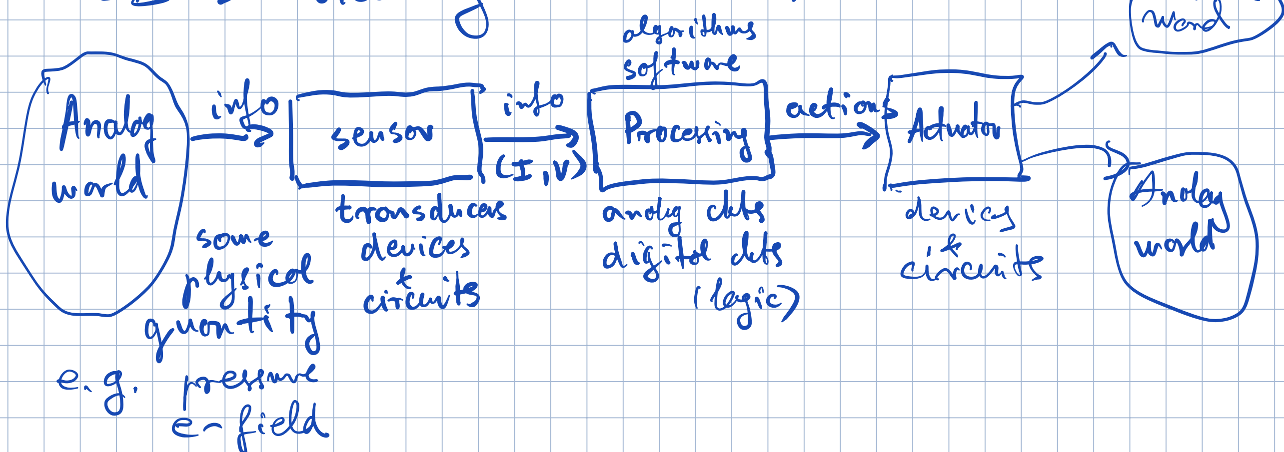
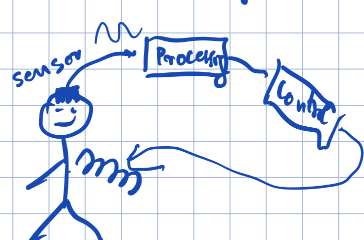
Lecture 1

* EECS 16B Intro
* Computing : Transistors & logic
  * Transistor models
    * logic
    * RC

EECS view of the world:



16B examples:

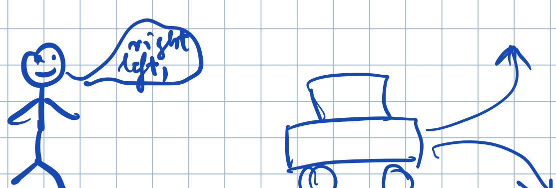


M1: Sensors : Interface circuits
- Diff. eqns.
- Phasors

M2: Control : Feedback

M3: Processing : Classification
- SVD / PCA

Lab



voice controlled robo-car

sensor : mic-board & filters

control : MC & motors

processing : SVD/PCA on MC

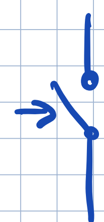
Processing How do we implement computation?

map numbers to distinct voltage levels

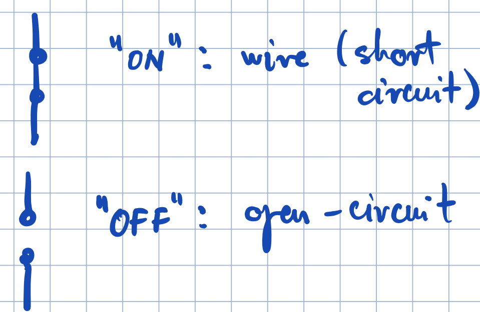
(e.g. binary) logic 0 → V_0, logic 1 → V_1

16A: Switch

Symbol



Model:

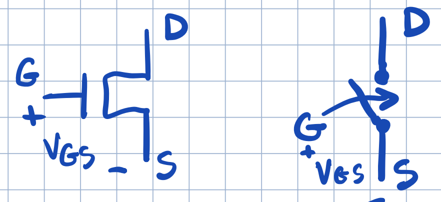


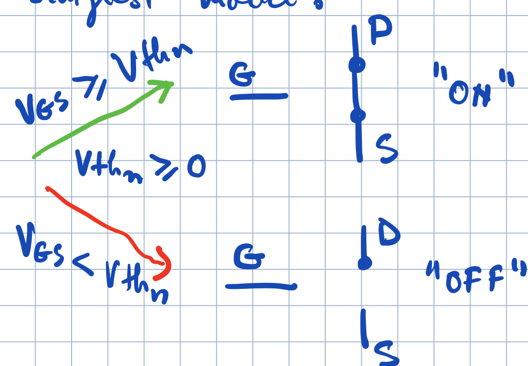
16B: Transistor

Sch. Symbol

Simplest model:

NMOS:



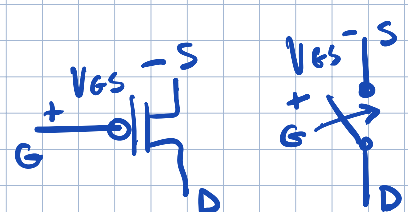


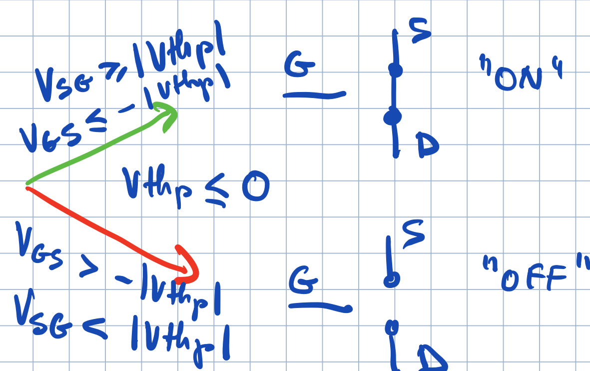
$V_{GS} = V_G - V_S$

$V_{SG} = V_S - V_G$

$V_{GS} = -V_{SG}$

PMOS:



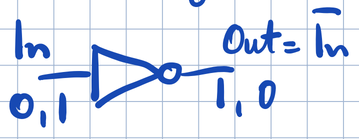


How do we use transistors to make digital logic?

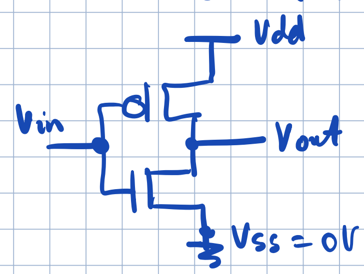
Simplest logic operation: NOT

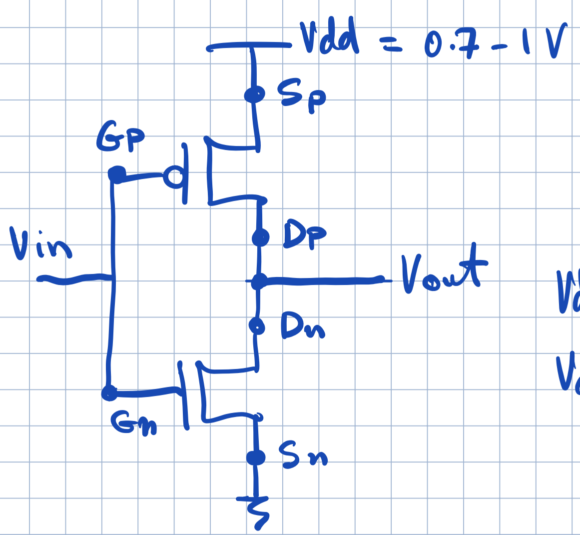
Simplest logic gate: inverter

"logic symbol"



CMOS (Complementary MOS)



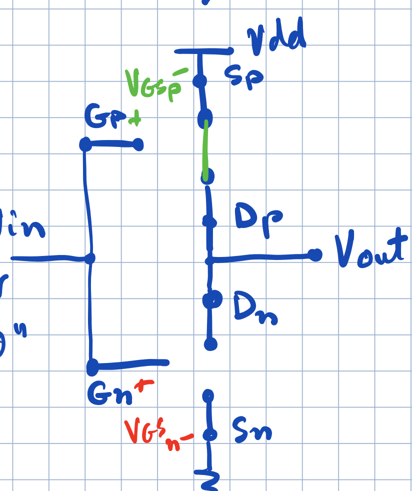


(16B assumption)

$V_{thn} + |V_{thp}| \geq V_{dd}$

$V_{dd} \geq V_{thn} \geq 0$

$V_{dd} \geq |V_{thp}| \geq 0$



$= V_{dd}$ (logic 1)

$V_{in} = 0V$
"logic 0"
(False)

$V_{GSp} = V_{Gp} - V_{Sp} = V_{in} - V_{dd} =$

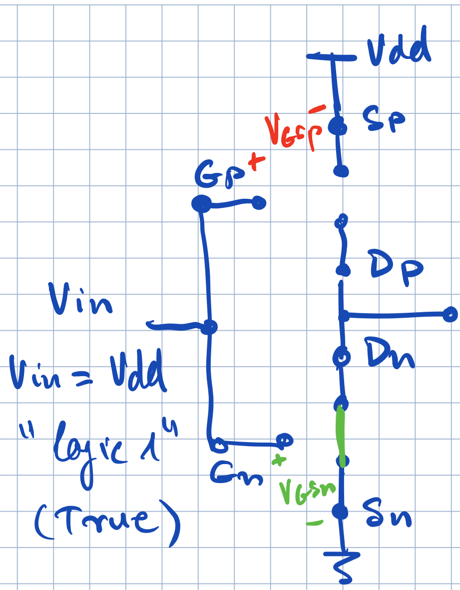
$= 0V - V_{dd} = -V_{dd} \leq -|V_{thp}|$

PMOS ON

$V_{GSn} = V_{Gn} - V_{Sn} = V_{in} - 0V =$

$= 0V - 0V = 0V < V_{thn}$

NMOS OFF



$V_{GSp} = V_{Gp} - V_{Sp} = V_{in} - V_{dd} = V_{dd} - V_{dd} =$

$= 0V > -|V_{thp}|$

"PMOS OFF"

$V_{out} = 0V$ (logic 0)

$V_{GSn} = V_{Gn} - V_{Sn} = V_{in} - 0V =$

$= V_{dd} - 0V = V_{dd} \geq V_{thn}$

NMOS ON

Summary of inverter operation : Truth table

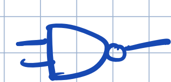
	Vin	Vout	
"logic 0"	0V	Vdd	"logic 1"
"logic 1"	Vdd	0V	"logic 0"

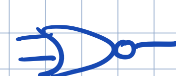
Boolean :

In	Out
0	1
1	0

$Out = \overline{In}$

Indeed the circuit performs a logic NOT in the voltage domain - provided digital inputs (0V and Vdd).

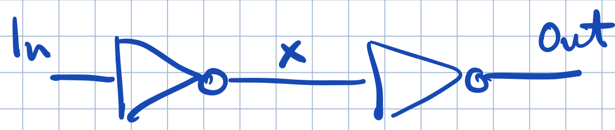
Other logic operations : NAND  $Out = \overline{A \cdot B}$

NOR  $Out = \overline{A + B}$

Can implement any logic function once you have these.

Let's make a processor: cascading logic

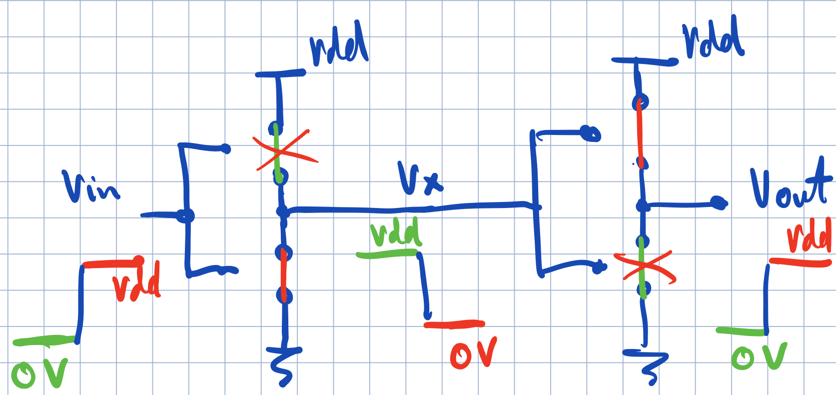
Simplest model:



$x = \overline{In}$

$Out = \overline{x}$

$Out = \overline{\overline{In}} = In$



State:

① $V_{in} = 0V \Rightarrow V_x = V_{dd}$
$\Rightarrow V_{out} = 0V$

② $V_{in} = V_{dd} \Rightarrow V_x = 0V \Rightarrow$
$\Rightarrow V_{out} = V_{dd}$

It looks like if the input changes suddenly, the output follows instantaneously!

Would make a super-fast processor & super-cool!

Not real! ☹

Model is good-enough for figuring out the logic function, but not for figuring out the speed (delay) & power.

Need to look at how a device is made and how it works to make a better model.