Alexandria University
Faculty of Engineering
Electrical Engineering Department

Lab 4: Design of a CMOS D-Flip-Flop and CMOS NAND based S-R Latch

Objective:

Explains the concepts in creating the memory units called a D-Flip-Flop and SR-Latch. It explains how to use a Transmission gate and a few inverters to store one bit of data.

D-Flip-Flop

Introduction:

One type of memory that is most used in today's technology is the D-Flip Flop. It is used registers to store data. There is different ways to construct D-Flip Flop, such as using NAND gates or the method we used with Transmission gates (TG) and inverters. Figure 1 shows the connections of two TG with inverters.

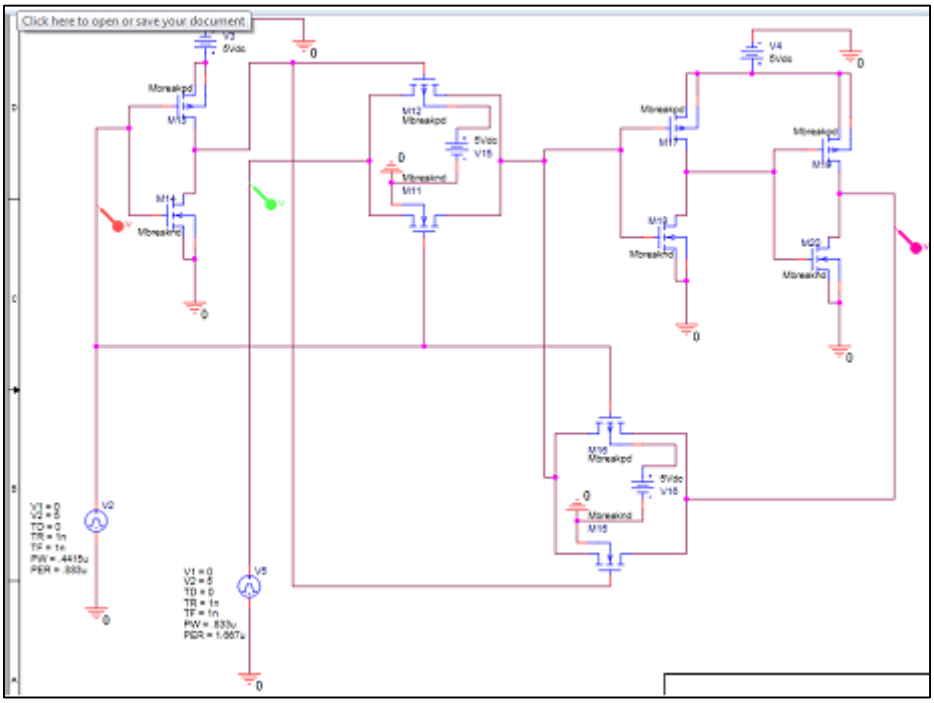

Figure 1 D-Flip-flop

$\overline{\text{SET}}$	$\overline{\text{RESET}}$	D	CK	Q	$\overline{\text{Q}}$
0	1	-	-	1	0
1	0	-	-	0	1
0	0	-	-	1	0
1	1	1	↑	1	0
1	1	0	↑	0	1

Table 1 D-Flip-Flop Truth Table

The D-Flip-Flop is one of the easiest sequential memory types to implement but is also one of the most powerful. Table 1 shows how the D Flip Flop works when the clock is raising then the data in the D input will be outputted to the Q output. This feature is allows D flip flops to be connected easily together to make shift registers.

Requirements:

You are required to design D-Flip-Flop like the one on figure 1. It consists of TGs and inverters. It has two input signals: clock signal and data signal.
Then you are required to test this circuit to make sure that it follows the truth table described in table 1.

Procedure:

- Build the circuit shown in figure 1.
- Apply input signal and clock with different frequency.
- D Flip Flop theory of operation: Q copies D when there is a rising clock, When the clock isn't rising though it is also doing what is supposed to do by "Holding" the same value.
- Increase the frequency and prove that the clock should be at least twice as fast as the D input (for the same input frequency).

SR-Latch

Introduction:

One for the basic dynamic logic memory cell is the Set/Reset Latch or simply SR. The SR has 3 combinations as acceptable inputs. These are illustrated in Table 2. The Last stage is undefined because two opposite states conditions are trying to be satisfied at the same time.
Figure demonstrates the NAND gate implementation of the SR-latch which was used for the purposes of the lab.

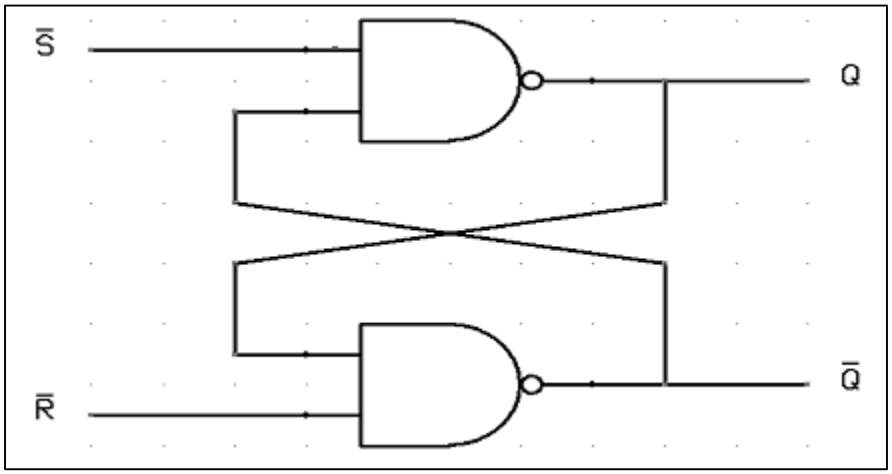

Figure 2 NAND based SR-Latch

Requirements:

You are required to design SR-Latch like the one on figure 3. It consists of 2 back to back NAND gates. It has two input signals: set and reset.
Then you are required to test this circuit to make sure that it follows the truth table described in table 2.

Procedure:

- Build the circuit shown in figure 3. Do not forget to size the NAND gate to get delay like the reference inverter.
- Apply two inputs with one with frequency f and the other f/2 to S and R inputs then plot Q and Q_bar and check the operation of SR-Latch.

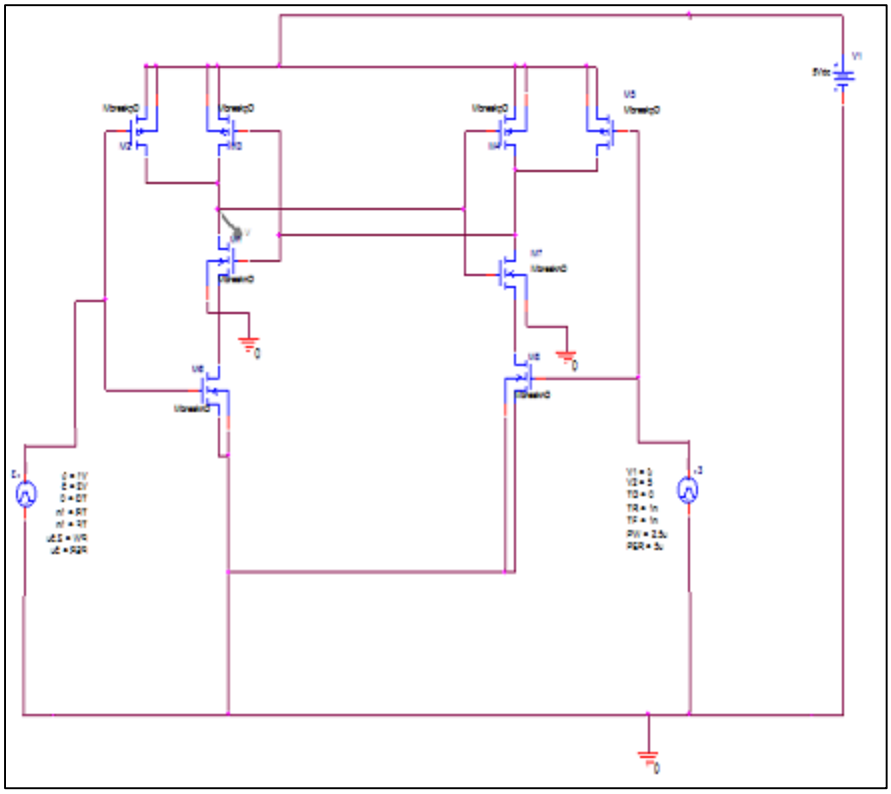

Figure 3 NAND based SR-Latch

Table 2 Truth Table for SR-Latch

Q	S	R	Q(t+1)
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	Undetermined
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	Undetermined

Appendix:

PMOS Model

.model Mbreakp PMOS LEVEL = 3

+ TOX = 200E-10 NSUB = 1E17 GAMMA = 0.6

+ PHI = 0.7 VTO = -0.9 DELTA = 0.1

+ UO = 250 ETA = 0 THETA = 0.1

+ KP = 40E-6 VMAX = 5E4 KAPPA = 1

+ RSH = 0 NFS = 1E12 TPG = -1

+ XJ = 500E-9 LD = 100E-9

12

+ CGDO = 200E-12 CGSO = 200E-12 CGBO = 1E-10

+ CJ = 400E-6 PB = 1 MJ = 0.5

+ CJSW = 300E-12 MJSW = 0.5

==========

NMOS Model

.model Mbreakn NMOS LEVEL = 3

+ TOX = 200E-10 NSUB = 1E17 GAMMA = 0.5

+ PHI = 0.7 VTO = 0.8 DELTA = 3.0

+ UO = 650 ETA = 3.0E-6 THETA = 0.1

+ KP = 120E-6 VMAX = 1E5 KAPPA = 0.3

+ RSH = 0 NFS = 1E12 TPG = 1

+ XJ = 500E-9 LD = 100E-9

+ CGDO = 200E-12 CGSO = 200E-12 CGBO = 1E-10

+ CJ = 400E-6 PB = 1 MJ = 0.5

0

+ CJSW = 300E-12 MJSW = 0.5