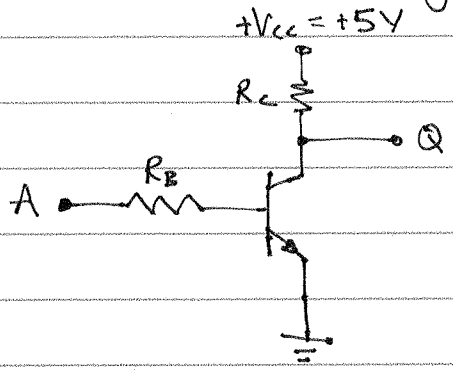
PHYS 220 Physical Electronics Fall 2010

Notes on Digital Logic Gates

- To this point we have been concerned with analog circuits, where the input and output voltages (and currents) can take on a continuum of values. Now we turn our attention to digital circuits, in which inputs/outputs take on only two values: 0 = low = false, and 1 = high = true.
- A logic inverter using a transitor (BJT):



If A is "low" (~ 0V) then transistor is cutoff and $I_c = 0$
→ Q is "high" (~ +5V)

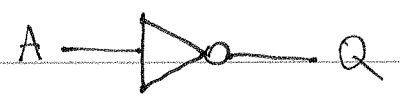
If A is "high" (~5V) then transistor is saturated and $I_c \simeq \frac{+V_{cc}}{R_c}$
→ Q is "low" (~ 0V)

Truth table for transistor inverter

A	Q
0	1
1	0

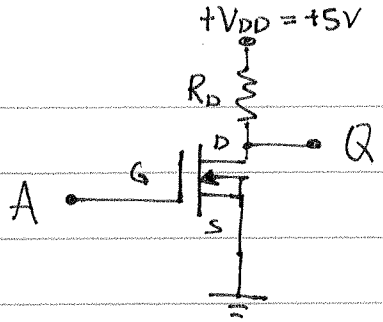
Symbolically $Q = \overline{A}$ "Q equals NOT A"

Logic gate schematic symbol

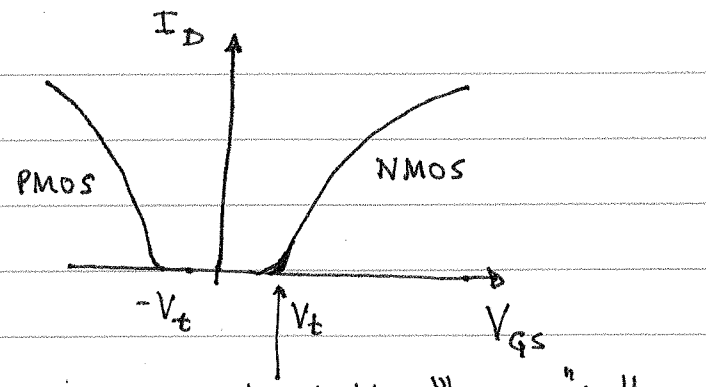


The output Q is relatively insensitive to the precise input voltage, or to noise on its input.

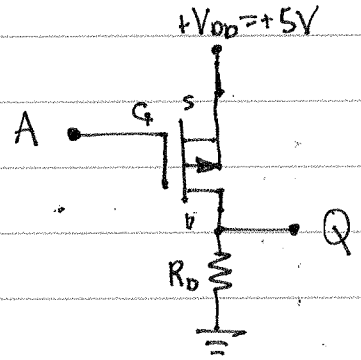
- Inverter using an n-channel (enhancement mode) MOSFET



NMOS Inverter



... or p-channel

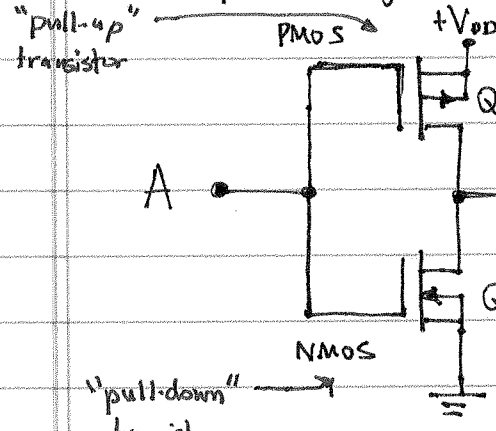


If A is low $V_{GS} < -|V_t|$, FET is "ON".
I_D is Big and Q is high

If A is high $V_{GS} > -|V_t|$, FET is "OFF"
$I_D = 0$ and Q is low

Note that power is dissipated in one of the logic states for both the NMOS and the PMOS inverter.

- CMOS Inverter
  - "complementary" ... uses both NMOS and PMOS



If A is low, Q_1 is off and Q_2 is on
Q is high

If A is high, Q_1 is on and Q_2 is off
Q is low

No current flows in either state (except to charge stray or load capacitance).

- Logic gates with two (or more) input variables:

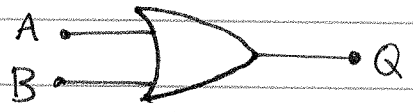
OR gate: A "OR" B means if A OR B is high (true) then Q (output) is high (true), otherwise Q is low (false).

Symbolically $A + B = Q$

Truth table

A	B	Q
0	0	0
0	1	1
1	0	1
1	1	1

Electronic circuit schematic



What is in there? Mostly BJT's → Transistor-transistor logic family (TTL)

or Mostly MOSFET's → CMOS logic family

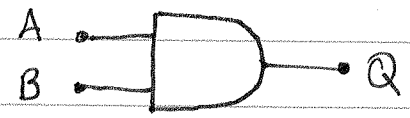
AND gate: If A and B are high, then Q is high, otherwise Q is low.

Truth table

A	B	Q
0	0	0
0	1	0
1	0	0
1	1	1

Symbolically $A \cdot B = Q$ or $AB = Q$

Electronic circuit



- Combining the inverter with OR and AND gates

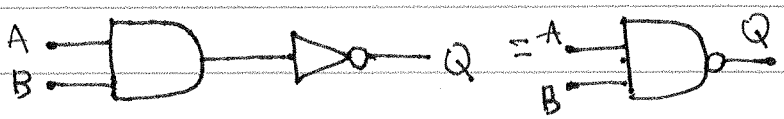
NAND = NOT AND or ... "negated output AND"

Symbolically $\overline{A \cdot B} = Q$

Truth table

A	B	Q
0	0	1
0	1	1
1	0	1
1	1	0

Electronic schematic



NOR = NOT OR or "negated output OR"

Symbolically $\overline{A + B} = Q$

Truth table

A	B	Q
0	0	1
0	1	0
1	0	0
1	1	0

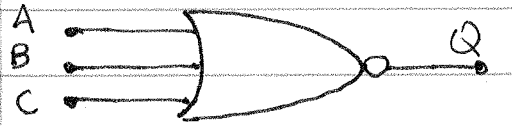
Electronic schematic

- Gates with more than two inputs

3-input NOR $\overline{A+B+C} = Q$

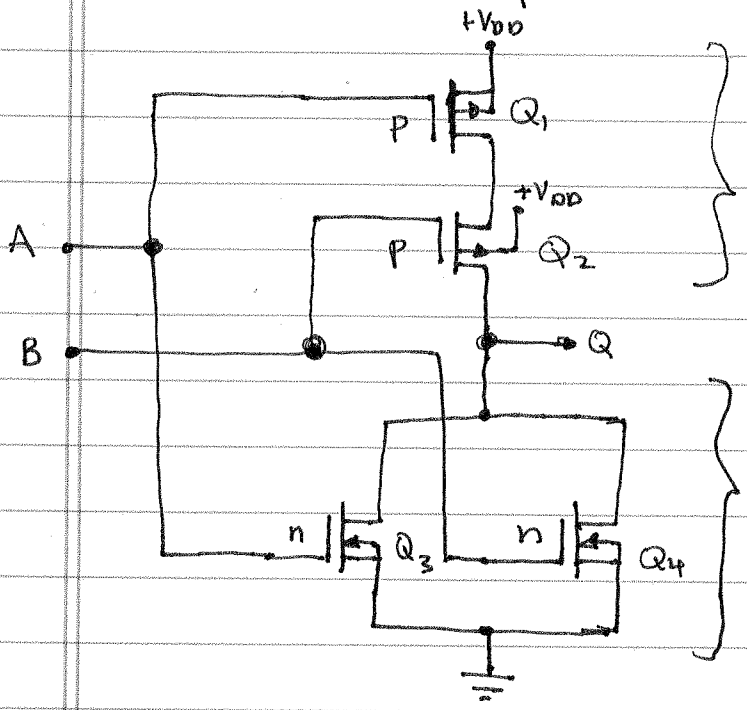
Truth table

A	B	C	Q
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	0



Can also have 3 (or more) input AND, NAND, OR gates

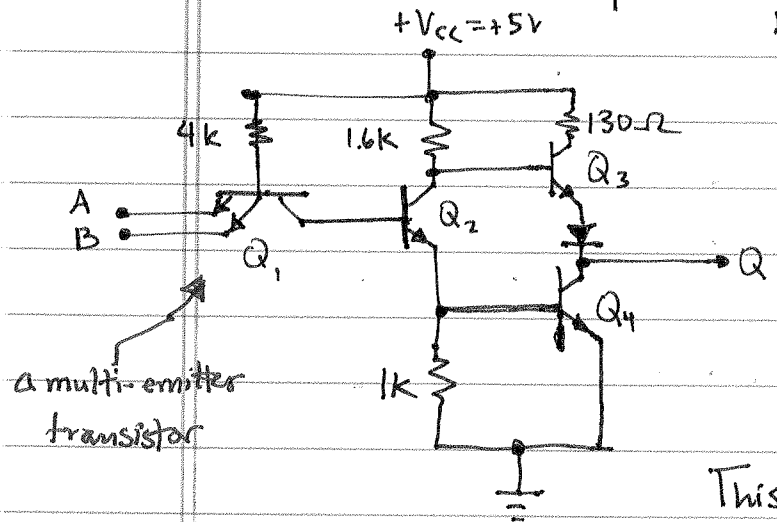
- A CMOS Gate Example



A	B	Q_1	Q_2	Q_3	Q_4	Q
0	0	ON	ON	OFF	OFF	1
0	1	ON	OFF	OFF	ON	0
1	0	OFF	ON	ON	OFF	0
1	1	OFF	OFF	ON	ON	0

This is a NOR gate

- A TTL Gate Example (also, see Horowitz + Hill, Fig. 8.17 on p. 485 for the Low-Power Schottky version LSTTL)



A	B	Q_1	Q_2	Q_3	Q_4	Q
0	0	on	off	on	off	1
0	1	on	off	on	off	1
1	0	on	off	on	off	1
1	1	off	on	off	on	0

This is a NAND gate.

- DeMorgan's Theorem (or "Laws")

Consider the truth table for a "negated input AND"

Not the same as a NAND which is a negated output AND.

$\overline{A} \cdot \overline{B} = 0$

(NOT A) AND (NOT B)

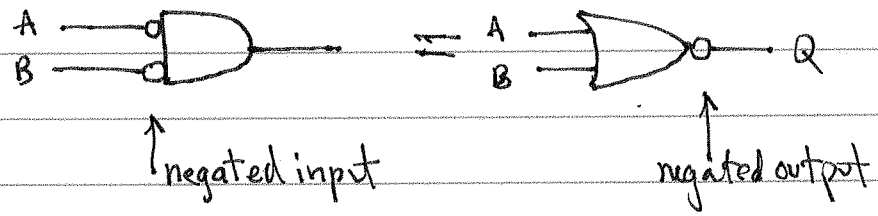
A	B	Q
0	0	1
0	1	0
1	0	0
1	1	0

This IS the same as the truth table for the negated output OR → NOR

So, $\overline{A} \cdot \overline{B} = \overline{A+B}$

This is an example of DeMorgan's Theorem

In electronic gate schematic representation



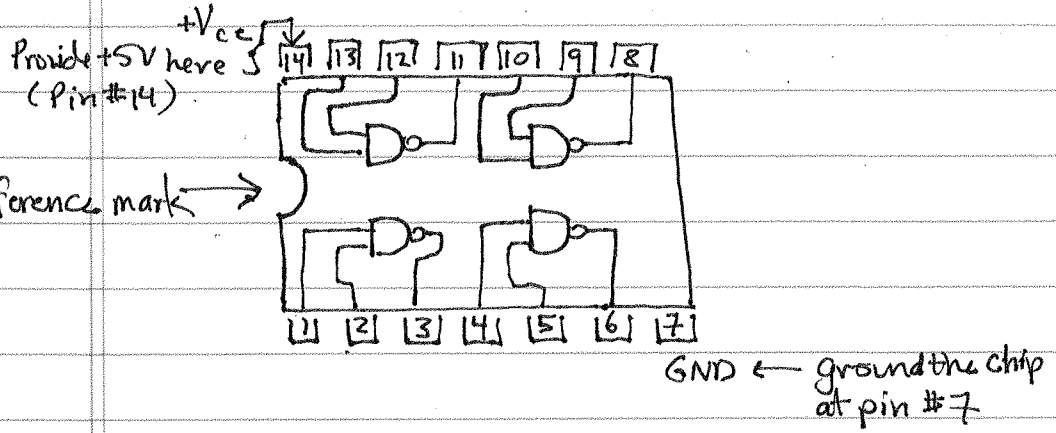
Another example of DeMorgan's Theorem

$\overline{A} + \overline{B} = \overline{A \cdot B}$ (NOT A) OR (NOT B) = NOT (A and B)

NAND

- All Boolean logic operations can be constructed from combinations of NAND gates alone (or NOR gates alone

- The 7400 IC Chip : "Quad two-input NAND"



This is the chip you will use in lab to explore logic gates and their combinations.

- Some exercises

— What does this gate do?

— Make an AND gate using two NAND gates

— From DeMorgan's Theorem $\overline{A+B} = \bar{A} \cdot \bar{B}$

negate both sides to get $A+B = \overline{\bar{A} \cdot \bar{B}}$

Use this result to make an OR gate from 3 NAND gates.

— The Exclusive OR (XOR) operation has this truth table

In lab you will see how it can be constructed from 4 NAND gates. Electronics symbol

$A \oplus B = Q$

A	B	Q
0	0	0
0	1	1
1	0	1
1	1	0

- Now let's begin doing some binary arithmetic

Binary (or base 2) numbers have only two values (0 or 1) in each digit (bit).

Counting in binary (use a 4 bit word example ... a "nibble". 8 bits = 1 byte)

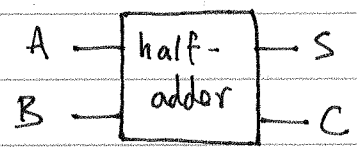
Decimal (base 10)	Binary (base 2)
00	0000
01	0001
02	0010
03	0011
04	0100
05	0101
06	0110
07	0111
08	1000
09	1001
10	1010
11	1011
12	1100
13	1101
14	1110
15	1111

4 bit can express $2^4 = 16$ possible values

The TDS 2004B Digital Oscilloscopes you have been using convert the analog input voltage to an 8-bit (1 byte) binary number at each sample time ... $2^8 = 256$ possible values.

- Binary addition (1 bit)

Truth table for the "half-adder"



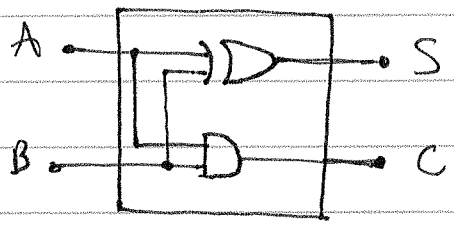
A	B	S	C
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

This unit is only useful for adding the least significant (right most) bit (digit). We need to have a carry input to make this work for other bits

↑ Sum ↑ Carry 1 to next more significant bit

Note that $S = A \oplus B$ and $C = A \cdot B$

So, the half-adder is..

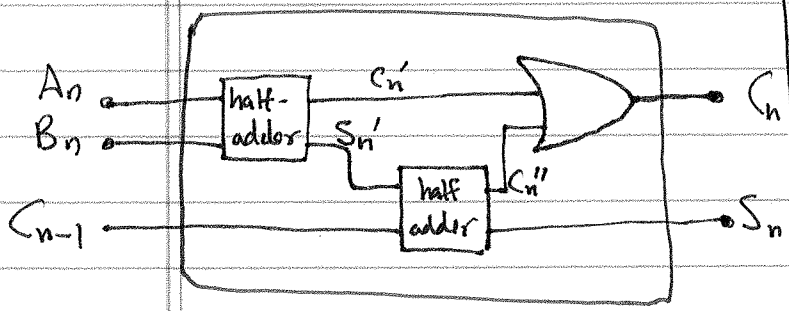


The Full-adder for the n^{th} bit... truth table

↓ carry from next less significant bit.

A_n	B_n	C_{n-1}	S_n	C_n
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

Show that this combination works



$S_n = (A_n \oplus B_n) \oplus C_{n-1}$

$C_n = (A_n \cdot B_n) + C_{n-1} \cdot (A_n \oplus B_n)$

- Parallel adder for two 4-bit words: $A_3A_2A_1A_0 + B_3B_2B_1B_0$

addition →