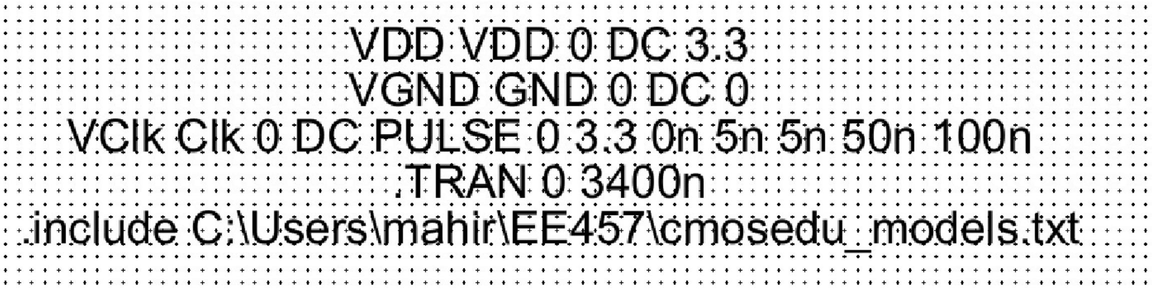
LTSPICE Simulation:



This is the LTSPICE simulation for electric schematic for the 4-bit up counter. Here all the outputs are at 0 making the first number 0000 which is 0 in decimal. When the first rising edge of the clock pulse comes in, the clock counts increment the 4-bit number to 0001 and it keeps incrementing it by 1 at every rising pulse of the clock. The number keeps going up to 15 as shown and then at the next rising edge pulse, the counter resets to 0 and starts counting again.

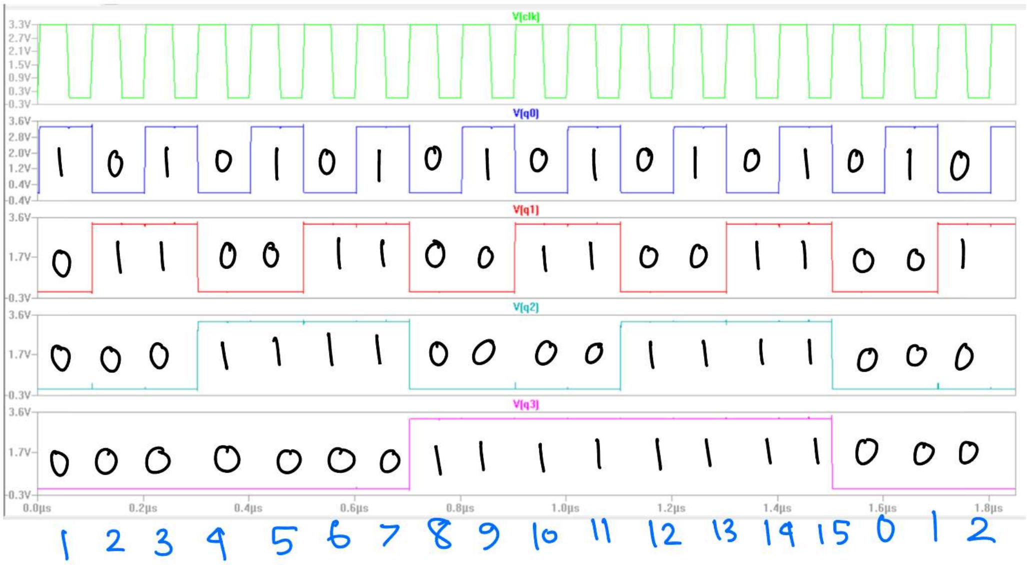
Layout:

Spice Code:



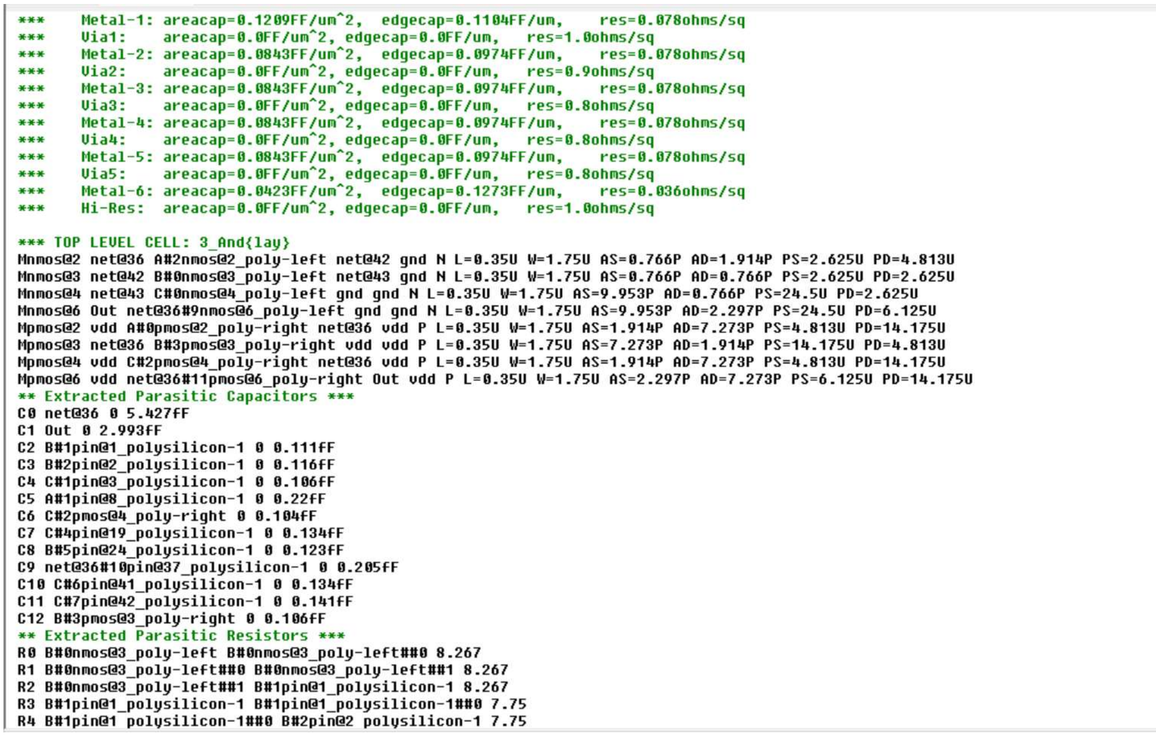
```
VDD VDD 0 DC 3.3
VGND GND 0 DC 0
VClk Clk 0 DC PULSE 0 3.3 0n 5n 5n 50n 100n
.TRAN 0 3400n
.include C:\Users\mahir\EE457\cmosedu_models.txt
```

LTSPICE Simulation:

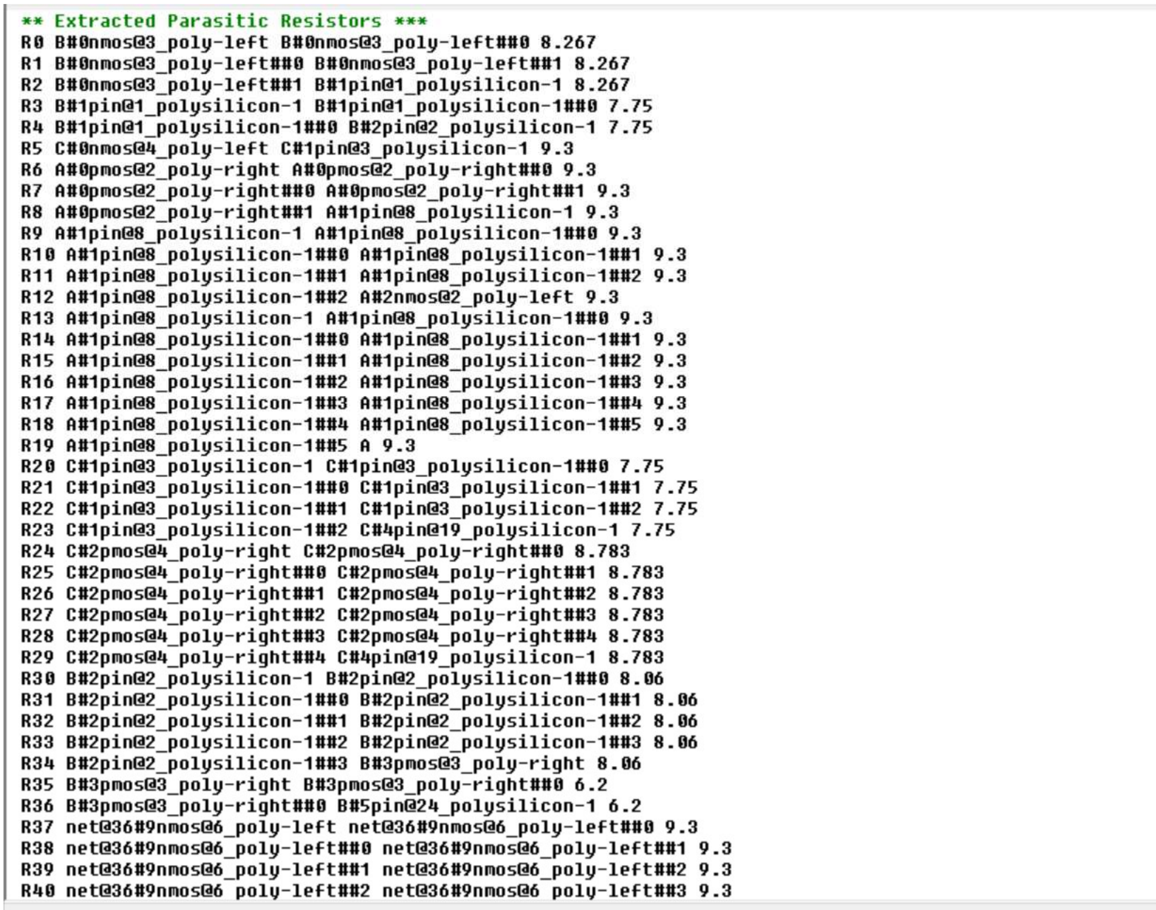


This is the LTSPICE simulation for the 4-bit counter. Here, when the rising clock pulse comes in, it starts counting. We see that it starts with Q0 being 1, and rest of the bits are 0 making the digit 0001 which is 1 in decimal. Then at each rising edge of clock the Q3-Q0 changes by an increment of 1 to count from 1 to 15 and once the next rising pulse comes in after the counter is at 15, it resets back to 0 and counts again.

Parasitic Extractions:



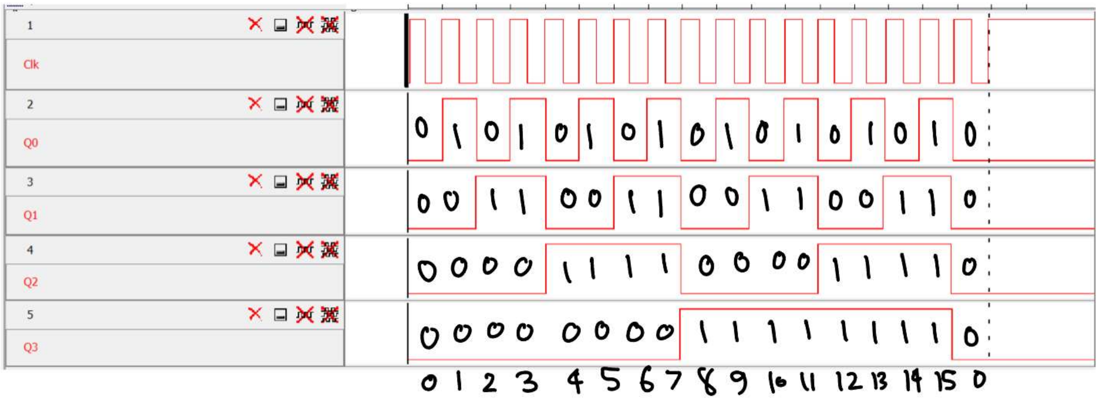
```
***     Metal-1: areacap=0.1209FF/um^2,  edgecap=0.1104FF/um,    res=0.078ohms/sq
***     Via1:    areacap=0.0FF/um^2, edgecap=0.0FF/um,   res=1.0ohms/sq
***     Metal-2: areacap=0.0843FF/um^2,  edgecap=0.0974FF/um,    res=0.078ohms/sq
***     Via2:    areacap=0.0FF/um^2, edgecap=0.0FF/um,   res=0.9ohms/sq
***     Metal-3: areacap=0.0843FF/um^2,  edgecap=0.0974FF/um,    res=0.078ohms/sq
***     Via3:    areacap=0.0FF/um^2, edgecap=0.0FF/um,   res=0.8ohms/sq
***     Metal-4: areacap=0.0843FF/um^2,  edgecap=0.0974FF/um,    res=0.078ohms/sq
***     Via4:    areacap=0.0FF/um^2, edgecap=0.0FF/um,   res=0.8ohms/sq
***     Metal-5: areacap=0.0843FF/um^2,  edgecap=0.0974FF/um,    res=0.078ohms/sq
***     Via5:    areacap=0.0FF/um^2, edgecap=0.0FF/um,   res=0.8ohms/sq
***     Metal-6: areacap=0.0423FF/um^2,  edgecap=0.1273FF/um,    res=0.036ohms/sq
***     Hi-Res:  areacap=0.0FF/um^2, edgecap=0.0FF/um,   res=1.0ohms/sq

*** TOP LEVEL CELL: 3_And{lay}
Mnmos@2 net@36 A#2nmos@2_poly-left net@42 gnd N L=0.35U W=1.75U AS=0.766P AD=1.914P PS=2.625U PD=4.813U
Mnmos@3 net@42 B#0nmos@3_poly-left net@43 gnd N L=0.35U W=1.75U AS=0.766P AD=0.766P PS=2.625U PD=2.625U
Mnmos@4 net@43 C#0nmos@4_poly-left gnd gnd N L=0.35U W=1.75U AS=9.953P AD=0.766P PS=24.5U PD=2.625U
Mnmos@6 Out net@36#9nmos@6_poly-left gnd gnd N L=0.35U W=1.75U AS=9.953P AD=2.297P PS=24.5U PD=6.125U
Mpmos@2 vdd A#0pmos@2_poly-right net@36 vdd P L=0.35U W=1.75U AS=1.914P AD=7.273P PS=4.813U PD=14.175U
Mpmos@3 net@36 B#3pmos@3_poly-right vdd vdd P L=0.35U W=1.75U AS=7.273P AD=1.914P PS=14.175U PD=4.813U
Mpmos@4 vdd C#2pmos@4_poly-right net@36 vdd P L=0.35U W=1.75U AS=1.914P AD=7.273P PS=4.813U PD=14.175U
Mpmos@6 vdd net@36#11pmos@6_poly-right Out vdd P L=0.35U W=1.75U AS=2.297P AD=7.273P PS=6.125U PD=14.175U
** Extracted Parasitic Capacitors ***
C0 net@36 0 5.427fF
C1 Out 0 2.993fF
C2 B#1pin@1_polysilicon-1 0 0.111fF
C3 B#2pin@2_polysilicon-1 0 0.116fF
C4 C#1pin@3_polysilicon-1 0 0.106fF
C5 A#1pin@8_polysilicon-1 0 0.22fF
C6 C#2pmos@4_poly-right 0 0.104fF
C7 C#4pin@19_polysilicon-1 0 0.134fF
C8 B#5pin@24_polysilicon-1 0 0.123fF
C9 net@36#10pin@37_polysilicon-1 0 0.205fF
C10 C#6pin@41_polysilicon-1 0 0.134fF
C11 C#7pin@42_polysilicon-1 0 0.141fF
C12 B#3pmos@3_poly-right 0 0.106fF
** Extracted Parasitic Resistors ***
R0 B#0nmos@3_poly-left B#0nmos@3_poly-left##0 8.267
R1 B#0nmos@3_poly-left##0 B#0nmos@3_poly-left##1 8.267
R2 B#0nmos@3_poly-left##1 B#1pin@1_polysilicon-1 8.267
R3 B#1pin@1_polysilicon-1 B#1pin@1_polysilicon-1##0 7.75
R4 B#1pin@1 polysilicon-1##0 B#2pin@2 polysilicon-1 7.75
```



```
** Extracted Parasitic Resistors ***
R0 B#0nmos@3_poly-left B#0nmos@3_poly-left##0 8.267
R1 B#0nmos@3_poly-left##0 B#0nmos@3_poly-left##1 8.267
R2 B#0nmos@3_poly-left##1 B#1pin@1_polysilicon-1 8.267
R3 B#1pin@1_polysilicon-1 B#1pin@1_polysilicon-1##0 7.75
R4 B#1pin@1_polysilicon-1##0 B#2pin@2_polysilicon-1 7.75
R5 C#0nmos@4_poly-left C#1pin@3_polysilicon-1 9.3
R6 A#0pmos@2_poly-right A#0pmos@2_poly-right##0 9.3
R7 A#0pmos@2_poly-right##0 A#0pmos@2_poly-right##1 9.3
R8 A#0pmos@2_poly-right##1 A#1pin@8_polysilicon-1 9.3
R9 A#1pin@8_polysilicon-1 A#1pin@8_polysilicon-1##0 9.3
R10 A#1pin@8_polysilicon-1##0 A#1pin@8_polysilicon-1##1 9.3
R11 A#1pin@8_polysilicon-1##1 A#1pin@8_polysilicon-1##2 9.3
R12 A#1pin@8_polysilicon-1##2 A#2nmos@2_poly-left 9.3
R13 A#1pin@8_polysilicon-1 A#1pin@8_polysilicon-1##0 9.3
R14 A#1pin@8_polysilicon-1##0 A#1pin@8_polysilicon-1##1 9.3
R15 A#1pin@8_polysilicon-1##1 A#1pin@8_polysilicon-1##2 9.3
R16 A#1pin@8_polysilicon-1##2 A#1pin@8_polysilicon-1##3 9.3
R17 A#1pin@8_polysilicon-1##3 A#1pin@8_polysilicon-1##4 9.3
R18 A#1pin@8_polysilicon-1##4 A#1pin@8_polysilicon-1##5 9.3
R19 A#1pin@8_polysilicon-1##5 A 9.3
R20 C#1pin@3_polysilicon-1 C#1pin@3_polysilicon-1##0 7.75
R21 C#1pin@3_polysilicon-1##0 C#1pin@3_polysilicon-1##1 7.75
R22 C#1pin@3_polysilicon-1##1 C#1pin@3_polysilicon-1##2 7.75
R23 C#1pin@3_polysilicon-1##2 C#4pin@19_polysilicon-1 7.75
R24 C#2pmos@4_poly-right C#2pmos@4_poly-right##0 8.783
R25 C#2pmos@4_poly-right##0 C#2pmos@4_poly-right##1 8.783
R26 C#2pmos@4_poly-right##1 C#2pmos@4_poly-right##2 8.783
R27 C#2pmos@4_poly-right##2 C#2pmos@4_poly-right##3 8.783
R28 C#2pmos@4_poly-right##3 C#2pmos@4_poly-right##4 8.783
R29 C#2pmos@4_poly-right##4 C#4pin@19_polysilicon-1 8.783
R30 B#2pin@2_polysilicon-1 B#2pin@2_polysilicon-1##0 8.06
R31 B#2pin@2_polysilicon-1##0 B#2pin@2_polysilicon-1##1 8.06
R32 B#2pin@2_polysilicon-1##1 B#2pin@2_polysilicon-1##2 8.06
R33 B#2pin@2_polysilicon-1##2 B#2pin@2_polysilicon-1##3 8.06
R34 B#2pin@2_polysilicon-1##3 B#3pmos@3_poly-right 8.06
R35 B#3pmos@3_poly-right B#3pmos@3_poly-right##0 6.2
R36 B#3pmos@3_poly-right##0 B#5pin@24_polysilicon-1 6.2
R37 net@36#9nmos@6_poly-left net@36#9nmos@6_poly-left##0 9.3
R38 net@36#9nmos@6_poly-left##0 net@36#9nmos@6_poly-left##1 9.3
R39 net@36#9nmos@6_poly-left##1 net@36#9nmos@6_poly-left##2 9.3
R40 net@36#9nmos@6 poly-left##2 net@36#9nmos@6 poly-left##3 9.3
```

Section 6: IRSIM Logic Simulations and Measurements for Layout and Schematic

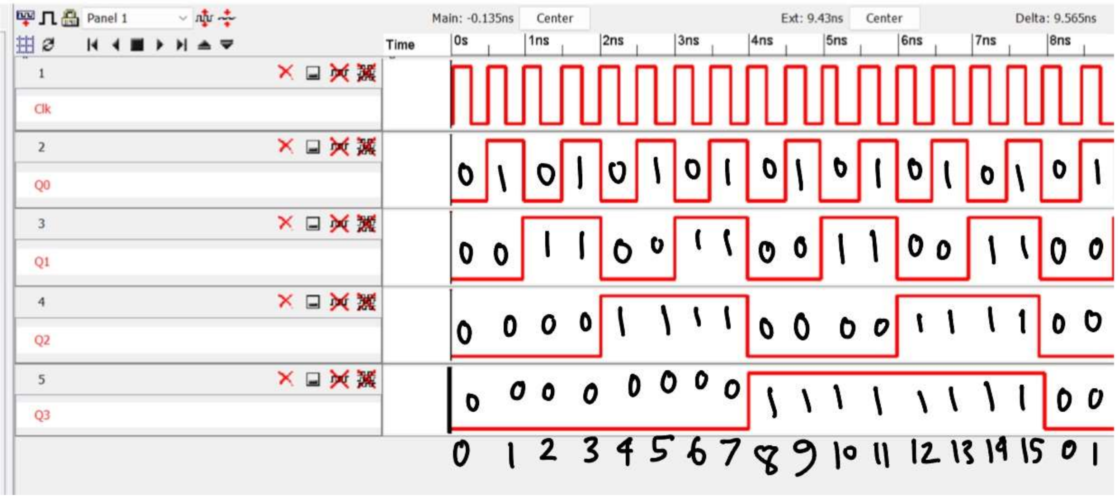
Schematic:



This is the IRSIM logic simulation for the transistor circuit in schematic. Here we started by setting Q3-Q0 to 0000 and at the next rising clock pulse the number increments from

0000 to 0001. The number consistently increases by 1 at each rising edge signal of the clock thus making it an up counter. Since our counter is 4-bit, the highest unsigned binary number it represents is 1111 which is 15 in decimal and then at the next rising edge of the clock it resets back to 0 and starts and keeps counting in a cycle of 0 to 15.

Layout:



This is the IRSIM logic simulation for the layout circuit in electric. The outputs are initially set to all zeroes making it 0000 and at the next rising clock the number increments from 0000 to 0001. The number counter constantly increases by 1 whenever te clock pulse is at each rising edge. Since our counter is 4-bit, the highest unsigned binary number represents decimal number 15 as 1111 in binary and then at the next rising edge of the clock it resets back to 0 and then keeps counting in a cycle of 0 to 15.

Section 7: Measurements in LTSPICE for delays for Layout and Schematic:

Propagation delay: $tP = \frac{(tPHL+tPLH)}{2}$ (Average of low-high delay and high-low delay)

	Rise Time	Fall Time	TPHL	TPLH	Propagation Delay (TP)
CMOS Schematic	4.533 ns	2.873 ns	1668.75 ns	802.32 ns	1235.535 ns
CMOS Layout	6.827 ns	3.016 ns	1753.76 ns	968.98 ns	1361.37 ns

As seen from the data table above, the delay from high to low, the delay from low to high and the propagation delay are all higher in the electric layout design compared to electric schematic design of the counter. This comparison is justifiable because our layout design consists of more complex paths and wiring such as use of different metals and different contacts. The size of our layout area is also bigger which causes longer delays in transferring and updating data.

Section 8: Measurements of power, delay, chip area, timing, number of transistors for the layout:

CMOS Schematic:

Chip Size (W x L): 659.25 x 623.25 (lambda)
Transistor Size: 175 nm

Chip Width = 659.25 lambda x 175 nm
= 115368.75 nm
= 115.369 µm
Chip Length = 623.25 lambda x 175 nm
= 109068.75 nm
= 109.0688 µm
Chip Area = 115.369 µm x 109.0688 µm
= 12583.16 µm^2

Transistor Count:
2_input AND gate = 6 (one such gate)
3_input AND gate = 8 (one such gate)
XOR gate = 14 (three such gates)
D Flip-flop = 40 (four such gates)

CMOS Layout:

Chip Size (W x L): 1615 x 1319.5 (lambda)
Transistor Size: 175 nm

Chip Width = 1615 lambda x 175 nm
= 282625 nm
= 282.625 µm
Chip Length = 1319.5 lambda x 175 nm
= 230912.5 nm
= 230.9125 µm
Chip Area = 282.625 µm x 230.9125 µm
= 65261.65 µm^2

Transistor Count:
2_input AND gate = 6 (one such gate)
3_input AND gate = 8 (one such gate)
XOR gate = 14 (three such gates)
D Flip-flop = 40 (four such gates)

	Transistor size (W x L)	Transistor Count	Chip Area
CMOS Schematic	PMOS (10 x 2) NMOS (10 x 2)	6 + 8 + (14*3) + (40*4) = 216	12583.16 µm^2
CMOS Layout	PMOS (10 x 2) NMOS (10 x 2)	6 + 8 + (14*3) + (40*4) = 216	65261.65 µm^2

Section 9: Pathwave ADS simulations for RC circuits:

No Data Available for this section. Due to licensing issues and time constraints, unfortunately we couldn't perform the task properly on time. However, the parasitic extractions are successfully completed as shown above so with access to functional pathwave software, the ADS simulation can be done at any time to further our analysis on our layout circuit of 4-bit up counter suing D flip flops.

Section 10: Conclusion and References

Conclusion:

We have reached the end of this experiment of creating a 4-bit synchronous up counter using D flip flops. During this experiment, we have been able to identify the functionality and structure of the 4-bit counter as well as the truth table of the circuit. Next, we successfully built all the components such as 2 input AND gates, 3 input AND gates, inverters, XOR gates, NAND gates by using digital transistors in electric schematic. We have then also created a master slave D flip flop using two d latches (consisting of 4 NAND gates and an inverter) and made the D flip flop circuit to function on the rising edge of the clock signal. In the final step of our schematic design, we have been able to connect the AND gates, XOR gates and 4 D flip flops as explained above to construct the full 4-bit synchronous up counter circuit. We successfully performed DRC (Design Rule Check) to make sure our circuit has no issues. We then proceeded to replicate and recreate the same design in electric layout using PMOS, NMOS, metal wires and contacts. Following the same logic we have used to build the schematic circuit, we have built all necessary gate and flip flop components and connect them together to construct the up counter. We have successfully run a DRC and ERC check to ensure our design carries no design issues or well/contact issues. In the next step of our experiment, we verified if our design works as intended by using LTSPICE simulation for both layout and schematic. We have written the spice code as shown above and ran simulations to check that our 4-bit counter indeed counts from 0 to 15 and then resets back to as intended. We have also verified both our schematic and layout design in IRSM logic simulation, where we have been successfully able to run simulations once again to show the counter properly counts from 0 to 15. In addition to running simulations, we have also done a parasitic extraction from our LTSPICE code which demonstrates all the RC values of our circuit and calculation analysis as shown above. IN the following sections, we have made some comparisons between our schematic and layout designs. First, we have properly calculated the rise and fall time for both circuits as well as calculate the propagation delay from high to low, low to high and the average and we have established why the layout has more propagation delay than digital transition. Then, we made a chip size and chip area comparison for both circuits as well through the calculation of length and width of our circuit areas that was demonstrated above as well. Due to lack of proper technological access, we haven't been able to the pathwave simulation of the RC circuit however, overall, we have gathered plenty of information to display our knowledge and understanding of the synchronous up counter design through circuit construction and proper simulation verification and therefore, this experiment has allowed us to expand our knowledge on complex integrated chip designing even further.

References:

1) Urias, O. M. (2023, February 21). *The D flip-flop (QuickStart tutorial)*. Build Electronic Circuits. https://www.build-electronic-circuits.com/d-flip-flop/

2) 17. the BCD (MOD10) synchronous up counter circuit constructed with D... | download scientific diagram. (n.d.). https://www.researchgate.net/figure/The-BCD-MOD10-synchronous-up-counter-circuit-constructed-with-D-flip-flops_fig19_319203501

3) Lab6 - designing NAND, nor, and XOR gates for use to design full-adders. (n.d.). https://cmosedu.com/jbaker/courses/ee421L/f13/students/wolvert9/Lab%206/Lab6.html

4) *Q. 6.17: Design a four-bit binary synchronous counter with D Flip-flops || Complete Design Steps*. YouTube. (2020, June 10). https://youtu.be/pZ6OP2muIZg?si=TGWNSPcbRHKtPS3c