LABORATORY 2

DIGITAL LOGIC COMPONENTS

OBJECTIVES

No.	Topics	Requirements
1	Design CMOS Logic Gates.	▪ Implement CMOS-based NAND2, NOR2, and EXOR2 gates.
2	Implement Basic CMOS Combinational Components	▪ Explore the design and functionality of a 2-to-1 multiplexer.
3	Implement Simple Sequential Storage Elements	▪ Investigate the operation of a single positive-edge-triggered Modified TSPC D flip-flop.
4	**[OPTIONAL]** Pass Transistors, Transmission Gates, and Tristate Inverters: Operation and Analysis	▪ Analyze the operation of pass transistors, transmission gates, and tristate inverters.

EXPERIMENT 1

Objective: Implement CMOS-based logic gates.

Requirements:

- Complete the truth table, schematic, and symbol for each component.
- Run DC analysis and transient simulation.
- Create layouts for each logic gate, then show DRC confirmation and corresponding schematic with proof of LVS.

Instructions:

- In general, students should present their work in four parts: schematic, DC analysis, transient simulation, and layout.

a. Schematic:

The truth table of an inverter:

INPUT	OUTPUT
A	Y
0	
1	

Table 1 The truth table of an inverter.

The schematic of a CMOS inverter (please show us how it works):

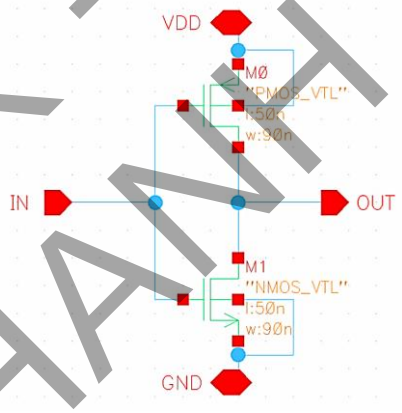


Figure 1 The schematic of a CMOS inverter.

The symbol of inverter:

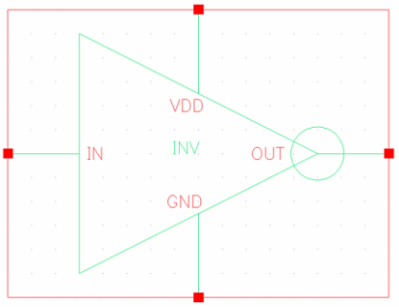


Figure 2 Symbol of INV

b. DC Analysis simulation:

Use **ADE-L** to simulate the DC response of an inverter (INV) gate. Apply an input signal as a ramp voltage ranging from 0V to 1V or perform a voltage sweep from 0V to 1V. Observe the corresponding output response. The circuit parameters are configured as follows:

Parameters	Value
V_{dd}	$1V$
C_{load}	$1fF$
V_{in}	$[0, 1]V$

Table 2 Parameters in DC analysis

During the simulation process, students record the following two results: output voltage values at various values of V_{in} with a 0.1V step and plot the curve of V_{out}. Besides, *students can decrease the step of* V_{in} *to draw a curve of* V_{out} *more precise*.

V_{in} (V)	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
V_{out} (V)										

Table 3 The output voltage values at various values of V_{in} with 0.1V step.

Figure 3 Voltage transfer curve (VTC) of CMOS inverter.

c. Transient simulation:

Use ADE-L to perform a time-domain simulation to verify the operation according to the truth table of the INV gate, represented in the form of an output waveform. To conduct this simulation, assemble a testbench circuit consisting of a pulse source (vpulse), an output capacitor, and provide power to the circuit (vdc). The circuit parameters are configured as follows:

Parameters	Value	Note
V_{dd}	$1V$	
C_{load}	$1fF$	
Voltage 1	$0V$	Parameters for vpulse
Voltage 2	$1V$	
Rise time	$1ps$	
Fall time	$1ps$	
Delay	$0ns$	
Pulse width	$1ns$	
Period	$2ns$	

Table 4 Parameters for transient simulation.

Testbench circuit for INV

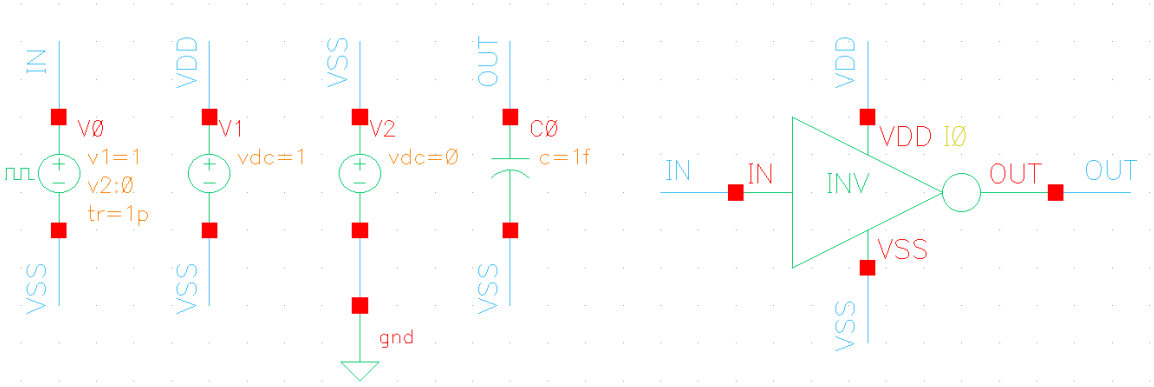


Figure 4 The testbench of a CMOS inverter.

During simulation process, students record input and output waveforms. Then, students measure some timing measurements.

Parameters	Result
t_{rise} – Rising time (10% – 90%)	
t_{fall} – Falling time (90% – 10%)	
t_{pdr} – Rising propagation delay (90% – 50%)	
t_{pdf} – Falling propagation delay (10% – 50%)	
t_{pd} – Average propagation delay (50% - 50%)	
Dynamic power	
Static power	

Table 5 Measurement requirements for an inverter.

d. Layout of INV:

Show the stick diagram and the complete layout of a CMOS inverter with DRC and LVS confirmation.

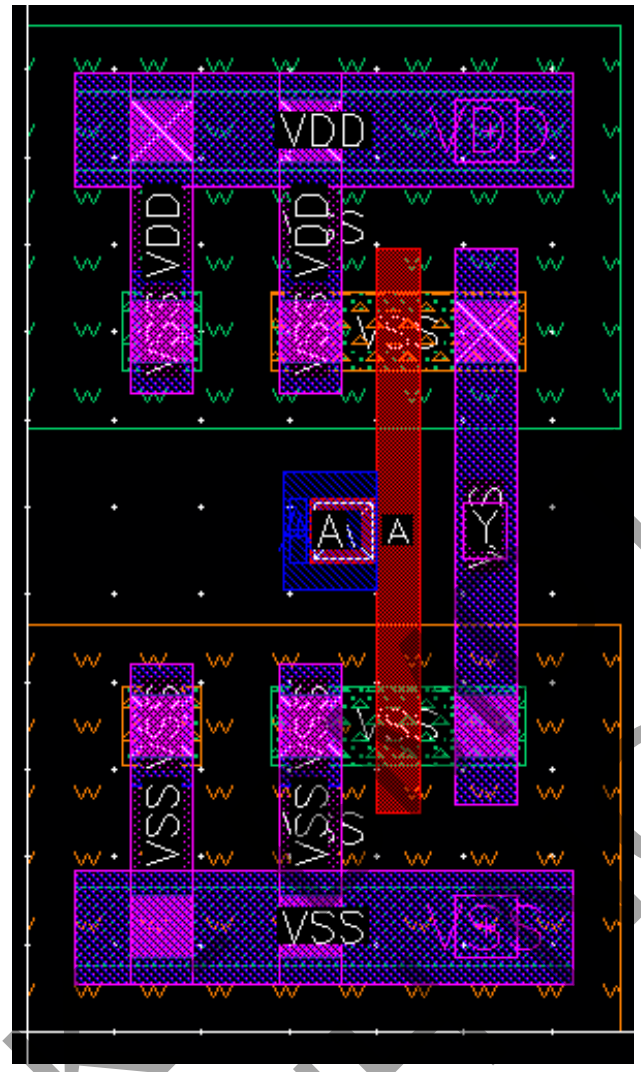


Figure 5 The complete layout of a CMOS inverter.

Check: Students must show these results in the report.

➢ The waveform to prove the circuit works correctly. Besides, students apply these parameters in **vpulse** and then simulate design.

	In1	**In2**
Voltage 1	0	0
Voltage 2	1	1
Period	4n	2n
Delay time	0.65n	0.8n
Rise time	1p	1p
Fall time	1p	1p
Pulse width	2n	1n

Table 6 Testbench for NAND2, NOR2, and EX-OR2.

➢ Using the simulation result above, students measure some parameters shown in **Table 7** for each logic gate.

Parameters	Result
t_{rise} – Rising time (10% – 90%)	
t_{fall} – Falling time (90% – 10%)	
t_{pdr} – Rising propagation delay (90% – 50%)	
t_{pdf} – Falling propagation delay (10% – 50%)	
t_{pd} – Average propagation delay (50% - 50%)	
Power consumption	

Table 7 Requirements for measuring NAND2, NOR2, and EX-OR2.

(Hint: Students can use built-in functions in Calculator and then re-use it for other designs)

EXPERIMENT 2

Objective: Implement Basic CMOS Combinational Components.

Requirements:

- Complete the truth table, schematic, and symbol for a 2-to-1 channel multiplexer using compound gate.
- Run DC analysis and transient simulation.
- Create layouts for each logic gate, then show DRC confirmation and the corresponding schematic with proof of LVS.

Remind:

- Multiplexers are the key components in CMOS memory elements and data manipulation structures. A multiplexer selects the output from among several inputs based on a specific signal. First, students complete the truth table and give the logic function of a 2-to-1 channel multiplexer.
- This section presents some basic topologies of 2-to-1 MUX, such as transmission gate, compound gate, and tristate inverter.
  - Two transmission gates can be tied together to form a compact 2-input multiplexer. However, the transmission gates produce a non-restoring multiplexer, and another issue is charge sharing from the output when S toggles from high to low.
  - A restoring, inverting multiplexer: compound gate or gang together two tristate inverters.
  - In practice, both inverting and noninverting multiplexers are simply called multiplexers or muxes.

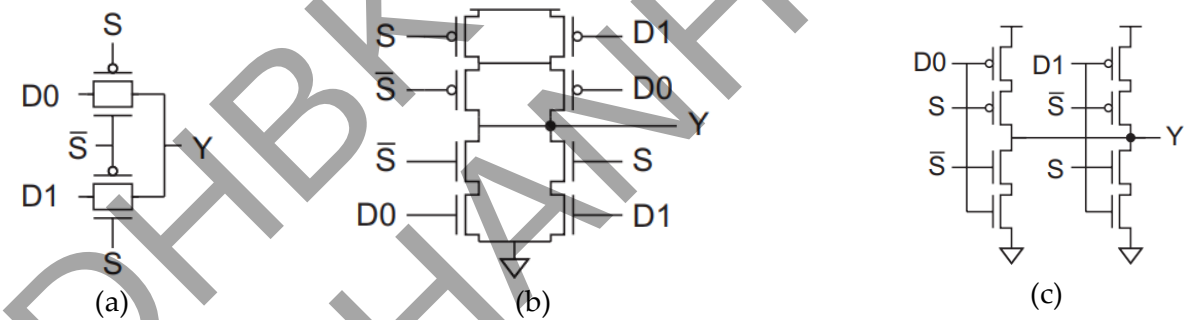


Figure 6 MUX 2-to-1 using (a) transmission gate (b) compound gate (c) tristate inverter. Source: from [1]

- Besides, the unit MUX can be composed of various types of logic gates. The detailed circuitry and pros and cons of each logic gate are summarized in the figure below:

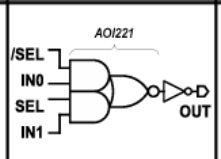
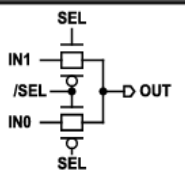
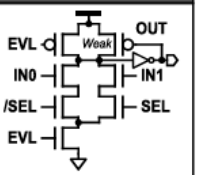
Type	Static CMOS	Transmission Gate	Domino Logic
Scheme			
Pros	Low Power	Compact Size	Fast Operation
Cons	Large Area	No Self Driving	High Power

Figure 7 Various types of unit MUX. Source: from [2]

Check: Students must show these results in the report.

- The waveform to prove the 2-to-1 channel multiplexer using *compound gate* works correctly, and measured parameters are shown in **Table 8**.

Parameters	Result
t_{rise} – Rising time (10% – 90%)	
t_{fall} – Falling time (90% – 10%)	
t_{pdr} – Rising propagation delay (90% – 50%)	
t_{pdf} – Falling propagation delay (10% – 50%)	
t_{pd} – Average propagation delay (50% - 50%)	
Power consumption	

Table 8 Measurement results of MUX 2-to-1

- Complete the layout of your schematic which must pass DRC and LVS.

EXPERIMENT 3

Objective: Implement simple storage elements - single-positive-edge-triggered Modified TSPC D flip-flop.

Requirements:

- Demonstrate latch and flip-flop by studying a D latch and a D flip-flop using transmission gates.
- Complete the truth table, schematic, and symbol for each component, then run the transient simulation.
- Known definitions and how to measure setup time, propagation, and hold time.
- Find the clock frequency of this element.

Instructions:

- Building on the previously developed combinational components, sequential circuits such as latches and flip-flops can now be designed. Figure 8(a) depicts a D latch implemented using a 2-input multiplexer, which itself can be realized using a pair of transmission gates, as shown in Figure 8(b).

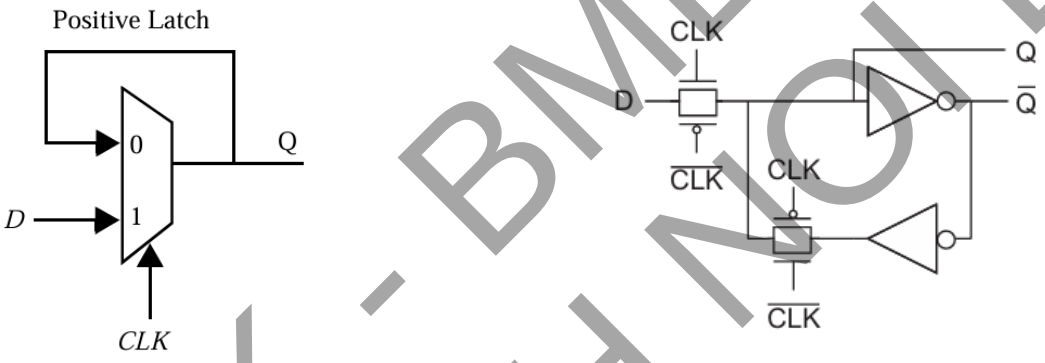
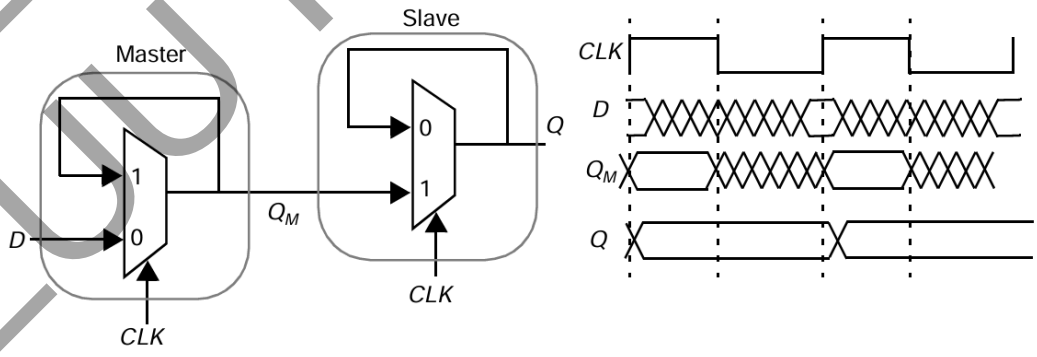
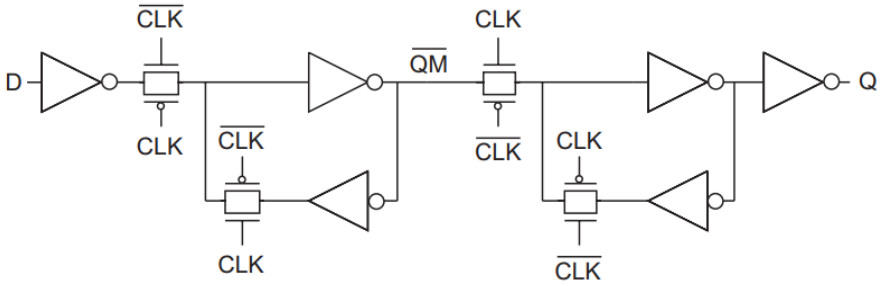


Figure 8 (a) Positive D-latch based on multiplexer. Source: from [3] ; (b) CMOS positive-level-sensitive D latch. Source: from [1]

- By combining two level-sensitive latches—one negative-sensitive in the first stage and one positive-sensitive in the second stage—a positive-edge-triggered flip-flop is formed. As shown in **Figure 9(a)**, the first latch, known as the master, captures the input data, while the second latch, called the slave, transfers the data on the clock's rising edge. This device is commonly referred to as a D flip-flop, D register, or master-slave flip-flop.



(a)

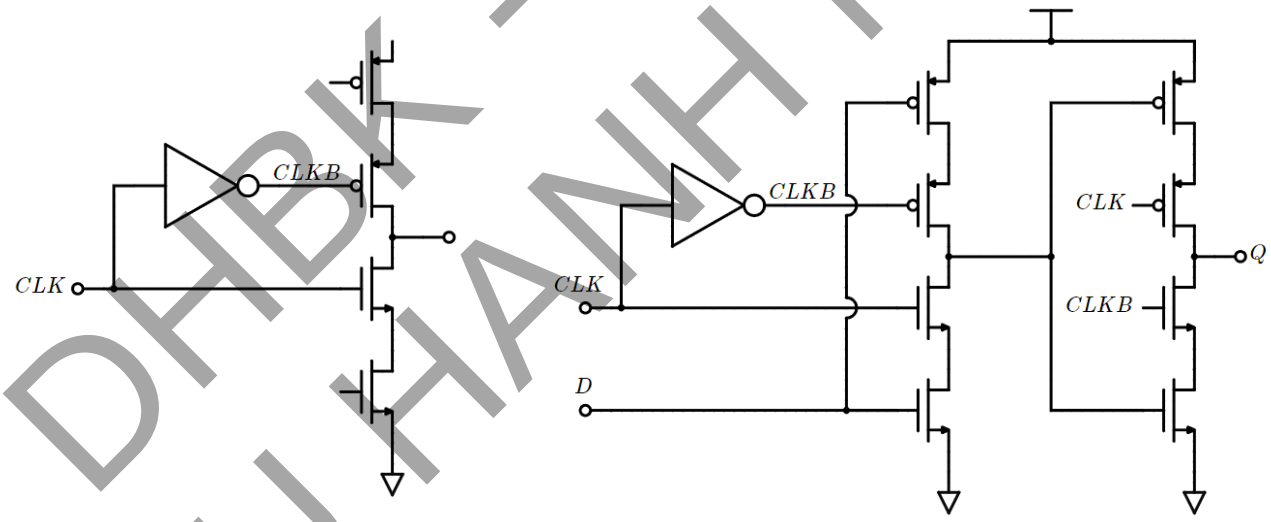


(b)

Figure 9 (a) Positive edge-triggered register based on a master-slave configuration. Source: from [3]; (b) Schematic of master-slave positive edge-triggered register using multiplexers. Source: from [1]

➢ The negative edge-triggered flip-flop is based on a master-slave architecture and is designed to be insensitive to clock overlap. This design, known as the Clocked CMOS (C2MOS) flip-flop, incorporates dynamic latches for efficient operation. As shown in **Figure 10(a)**, the C2MOS flip-flop consists of dynamic latch topologies, where the data signal is processed by PMOS transistors at the top and NMOS transistors at the bottom. This configuration functions as a clocked gate, integrating both logic and latch operations.

- When the clock signal (**CLK**) is HIGH (1), the input is inverted, and the output is generated. Conversely, when **CLK** is LOW (0), the output enters a high-impedance state, preventing data propagation. **Figure 10(b)** illustrates the schematic of a C2MOS D flip-flop, which consists of two cascaded C2MOS latches, ensuring reliable data storage and transfer..



(a) (b)

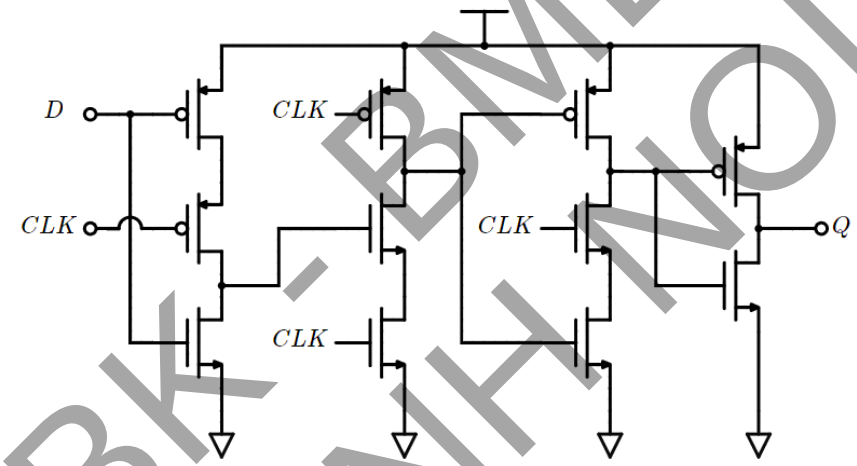
Figure 10 (a) A clocked CMOS latch; (b) The schematic of C2MOS negative-edge-triggered DFF. Source: Adapted from [3]

➢ True Single-Phase Clock (TSPC) [4] is a common dynamic Flip-Flop design that offers higher speed, smaller area, and lower power dissipation than other Flip-Flop architectures. Due to the requirement for sharp clock edge transitions and the stronger pull-up provided by PMOS devices, the design operates as a positive-edge-triggered Flip-Flop.

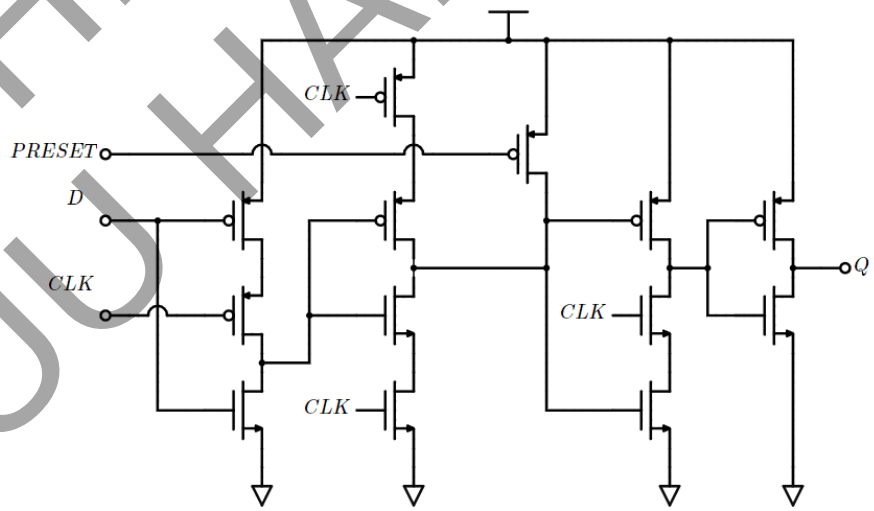
- When the clock signal CLK is 0, the input is isolated from the output, as the third stage's input is precharged to HIGH, allowing Q to retain its previous value. When CLK is 1, the

third stage's input remains unaffected. Therefore, as long as CLK is stable at either HIGH or LOW, the input remains isolated from the output. However, during a LOW-to-HIGH clock transition, QB latches the complement of the input while Q propagates the input to the output.

- Analyzing the behavior of node B reveals that whenever there is a discharge path to ground, the third stage's input is always precharged to HIGH when CLK is LOW and resets to LOW when CLK is HIGH. Consequently, if the input D remains at a stable LOW for an extended period, the third stage undergoes continuous toggling. This unnecessary switching not only contributes to excessive power consumption but also introduces noise at the output node Q, causing spurious glitches during each LOW-to-HIGH clock transition.
- A simple yet effective technique to mitigate this issue is to incorporate a PMOS transistor, which prevents the precharging phase from occurring without affecting the global operation of the Flip-Flop (Figure 11(b)).



(a)



(b)

Figure 11 The schematic of (a) negative-edge-triggered TSPC DFF (b) positive-edge-triggered Modified TSPC DFF with Preset.

➢ The Dual Edge-Triggered Flip-Flop (DETFF) samples data on both the rising and falling edges of the clock signal. Its design is based on multiplexers and consists of two distinct data paths.

- The upper path features a Single Edge-Triggered (SET) flip-flop implemented using PMOS pass transistors, which operate on the falling edge of the clock. Similarly, the lower path consists of an SET flip-flop but utilizes NMOS pass transistors, functioning on the rising edge of the clock.
- Both data paths are connected to feedback loops formed by two cascaded inverters. Additionally, each path includes a transistor (NMOS for the upper path, PMOS for the lower path) positioned after the loop. This transistor acts as a switch, controlled by the inverted clock signal.

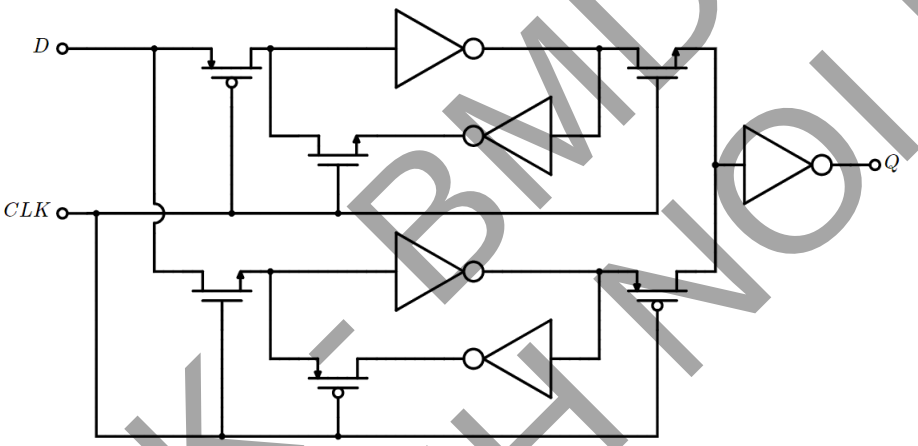


Figure 12 Schematic of D Flip-Flop dual edge triggered.

➢ In this experiment, students use Cadence Virtuoso to measure the setup and hold time of a D flip-flop. Recalling that the setup and hold time is the minimum time before and after the rising-edge clock the input signal must remain constant to store the signal and to generate a stable output, respectively.

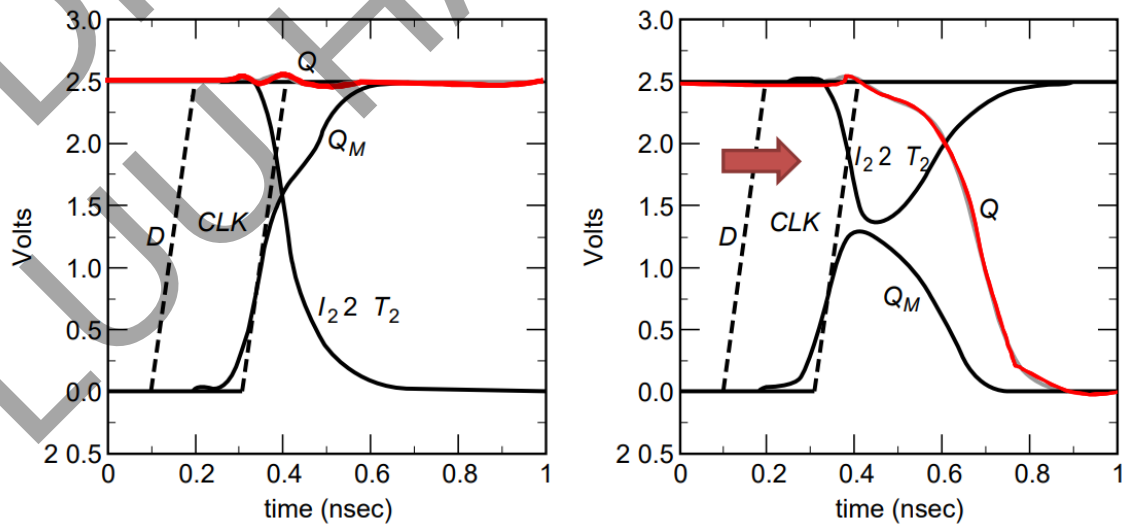


Figure 13 Overview measurement setup and hold time method. Source: from [3]

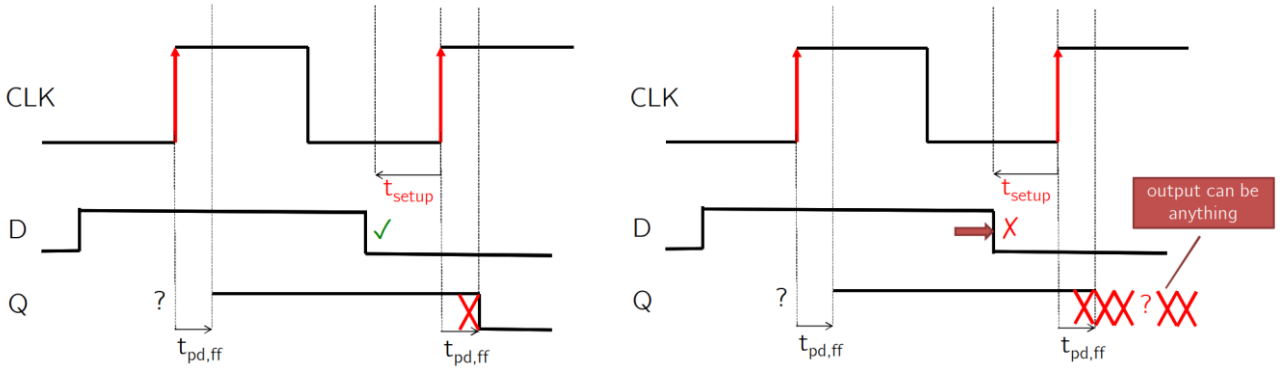


Figure 14 Waveform describing measurement setup time method. Source: from [4]

➢ Students can measure setup and hold time using this method: ***Shift input D closer to the rising CLK signal until Q output is incorrect.***

➢ In addition, the table below defines the delays and timing constraints of the combinational logic and sequencing elements.

Term	Name
t_{pd}	Logic Propagation Delay
t_{cd}	Logic Contamination Delay
t_{pcq}	Flop Clock-to-Q Propagation Delay
t_{ccq}	Flop Clock-to-Q Contamination Delay
t_{setup}	Flop Setup Time
t_{hold}	Flop Hold Time

Table 9 Sequencing element timing notation.

Figure 15(a) shows the response of the combinational logic to the input A changing from one arbitrary value to another. **Figure 15(b)** shows the response of the flip-flop.

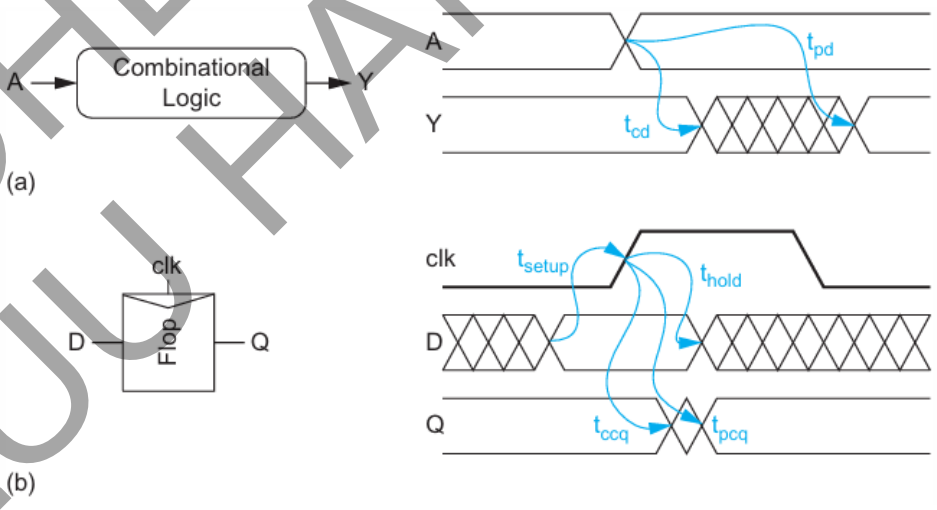


Figure 15 Timing diagrams. Source: from [1]

Check: You must show in your report these results.

➢ Simulate the positive-edge-triggered TSPC D flip-flop with Preset shown in **Figure 11(b)**. In addition, measure some timing parameters, and power consumption.

Parameters	Result
t_{setup} – Flop setup time	
t_{hold} – Flop hold time	
t_{pcqr} – Flop Clock-to-Q Propagation Delay (Low to High)	
t_{pcqf} – Flop Clock-to-Q Propagation Delay (High to Low)	
t_{pcq} – Flop Clock-to-Q Propagation Delay (average)	
Power consumption	

Table 10 Measurement results of positive-edge-triggered TSPC D flip-flop.

EXPERIMENT 4 [OPTIONAL]

Objective: Review the operation of pass transistor, transmission gate, and tristate inverter.

Requirements

- Analyze the charge and discharge period of MOS transistors, then prove by simulation.
- Analyze operations of a transmission gate, then prove by simulation.
- Analyze operations and topologies of a tristate inverter, then prove by simulation.

Instructions:

- Schematic, symbol, and operation description of pass transistor, transmission gate, and tristate inverter.

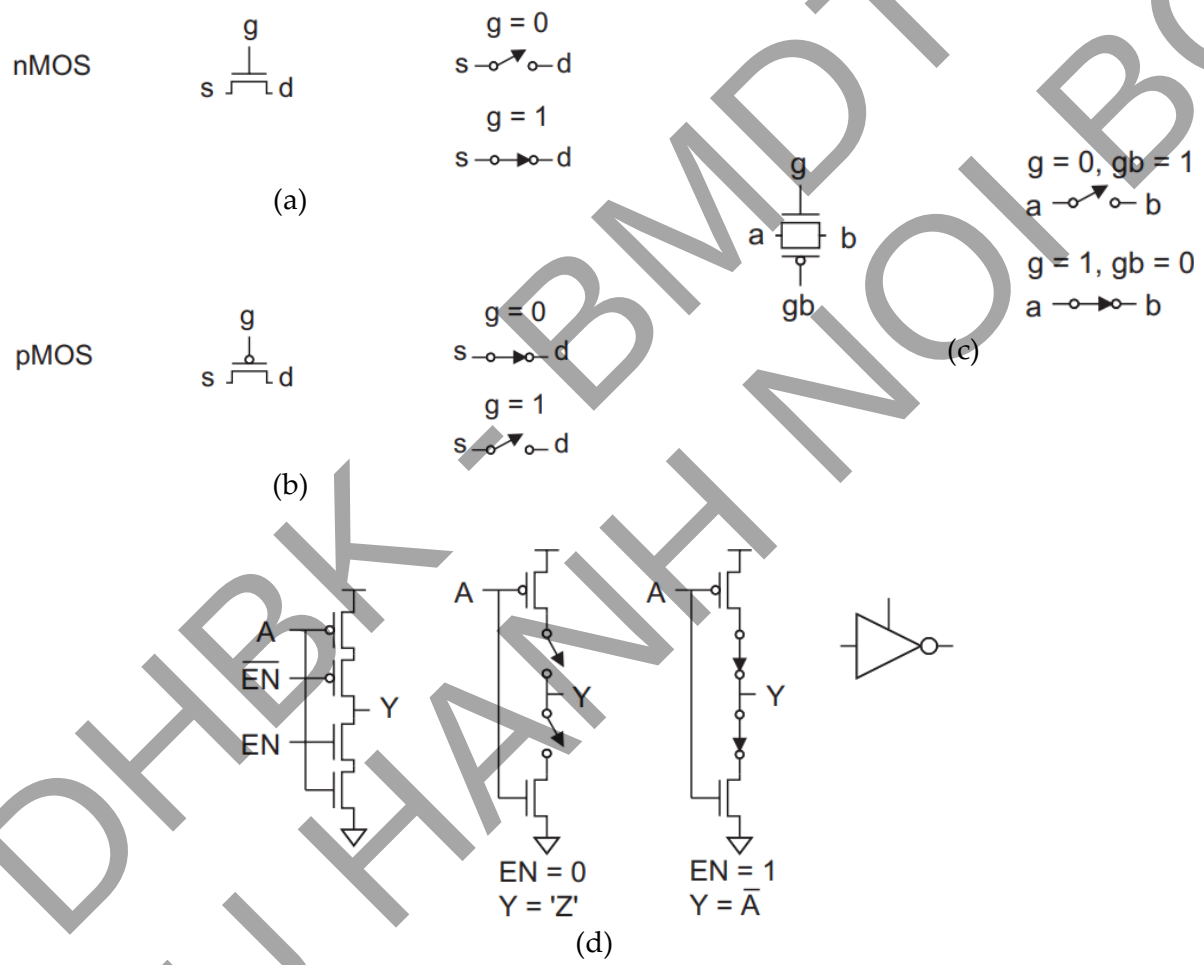


Figure 16 Schematic and operation of (a) pass transistor NMOS (b) pass transistor PMOS (c) transmission gate (d) tristate inverter. Source: from [1]

- Following these schematics shown in **Figure 17** and **Figure 18**, students need to calculate and define when NMOS turns on and determines $v_C(0^-)$ value (shown in **Figure 17**), and vice versa PMOS (shown in **Figure 18**).

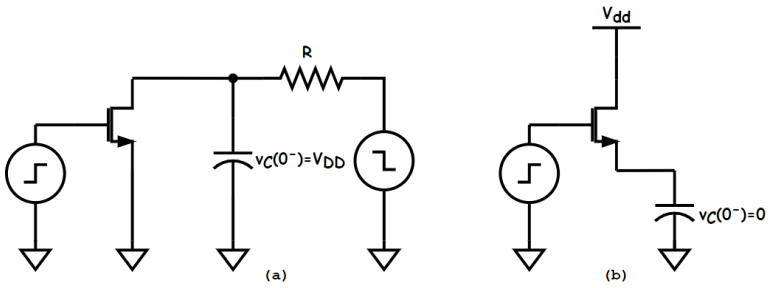


Figure 17 Testbench when (a) NMOS discharging (b) NMOS charging.

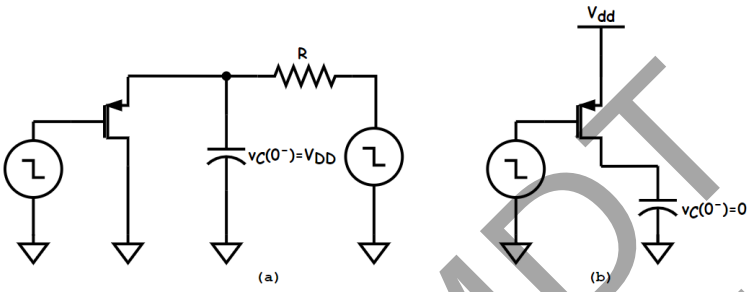


Figure 18 Testbench when (a) PMOS discharging (b) PMOS charging.

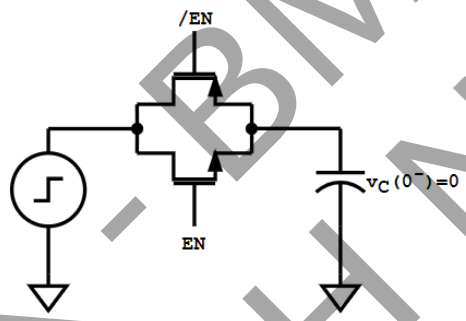


Figure 19 Schematic of transmission gate (testbench)

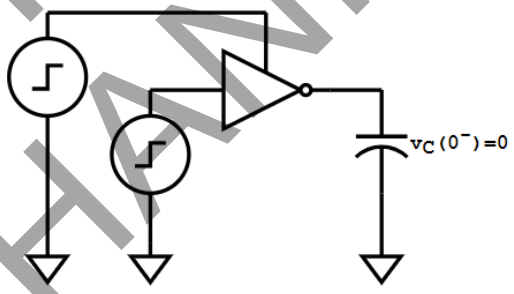


Figure 20 Schematic of tristate inverter (testbench)

Check:

➢ Students create schematics then simulate and show waveforms of the circuit as shown in **Figure 17** and **Figure 18**, then re-check calculations students perform in the instruction section.

➢ Students create schematics and then simulate and show waveforms of the circuit as shown in **Figure 19** and **Figure 20**.

Questions:

➢ According to the first check requirement, please let us know in which cases you use MOS transistors as a switch.

➢ The second check requirement shows us some advantages and disadvantages when transmission gates are used as a switch.

➢ Comment on the position of Drain and Source to Gate in MOSFET's structure. How can you demonstrate them? In your opinion, data, transmitted in a pass transistor and transmission gate, can just be flowed in one way, cannot it? Explain your idea.

➢ A student tries to move signal A, connected to M1 and M4 in **Figure 21(a)**, to M24 and M3 shown in **Figure 21(b)**, and he realizes that the function of the circuit does not change. Do you agree with his thoughts? Why do people prefer to use topology in **Figure 21(a)** instead of **Figure 21(b)**? Explain your opinion.

➢ Propose a method to design a tristate buffer.

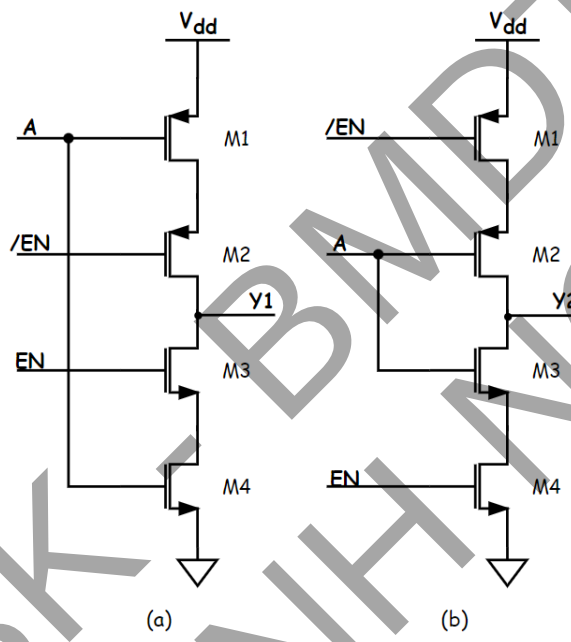


Figure 21 (a) schematic of the tristate inverter (b) schematic of the proposed tristate inverter.

ACKNOWLEDGEMENTS

This laboratory has been developed and refined through the contributions of numerous individuals over the years. It was originally designed for the *EE3117 Digital IC Design* course at the University of Technology – VNU, HCM by *Mr. Nguyen Trung Hieu* (2015) and *Mr. Duong Quang Ho* (2018). Subsequent revisions and enhancements have been implemented for the EE3117 Digital IC Design course under the guidance of Dr. Linh Tran, with additional contributions from Doanh Bui, Tan-Khai Pham, and Phuc T. Nguyen-Phan.

Versions of this lab have been used in the following courses:

- EE3117 Digital IC Design (2023 – present) – the University of Technology – VNU, HCM.
- EE5154 Advanced Digital IC Design (2024 – present) – the University of Technology – VNU, HCM.

REFERENCES

[1] N. H. E. Weste and D. M. Harris, "CMOS VLSI Design 4th Ed." Accessed: Feb. 24, 2025. [Online]. Available: https://pages.hmc.edu/harris/cmosvlsi/4e/index.html
[2] D. Chon and W. Choi, "Hardware Efficient Transposable 8T SRAM for Orthogonal Data Access," *IEEE Access*, vol. 12, pp. 981–992, 2024, doi: 10.1109/ACCESS.2023.3347913.
[3] J. M. Rabaey, A. Chandrakasan, and B. Nikolíc, "Digital Integrated Circuits Second Edition." Accessed: Apr. 03, 2025. [Online]. Available: https://icbook.eecs.berkeley.edu/resources/powerpoint-slides
[4] "ECE 4740 Course Details – Cornell ECE Open Courseware." Accessed: Apr. 03, 2025. [Online]. Available: https://ocw.ece.cornell.edu/ece-4740-course-details/