EECS 42 Introduction to Electronics for Computer Science

Andrew R. Neureuther

Lecture # 20 Logic Transients

Handout of Monday Lecture.

A) 2nd Midterm Review (Cont.)
B) Latch circuit to hold/release signals
C) Cascade CMOS elements with latches
D) Logic Feedback creates memory

http://inst.EECS.Berkeley.EDU/~ee42/

Game Plan 04/14/03

Last Week: Logic Delay; resistor model, CMOS operation and delay

Monday 4/14/03:

- **2nd Midterm review (Cont.)**
- **CMOS Latch and use in logic cascade**
- **Feedback in logic to produce memory**

Wednesday 04/09/03:

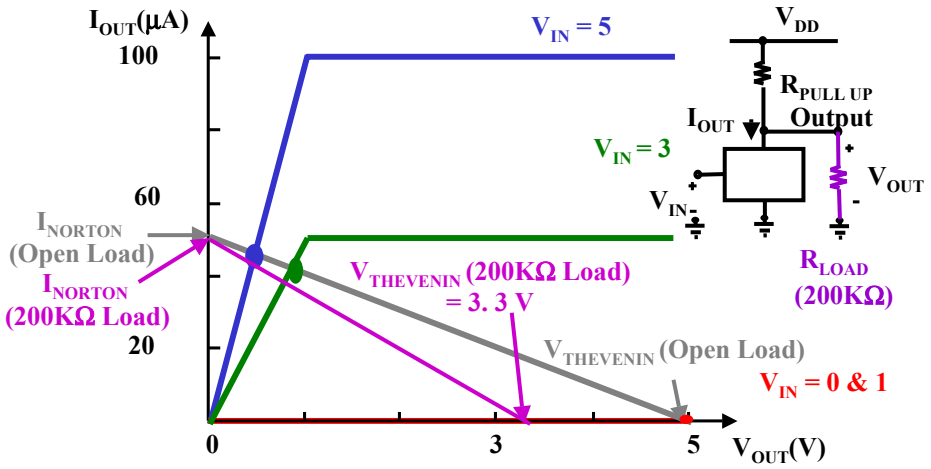
- **2nd Midterm**

Next (13th) Week: Diodes and MOS Operation

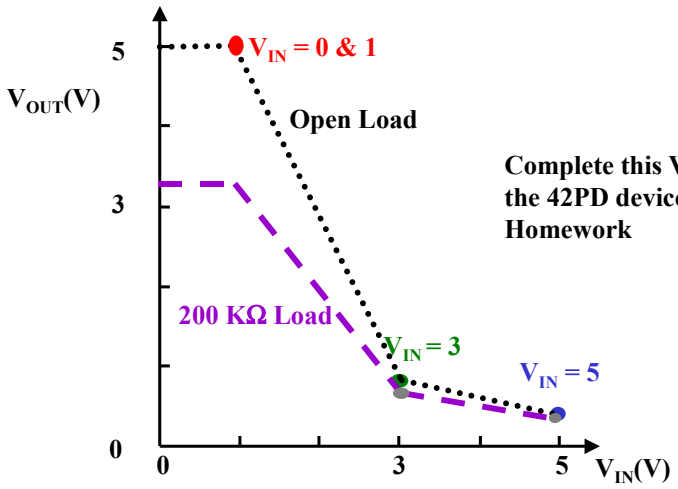
No Problem set for 4/16 as Midterm 4/16: Lectures 1-17 with emphasis on Lectures 10-17; Review Session Monday 5:30-7PM

Problem set #10 for 4/23: Logic Delay

Composite Current Plot for the 42S_NMOS Circuit with 200kΩ Load to Ground



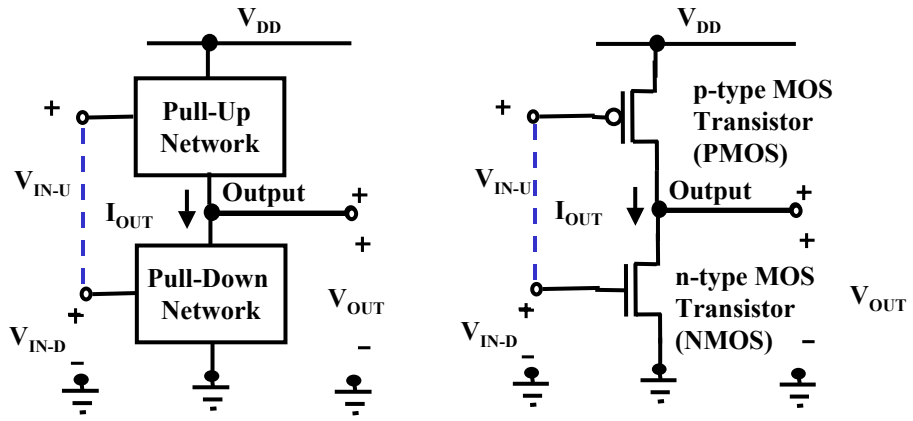
Voltage Transfer Function for the 42S_NMOS Logic Circuit w/wo Load



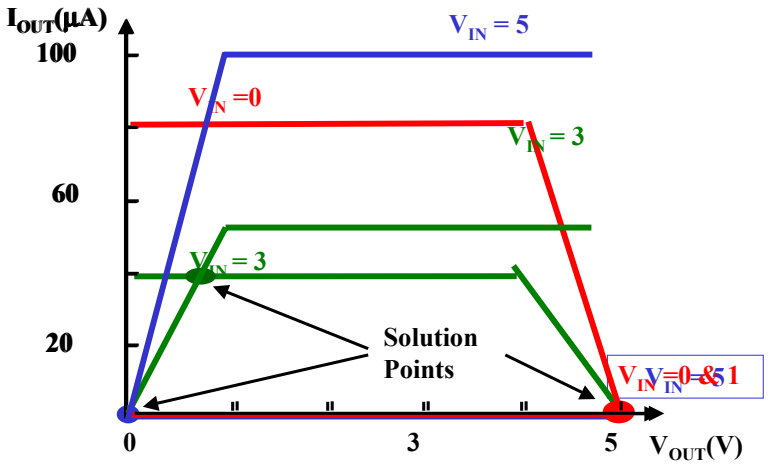
Complete this VTC for the 42PD device in the Homework

Transistor Inverter Example

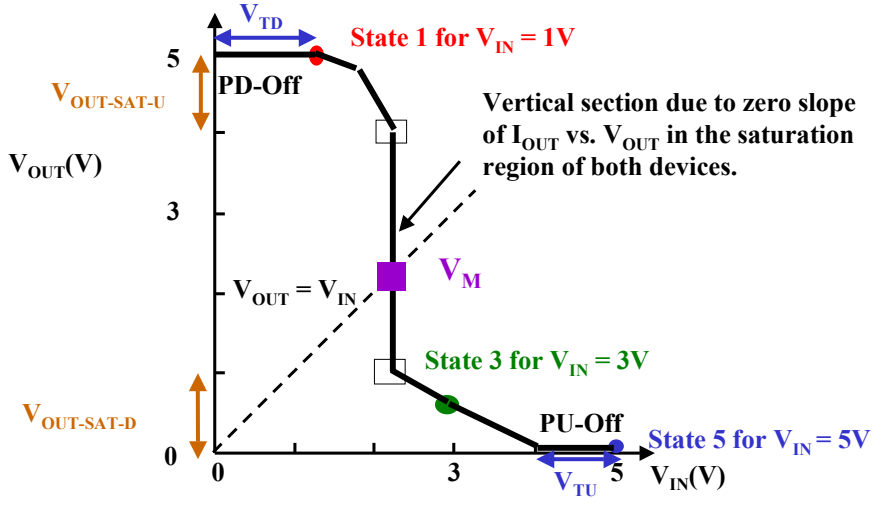
It may be simpler to just think of PMOS and NMOS transistors instead of a general 3 terminal pull-up or pull-down devices or networks.



Composite I_{OUT} vs. V_{OUT} to Find Points That Satisfies Both Devices for Each V_{IN}



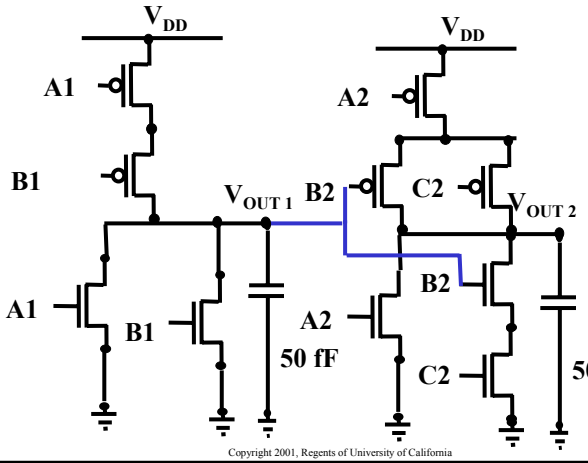
Voltage Transfer Function for the Complementary Logic Circuit



Vertical section due to zero slope of I_{OUT} vs. V_{OUT} in the saturation region of both devices.

Logic Gate Cascade

To avoid large resistance due to many gates in series, logic functions with 4 or more inputs are usually made from cascading two or more 2-4 input blocks.



$B2 = V_{OUT\ 1}$

The four independent input are A1, B1, A2 and C2.

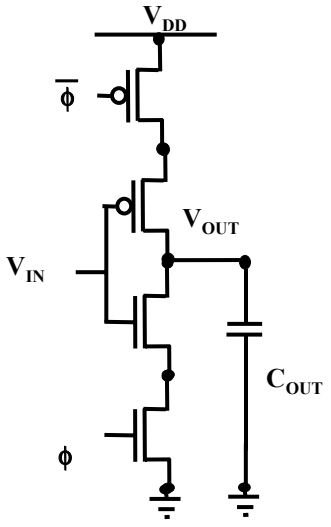
A2 high discharges gate 2 without even waiting for the output of gate 1.

C2 high and A2 low makes gate 2 wait for Gate 1 output

Data Synchronization problem

- Combinatorial logic gates can give incorrect answers prematurely and may take several gate propagation delays produce an answer.
- Clocks (signals as to when to proceed) and latches (which capture and hold the correct outputs) can provide synchronization.

Latch Controlled by a Clock

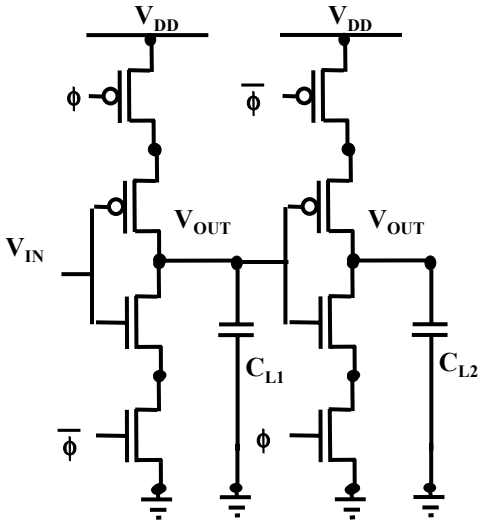


An inverter with clocked devices in series can form a latch.

When the clock ϕ is high its complement $\overline{\phi}$ is low and the inverter operates.

To synchronize the data the clock remains low until the data is correct at all locations on the chip. When the clock goes high the inverse of the data is passed.

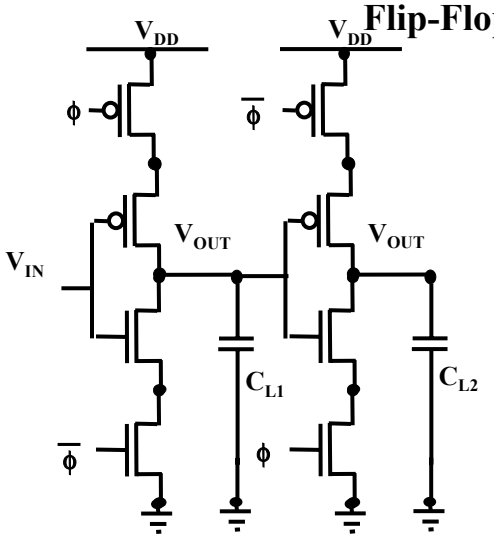
Latch Work Best In Pairs



The first stage operates while the clock is low and inverts and amplifies the arriving signal and charges or discharges C_{L1}.

The second stage operates while the clock is high and inverts the signal on C_{L1} to charge or discharge C_{L2} and downstream logic gate inputs.

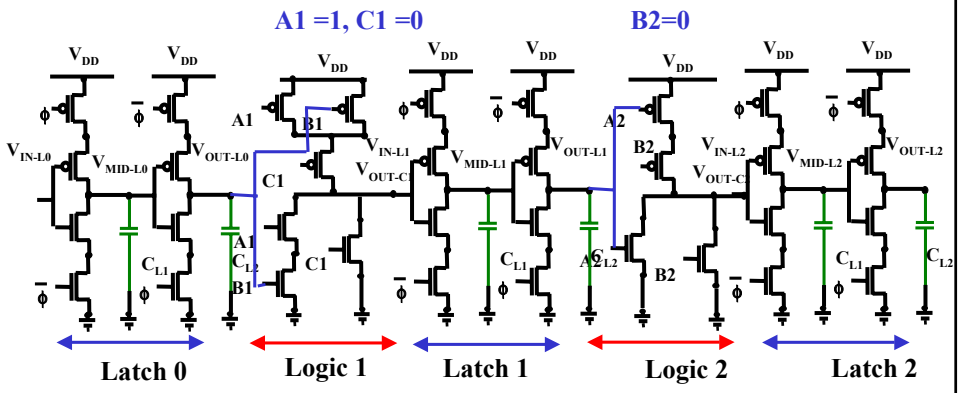
A Double Latch is an Edge-Triggered D Type Flip-Flop



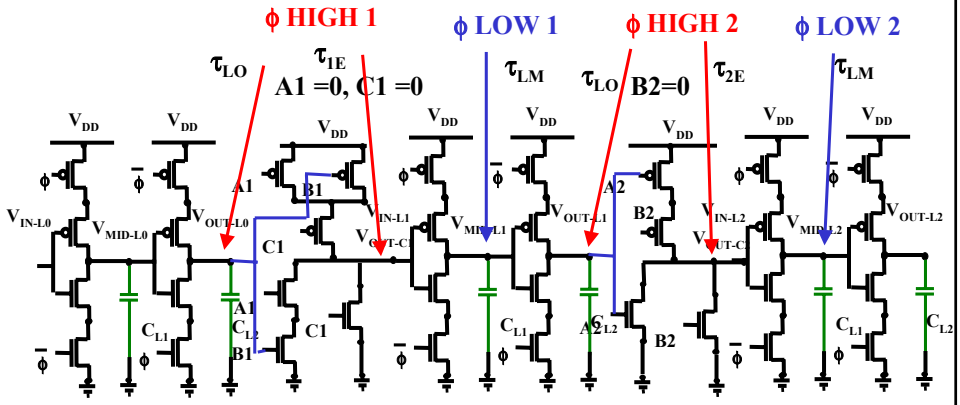
During the low part of the clock cycle this circuit records the input value and when the clock goes high drives $V_{OUT\,2}$ to the voltage level that arrived. (This is the classic function of a D flip-flop.)

Note that this circuit is not fooled by noise on the input and makes its decision on the rising edge of the clock (edge-triggered).

Combinatorial Logic and Clocked Latches: Wiring

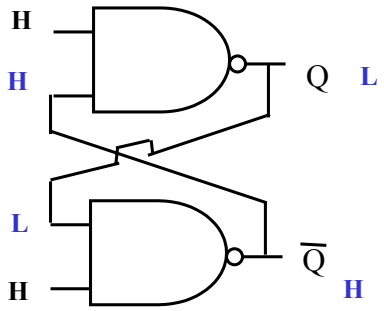


Combinatorial Logic and Clocked Latches: Signal Flow

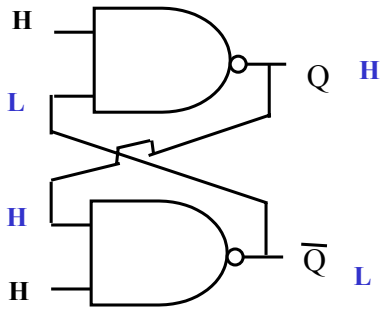


$\tau_{\phi-HIGH} > \tau_{LO} + Max(\tau_{1E}$ or $\tau_{2E})$ $\quad \tau_{\phi-LOW} > \tau_{LM}$ $\quad f_{CLOCK} = 1/(\tau_{\phi-HIGH} + \tau_{\phi-LOW})$

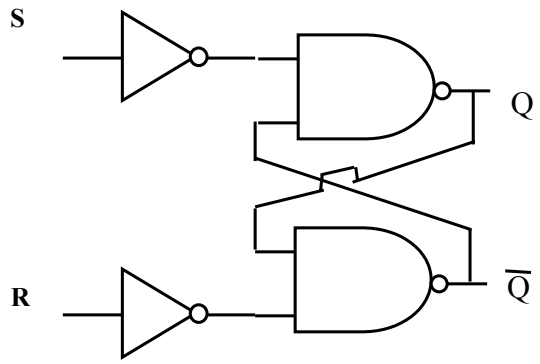
Feedback Can Provide Memory



Example of the Opposite State

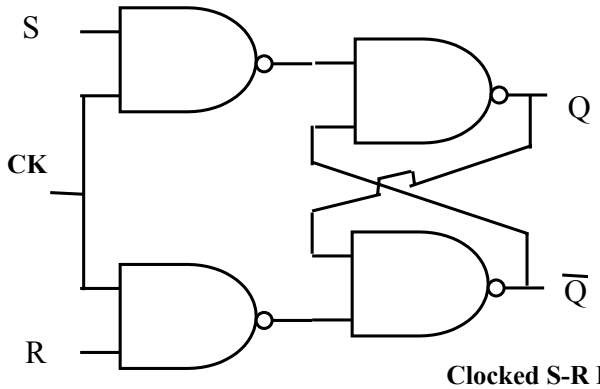


Adding Memory Controls



Set-Reset Flip-Flop

Adding a Clock



Clocked S-R Flip-Flop