ENR-325/325L Principles of Digital Electronics and Laboratory

Xiang Li

Fall 2025

COE COLLEGE

Hamming codes can be done in the EE way

- Before that, we need to acquire some basic skillsets.

Pre-step: Data forms

Step 1: Data manipulation

Step 2: Information storage

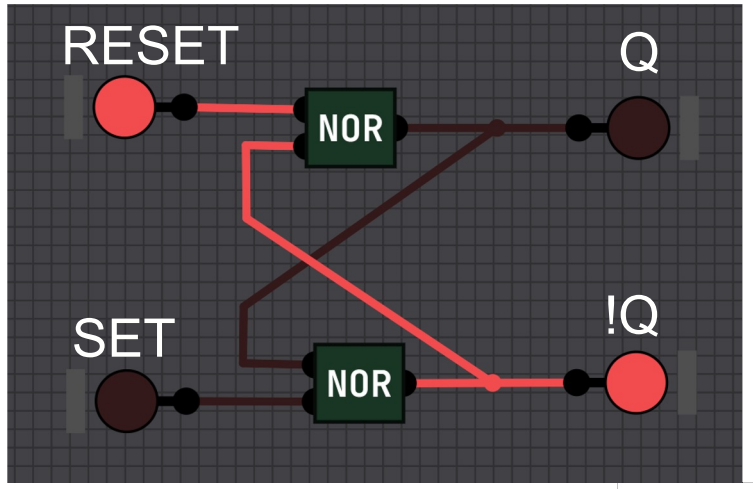
Step 3: Interface

3.1 Information flow

~~3.2 Physical contacts~~ (better stuff to talk about in PCB designs)

Understanding SR latch with truth table and timing diagram

SR latch

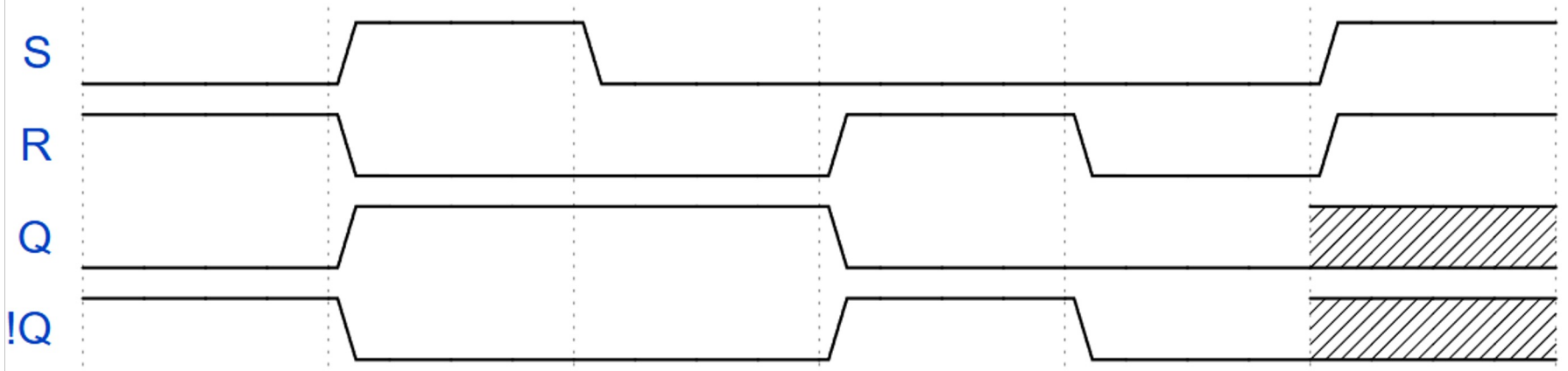


SR latch truth table

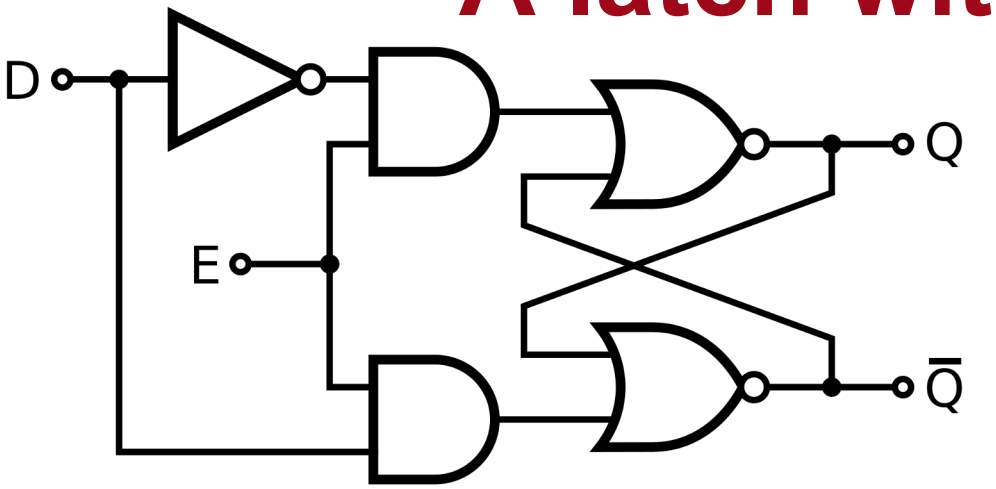
S	R	Output (Q)
0	0	**Previous State**
0	1	**0**
1	0	**1**
1	1	**0 (Invalid)**

(HOLD)

SR Latch Timing Diagram (NOR Gates)

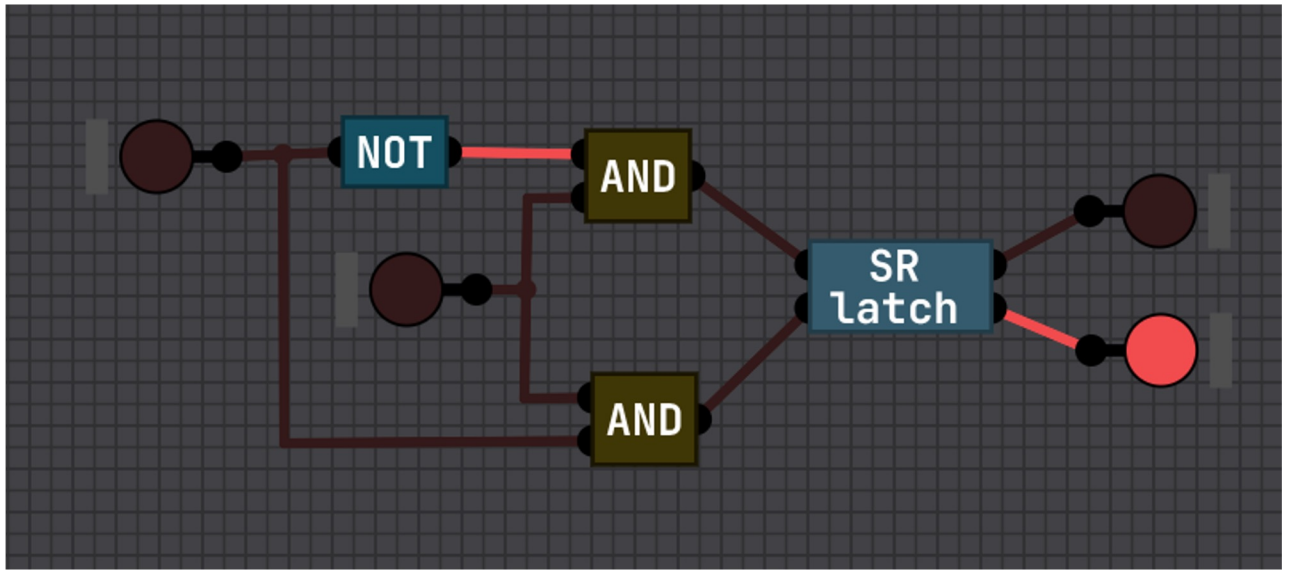


A latch with a “single” input



https://en.wikipedia.org/wiki/Flip-flop_(electronics)

A gated D latch



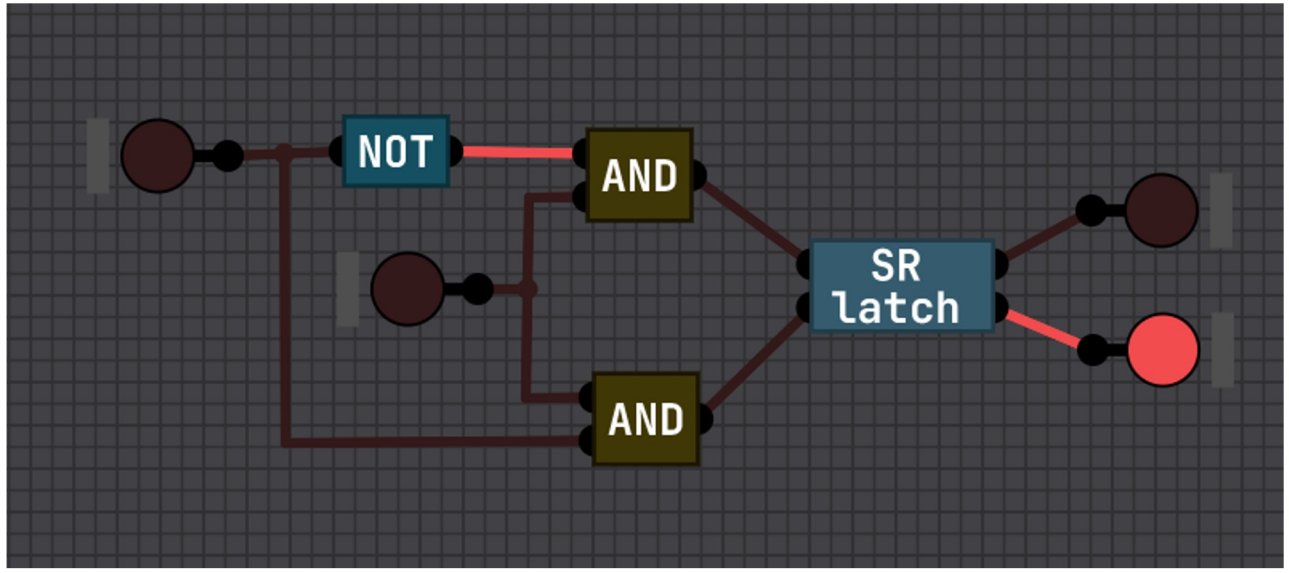
D latch truth table

D	E	Output (Q)
X	0	**Previous State**
0	1	**0**
1	1	**1**

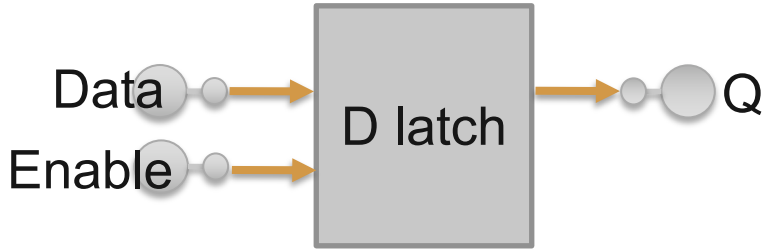
COE COLLEGE

A latch with a “single” input

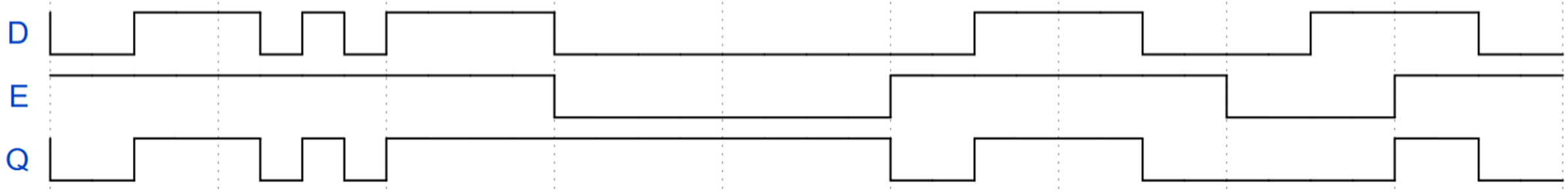
A gated D latch



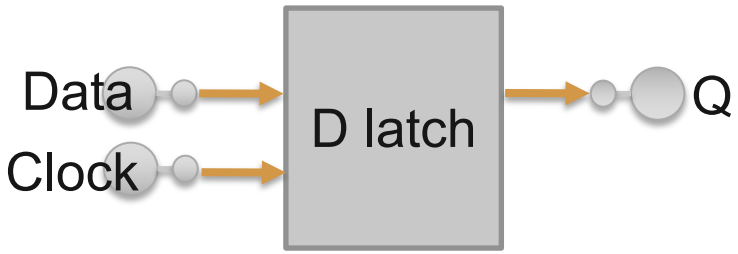
Also known as: D latch/data latch

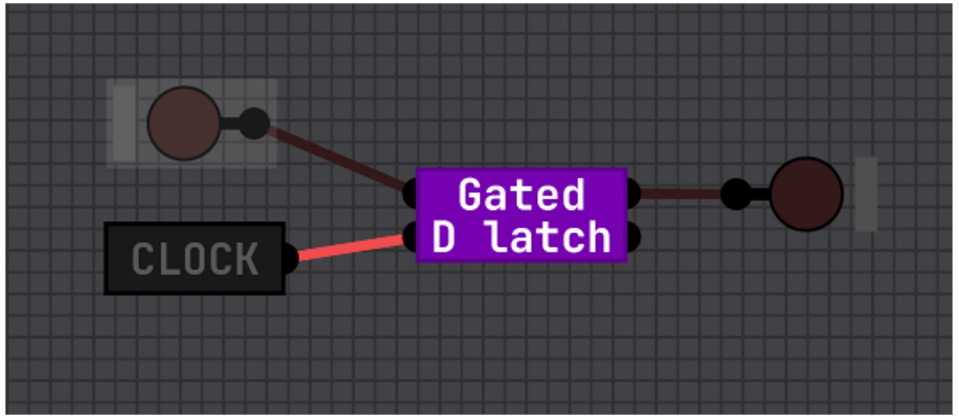


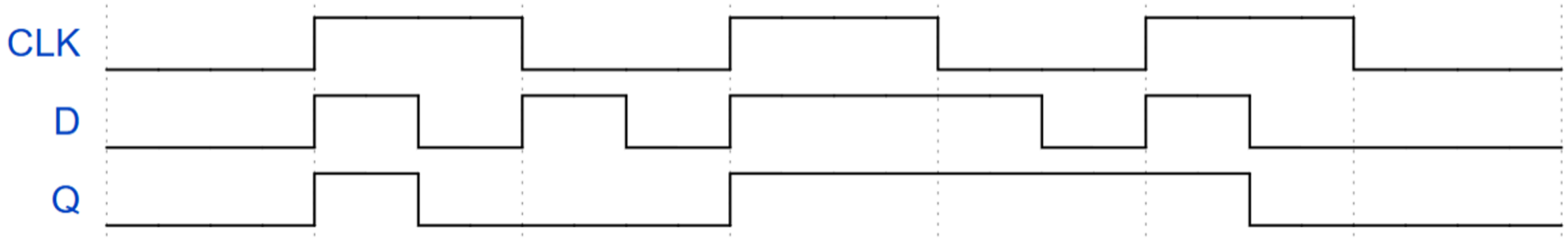
Gated D latch



A clocked D latch

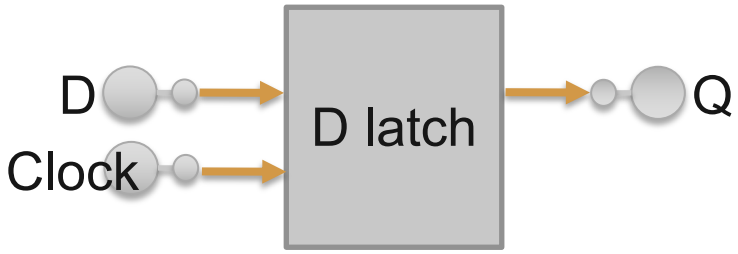




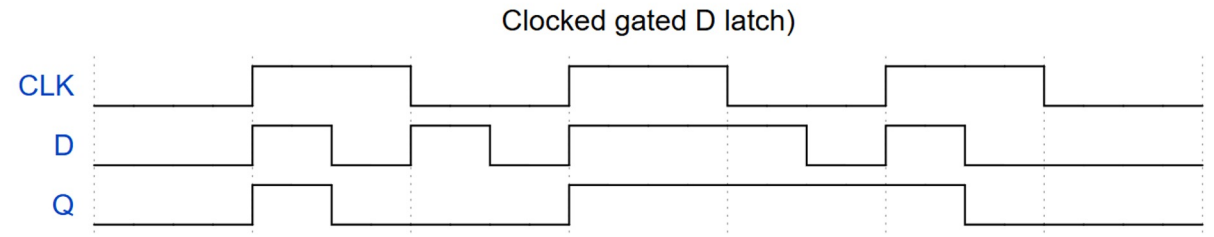


COE COLLEGE

Timing diagram code:

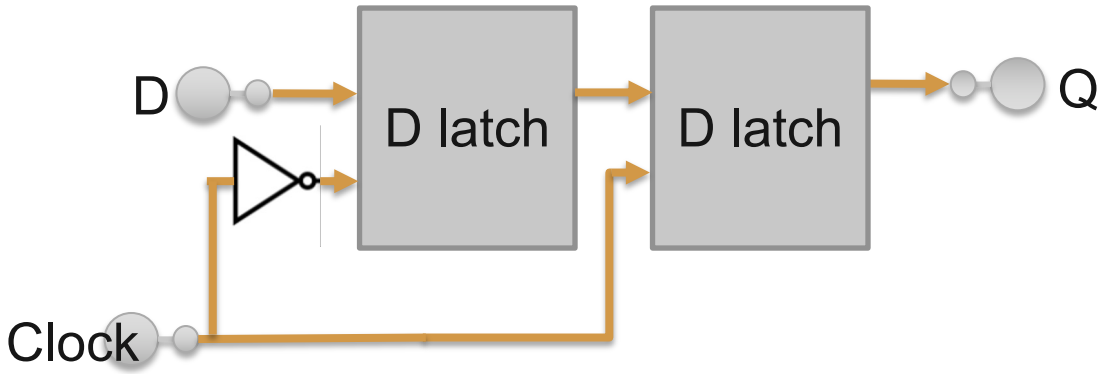



```
{
  "signal": [
    {"name": "CLK",        "wave": "lhlhlhl"},
    {"name": "D",        "wave": "lpphppl"},
    {"name": "Q",        "wave": "lplh.pl"},

  ],
  "head": {
    "text": "Clocked gated D latch)"
  },

  "config": {
    "hscale": 2
  }
}
```



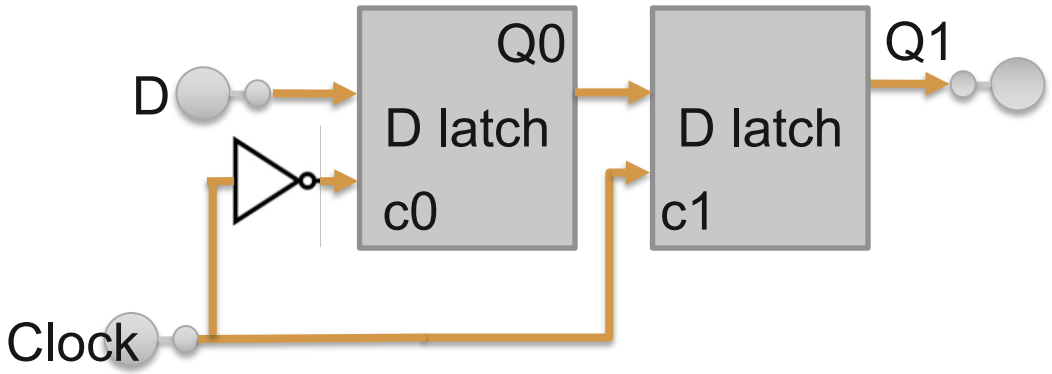
COE COLLEGE

Rising edge-triggered flip flops: synchronizing (timed trigger) achieved

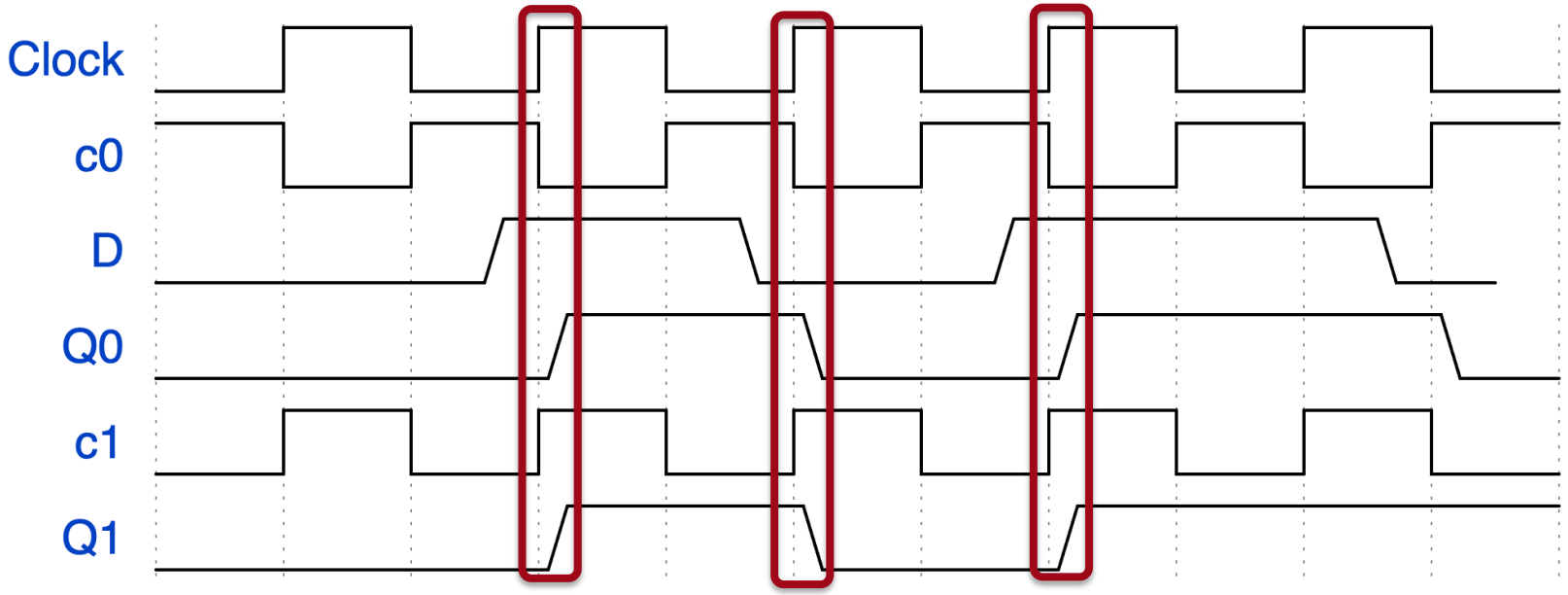


Also known as:
Master-slave D type flip flop

Rising edge-triggered flip flops: synchronizing (timed trigger) achieved

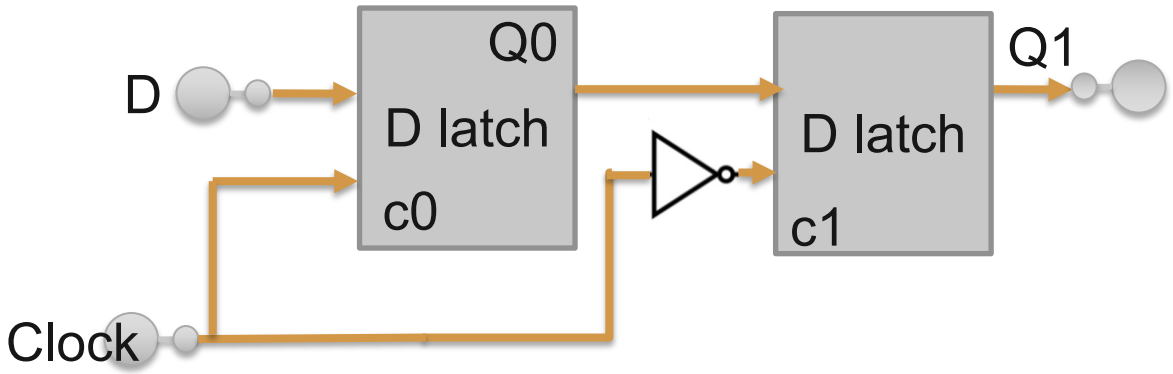


Also known as:
Master-slave D type flip flop



COLLEGE

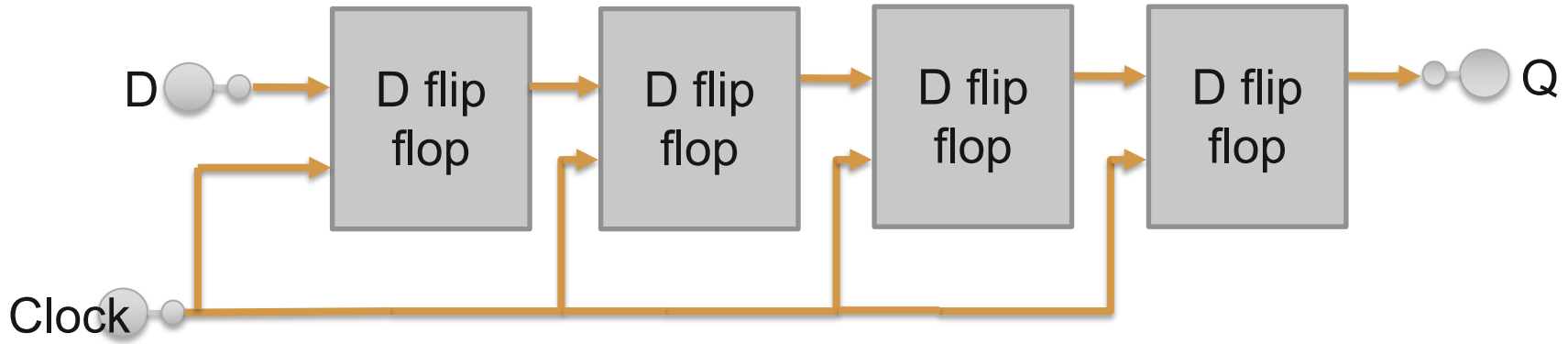
Falling edge-triggered flip flops: synchronizing (timed trigger) achieved



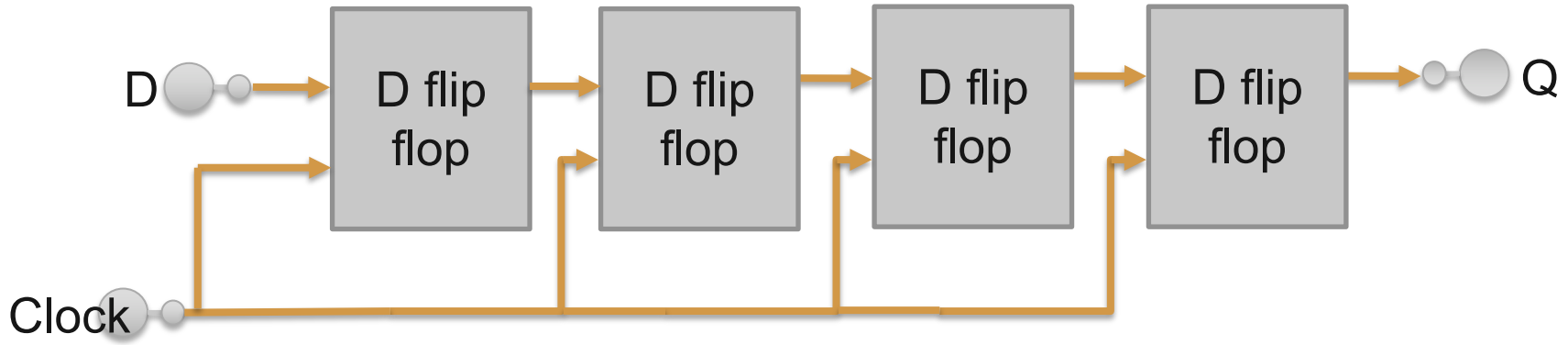
Also known as:
Master-slave D type flip flop

Timing diagram is in HW#7.

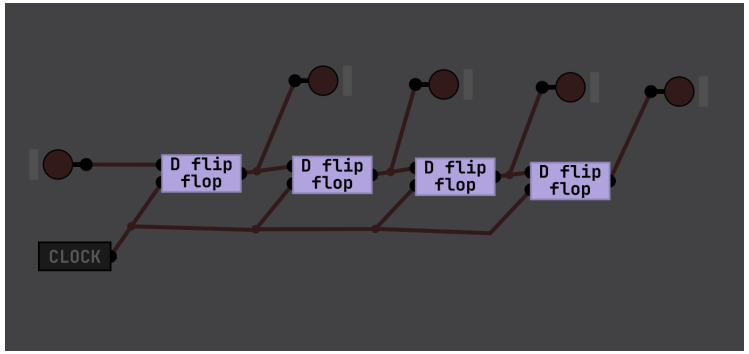
One more application with D flip-flops



So, this is a (4 bit) shift register



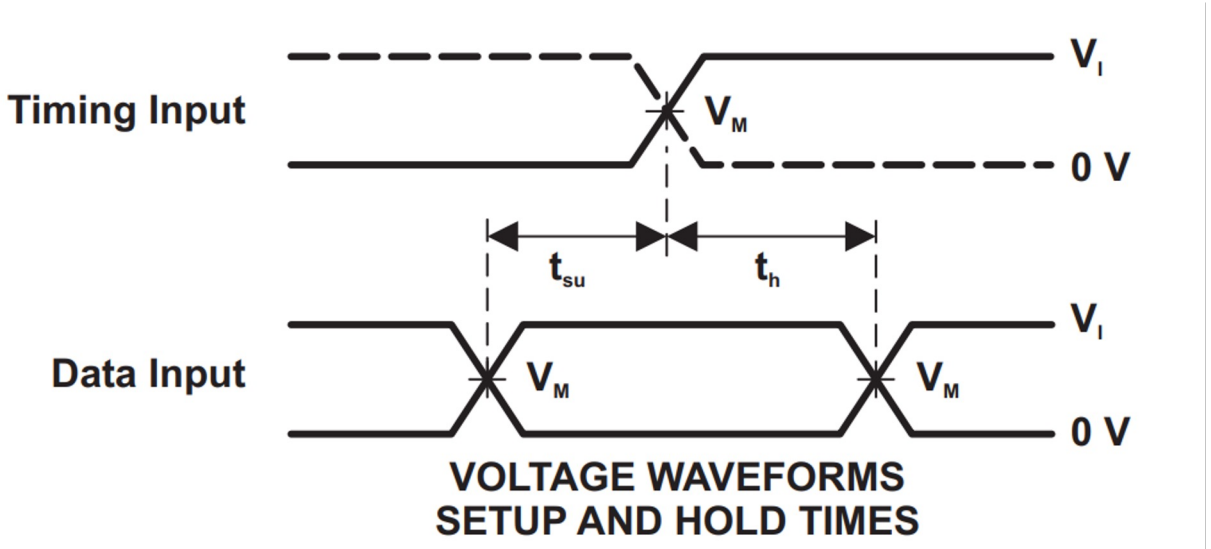
- Serial In (SI), Serial Out (SO)
- How about SIPO, PISO, and PIPO?



COE COLLEGE

Dynamic discipline: handling the interface between logics and time

- **A clock signal (with its edges if it's a flip-flop) to define transitions.**
- **Stable inputs during that transition window. (So, some *setup time* t_{su} and some *hold time* t_h)**
- **Thus, guaranteed viable output other than its own switching delays.**



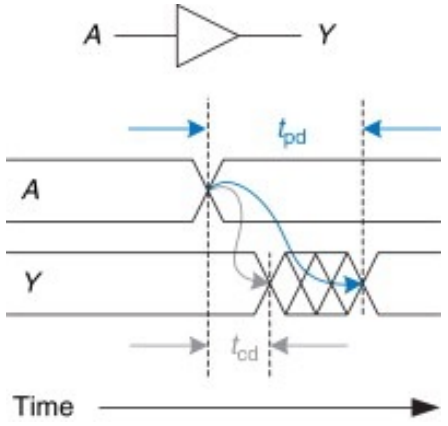
VOLTAGE WAVEFORMS
SETUP AND HOLD TIMES

COE COLLEGE

Nolan, Stephen M., Jose M. Soltero, and Shreyas Rao. "Understanding and Interpreting Standard-Logic Data Sheets." Application Report, Texas Instruments, 2002

Dynamic discipline: handling the interface between logics and time

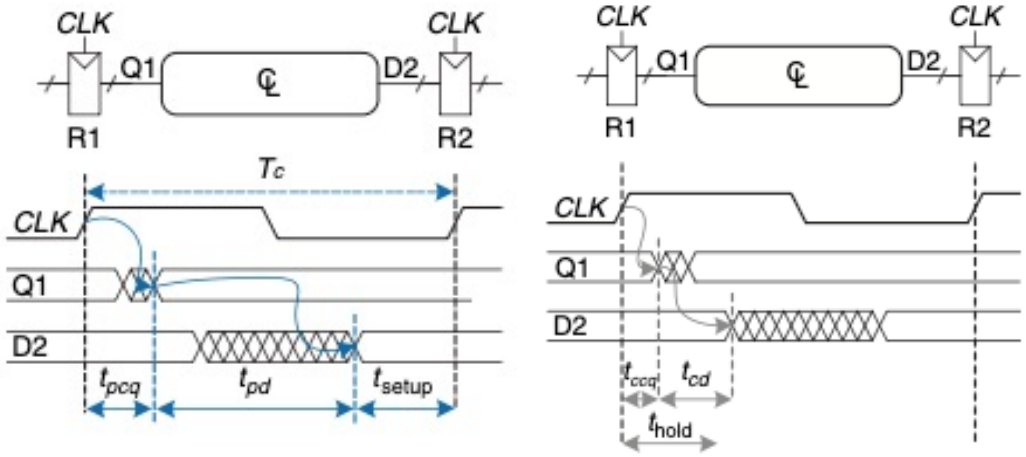
- Unlike propagation delay (t_{pd}) and contamination delay (t_{cd}) , the setup time t_{su} and hold time t_h are intentional.



- So, the final time constraint for sequential logics is:

  Minimum clock period: $T_c \geq \Sigma\, t_{pd} + t_{su}$

  Minimum delay constraint: $\Sigma\, t_{cd} \geq t_h$



COE COLLEGE

Haris, D., and S. Haris. "Digital Design and Computer Architecture ARM Edition." (2016).

Dynamic discipline:

Timing Requirements (Over Recommended Operating Free-air Temperature Range (unless otherwise noted)) (see Figure 2)

			SN54LVTH16646				SN74LVTH16646				UNIT
			V_{CC} = 3.3 V ± 0.3 V		V_{CC} = 2.7 V		V_{CC} = 3.3 V ± 0.3 V		V_{CC} = 2.7 V		
			MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX	
f_{clock}	Clock frequency			150		150		150		150	MHz
t_w	Pulse duration, CLK high or low		3.3		3.3		3.3		3.3		ns
t_{su}	Setup time, A or B before CLKAB↑ or CLKBA↑	Data high	1.2		1.5		1.2		1.5		ns
		Data low	2		2.8		2		2.8		
t_h	Hold time, A or B after CLKAB↑ or CLKBA↑	Data high	0.5		0		0.5		0		ns
		Data low	0.5		0.5		0.5		0.5		

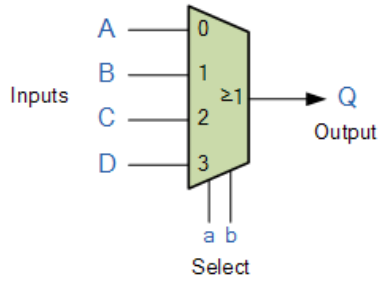
Figure 14. Example Timing-Requirements Section

COE COLLEGE

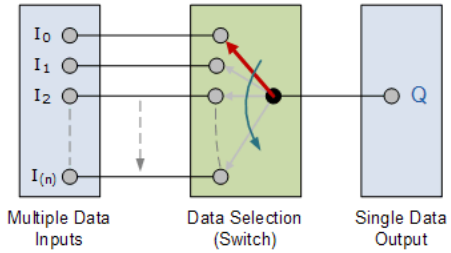
Nolan, Stephen M., Jose M. Soltero, and Shreyas Rao. "Understanding and Interpreting Standard-Logic Data Sheets." Application Report, Texas Instruments, 2002

Multiplexer (MUX):

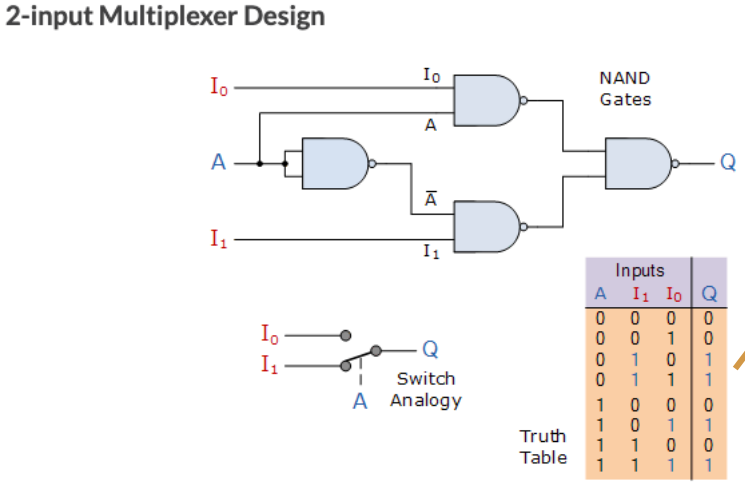
Typical symbol you'll see



Typical switching schematic you'll see



Typical logic schematic you'll see



Same truth table?

A	I_1	I_2	Q
0	0	X	0
0	1	X	1
1	X	0	0
1	X	1	1

Same truth table?

A	Q
0	I_1
1	I_2

COE COLLEGE

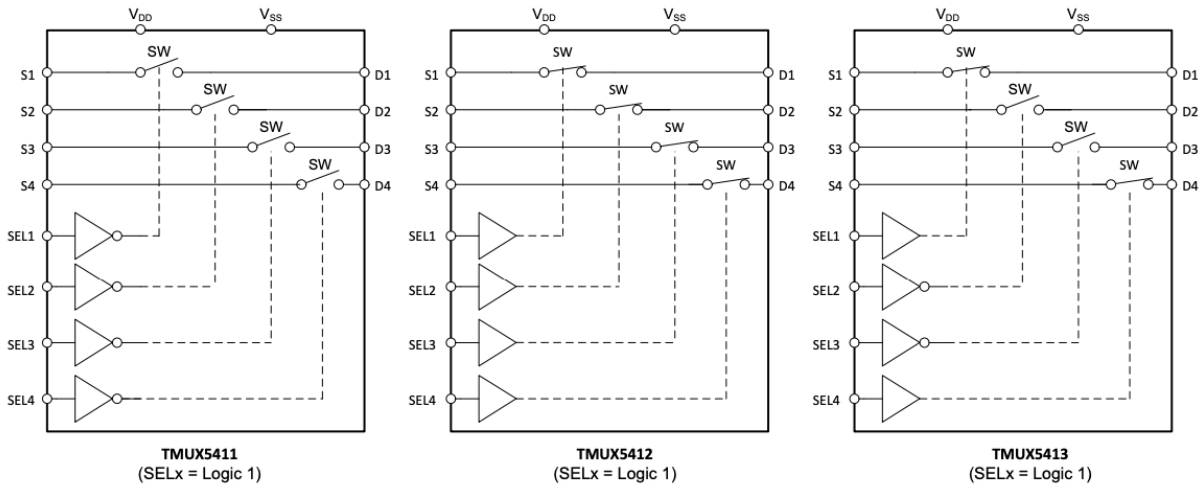
MUX diagrams from https://www.electronics-tutorials.ws/combination/comb_2.html

MUX IRL:

TMUX541x 50V, 21Ω, 1:1 (SPST) 4-Channel Switches with 1.8V Logic

- No logic schematics?
- When to use MUX?

7.2 Functional Block Diagram



TMUX5411 (SELx = Logic 1)

TMUX5412 (SELx = Logic 1)

TMUX5413 (SELx = Logic 1)

7.4.1 Truth Tables

TMUX5412 Truth Table provides the truth table for TMUX541x.

Table 7-1. TMUX5411 Truth Table

SEL x (1)	CHANNEL x
0	Channel x ON
1	Channel x OFF

Table 7-2. TMUX5412 Truth Table

SEL x (1)	CHANNEL x
0	Channel x OFF
1	Channel x ON

Table 7-3. TMUX5413 Truth Table

SEL1	SEL2	SEL3	SEL4	ON / OFF CHANNELS(1)
0	X	X	X	CHANNEL 1 OFF
1	X	X	X	CHANNEL 1 ON
X	0	X	X	CHANNEL 2 ON
X	1	X	X	CHANNEL 2 OFF
X	X	0	X	CHANNEL 3 ON
X	X	1	X	CHANNEL 3 OFF
X	X	X	0	CHANNEL 4 OFF
X	X	X	1	CHANNEL 4 ON

(1) x denotes 1, 2, 3, or 4 for the corresponding channel.

The last piece of interface: the flow of data

The plan is to cover: I^2C, SPI, and UART

I²C: Inter-integrated circuit protocol

(Born in 1982)

A Basic Guide to I²C

TEXAS INSTRUMENTS

Joseph Wu

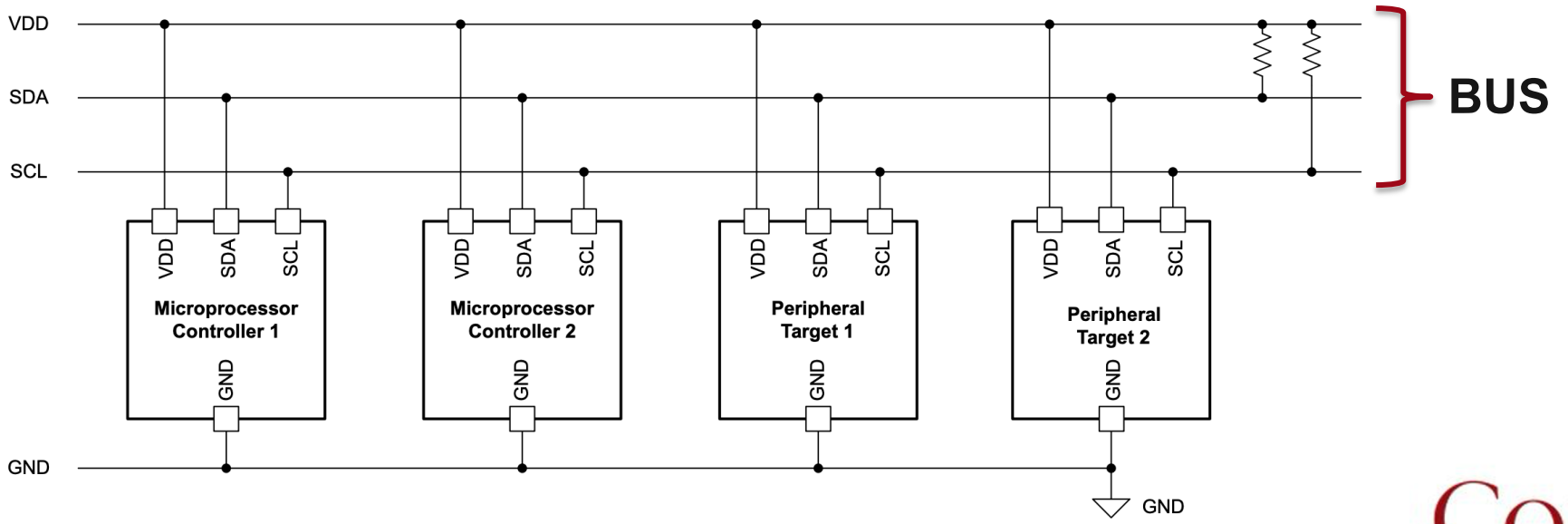


Figure 2-1. Typical I²C Implementation

COLLEGE

I²C: how signaling works (open-drain)

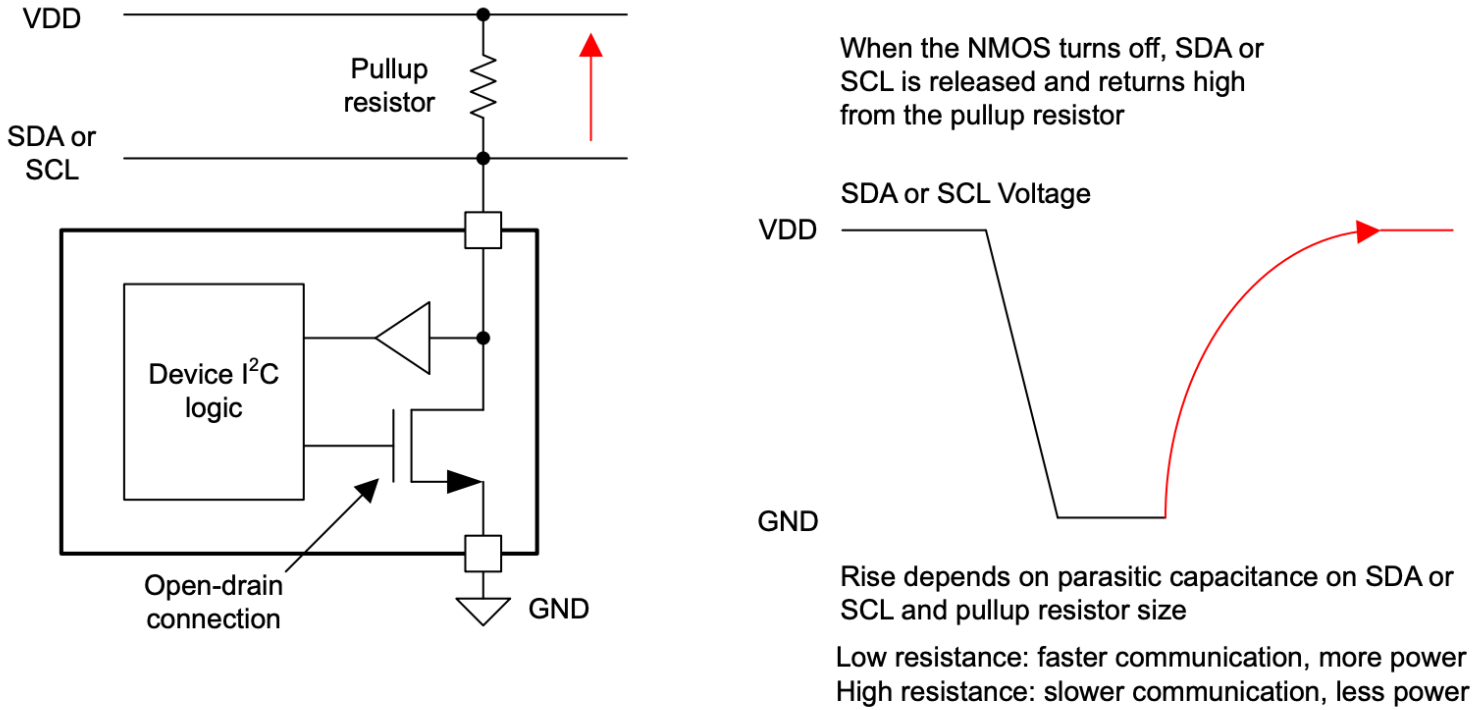
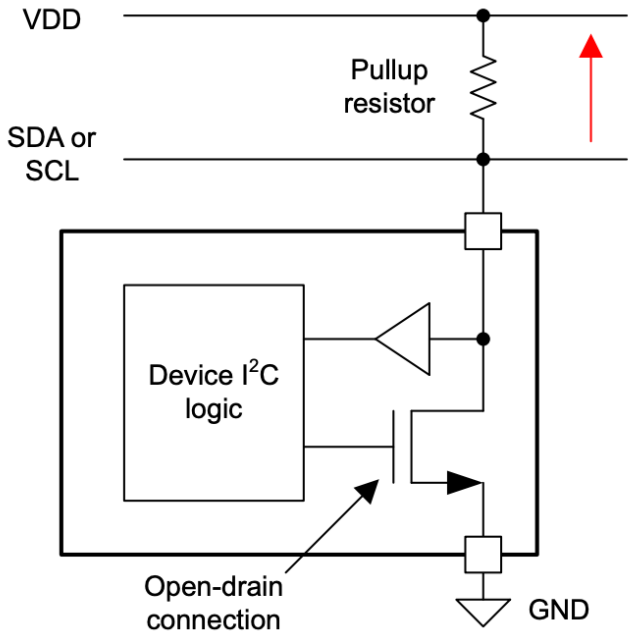
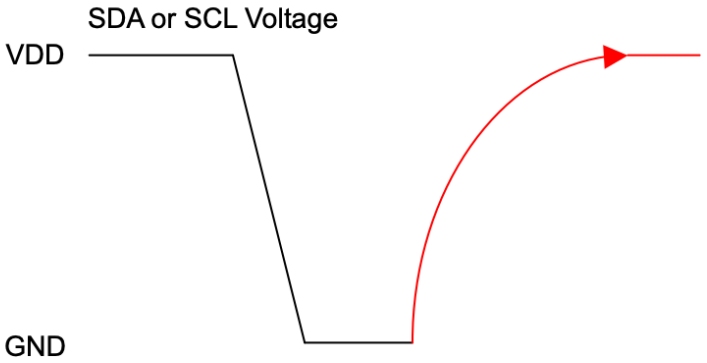


Figure 2-3. Pullup Resistor Pulls Line High When NMOS is Turned Off

I²C: how signaling works



When the NMOS turns off, SDA or SCL is released and returns high from the pullup resistor



Rise depends on parasitic capacitance on SDA or SCL and pullup resistor size

Low resistance: faster communication, more power
High resistance: slower communication, less power

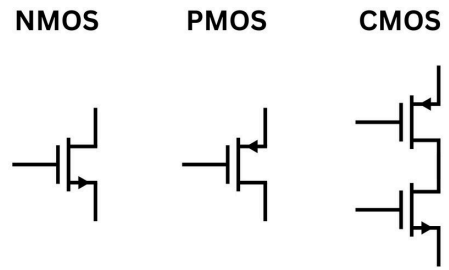


Figure 2-3. Pullup Resistor Pulls Line High When NMOS is Turned Off

Potential issue for multiple I²C on the line?

I²C: data frames

How to start and stop:

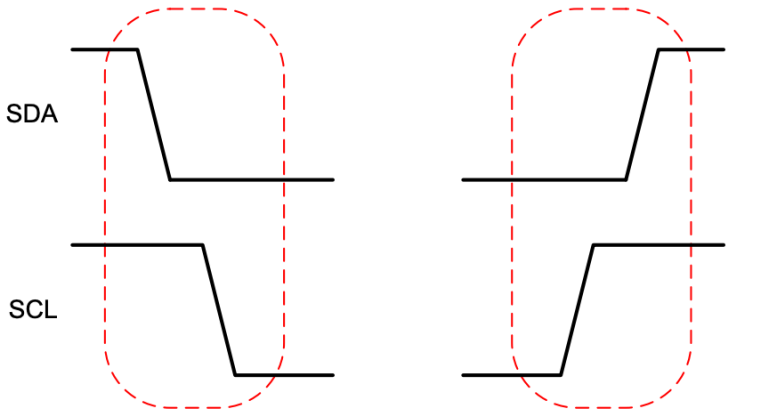


Figure 3-1. I²C START and STOP

How to tell high and low:

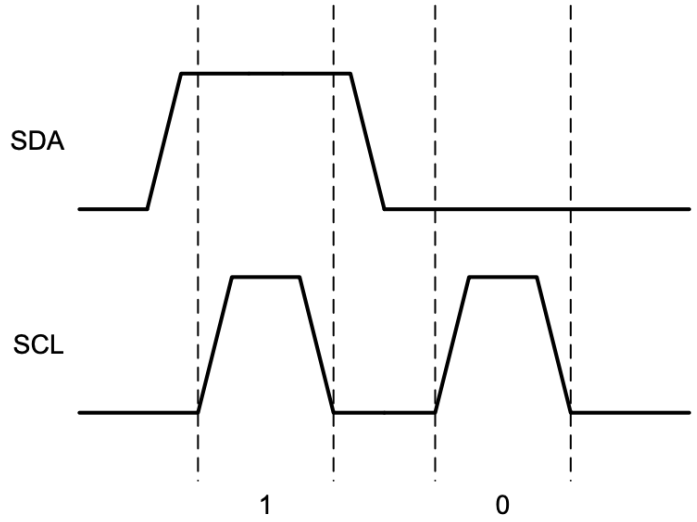


Figure 3-2. I²C Digital One and Zero Representations

I²C: data frames

How to pick who to read and write:

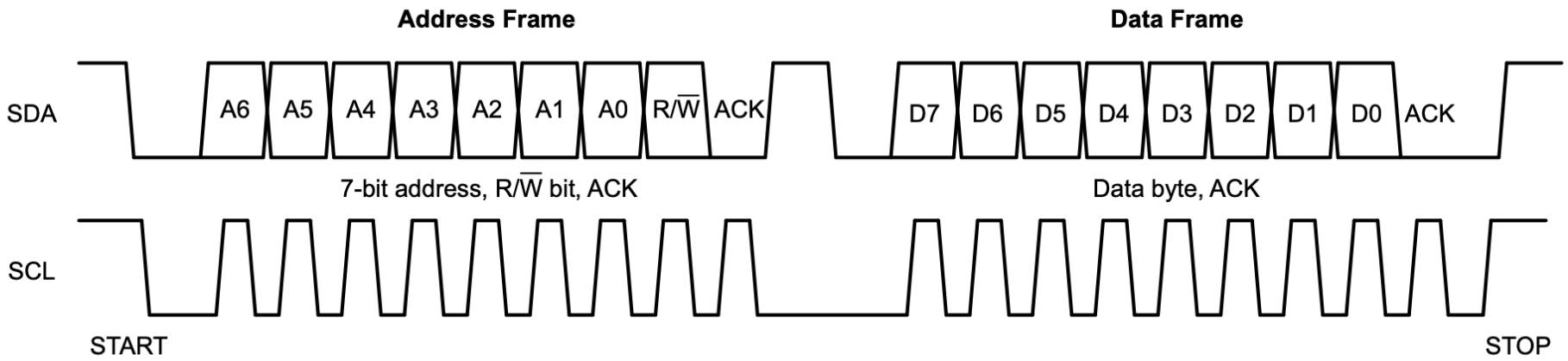
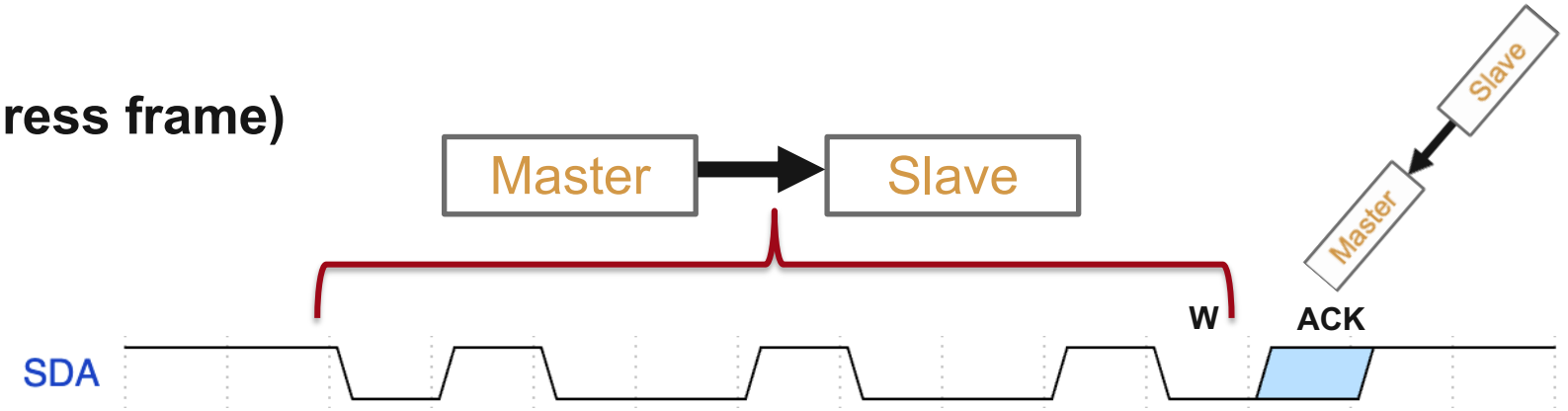


Figure 3-3. I²C Address and Data Frames

- 7-bit address, but some address are reserved.
- R (read):1
  W (write): 0
- ACK (acknowledge): 0
  NACK (no ACK): 1

I²C in works:

Step 1: who to talk (address frame)



Step 2: what to talk (data frame)