INTERNAL STRUCTURES OF ELECTRONIC DIGITAL CIRCUTS

- So far, we have seen abstract logic gates such as AND, OR, NAND, NOT, and so on.
- There are many different ways of implementing a logic gate as an electronic circuit.
- In this lecture, we will discuss how different types of transistors are used to design electronic logic circuits.
- In digital circuits, transistors are always used as **switches** (ON or OFF).
- First, we introduce the **bipolar junction transistor (BJT)**. A bipolar junction transistor is a three-terminal device that, in most logic circuits, acts like a current-controlled switch.
- Then, we will introduce the **MOSFET** (metal-oxide-semiconductor field-effect transistor) or simply the MOS transistor, which is used by almost all new integrated circuits.

Bipolar Junction Transistor:

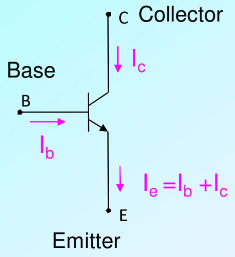
The Base is the control terminal.

- **State 1 (OFF)** $V_{BE} < 0.6V$:

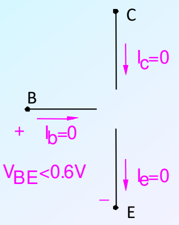
  If no current is flowing into the base, then no current can flow from the collector to the emitter (OFF).
- **State 2 (ON)** $V_{BE} \geq 0.6V$:

  However, if current is flowing from the base to the emitter, then current is also enabled to flow from the collector to the emitter (ON).

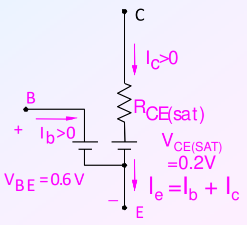
npn Bipolar Transistor:



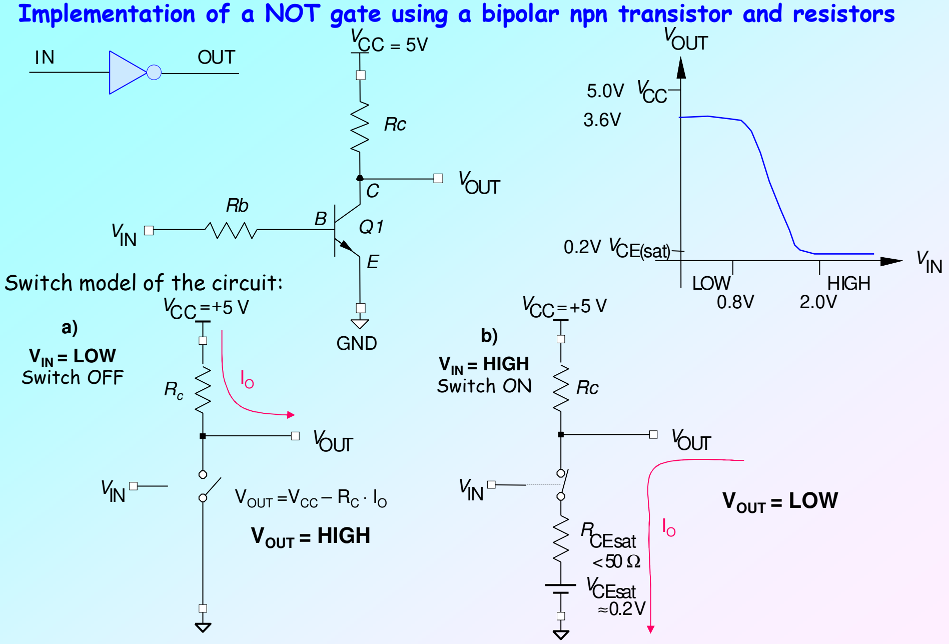
Transistor is cut off (OFF)
$V_{BE} < 0.6V$



Transistor is saturated (ON)
$V_{BE} \geq 0.6V$



Implementation of a NOT gate using a bipolar npn transistor and resistors



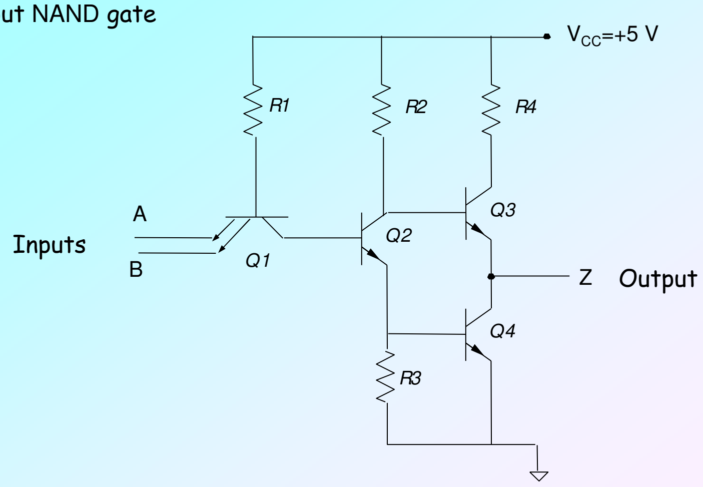
TTL (Transistor - Transistor) Logic Family

- Transistor–transistor logic (TTL) is a class of digital circuits built from bipolar junction transistors (BJTs) and resistors.
- Actually, there are many different TTL families (such as LS, ALS, L, F), with a range of speed, power consumption, and other characteristics.
  - Their current and voltage values can be found in data catalogs.
- The circuit examples in this section are based on a representative TTL family, Low-power Schottky (LS or LS-TTL).
- Although TTL has been largely replaced by CMOS (which we will discuss later), you may encounter TTL components in your labs; therefore, we will cover basic TTL concepts in this course.

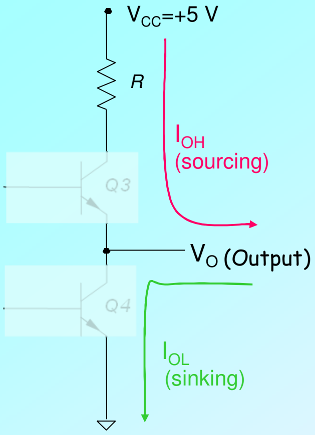
TTL (Transistor- Transistor) Logic Family

Example:

Two-input NAND gate



Operation of the Output Stage of a TTL Gate



- **Output is "0" (*LOW*):** Q_4 is ON, Q_3 is OFF.

  In this case, the current I_{OL} flows into the output. The output is said to be *sinking current*.

  $$V_{OL} = V_{CE(Q4)} + I_{OL} \cdot R_{Q4}$$

- **Output is "1" (*HIGH*):** Q_3 is ON, Q_4 is OFF.

  In this case, the current I_{OH} flows out of the output. The output is said to be *sourcing current*.

  $$V_{OH} = V_{CC} - (V_{CE(Q3)} + I_{OH} \cdot (R+R_{Q3}))$$

- If both Q_3 and Q_4 are OFF, the output is in the **high-impedance (Hi-Z)** state.
  - It is also called the **third state** (High, Low, Hi-Z).
  - In this state, the output behaves as if it is not even connected to the circuit.
  - The output is isolated from the line it was connected to.

The specifications guarantee that under all normal operating conditions, a logic 1 output voltage will always be greater than 2.4 V and a logic 0 output less than 0.4 V.

For TTL components $V_{OL(MAX)}$ = 0.4V $V_{OH(MIN)}$ = 2.4V (see slide 10.7)

TTL Logic Levels

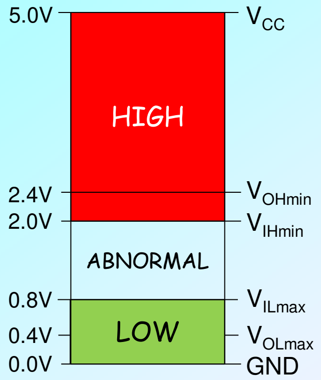
Abstract logic elements process binary digits, 0 and 1.

However, real logic circuits process electrical signals such as voltage levels.

In any logic circuit family, there is one range of voltages that is interpreted as logic 0 and another nonoverlapping range that is interpreted as logic 1.

TTL circuits are connected to 5-volt power supply (Vcc=5V).

Logic levels of a standard TTL unit:



V_{OHmin}: The minimum output voltage produced in the HIGH state.

V_{IHmin}: The minimum input voltage guaranteed to be recognized as a HIGH.

V_{ILmax}: The maximum input voltage guaranteed to be recognized as a LOW.

V_{OLmax}: The maximum output voltage produced in the LOW state.

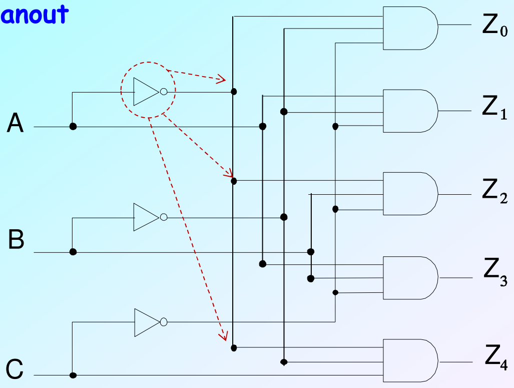
TTL Fanout

Remember that the output of a logic gate is connected to the inputs of other gates.

For example, in the circuit on the right, the first NOT drives inputs of three AND gates.

The **fanout** of a logic gate is the number of inputs that the gate can drive without exceeding its worst-case loading specifications.

Fanout is limited by the output current of the driving gate and the input current of the load gates.

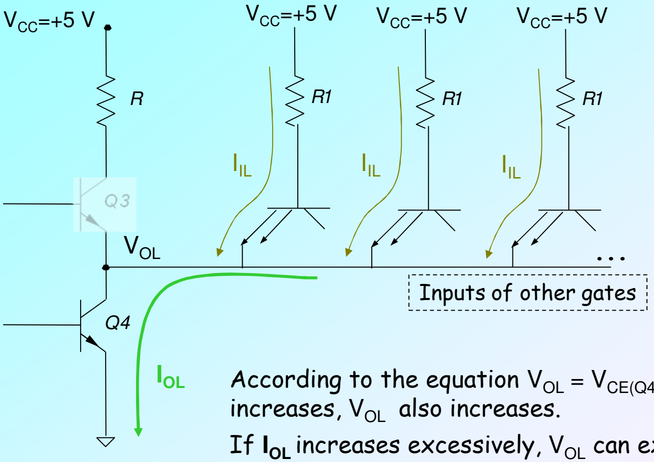


Fanout must be examined for both possible output states, HIGH and LOW.

The *overall fanout* of a gate is the minimum of the numbers calculated for its LOW-state and HIGH-state fanouts. Overall Fanout = Min($\text{Fanout}_{\text{LOW}}$, $\text{Fanout}_{\text{HIGH}}$)

TTL Fanout (cont'd)

When the Output is LOW:



The current $\mathbf{I_{IL}}$ flows from the input of each component with a LOW input to the output of the gate.

The output of the gate sinks the sum of these currents.

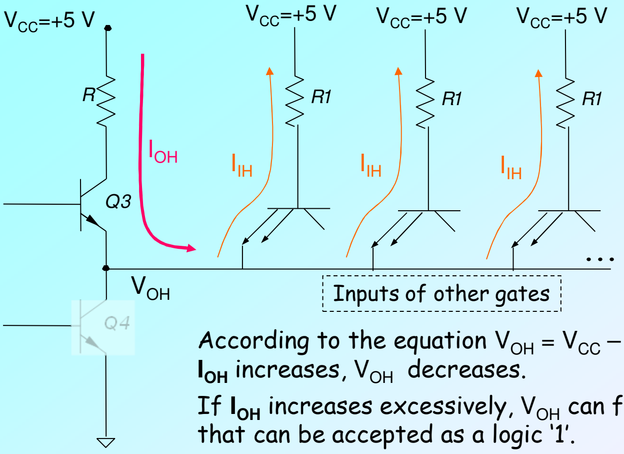
$$\mathbf{\Sigma I_{IL} < I_{OL} < I_{OLMAX}}$$

According to the equation $V_{OL} = V_{CE(Q4)} + I_{OL} \cdot R_{Q4}$, if $\mathbf{I_{OL}}$ increases, V_{OL} also increases.

If $\mathbf{I_{OL}}$ increases excessively, V_{OL} can exceed the limit value that can be accepted as a '0'.

This condition must be satisfied: $V_{OL} < V_{ILmax}$ (see slide 10.7)

When the Output is HIGH:



The leakage current I_{IH} flows into the input of each component with a HIGH input.

The output of the gate sources the sum of these currents.

$$\mathbf{\Sigma I_{IH} < I_{OH} < I_{OHMAX}}$$

According to the equation $V_{OH} = V_{CC} – (V_{CE(Q3)} + I_{OH} \cdot (R+R_{Q3}))$, if $\mathbf{I_{OH}}$ increases, V_{OH} decreases.

If $\mathbf{I_{OH}}$ increases excessively, V_{OH} can fall below the limit value that can be accepted as a logic '1'.

This condition must be satisfied: $V_{OH} > V_{IHmin}$ (see slide 10.7)

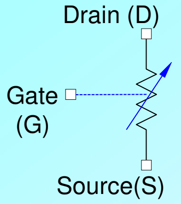
Note that the HIGH-state and LOW-state fanouts of a gate are not necessarily equal.

The *overall fanout* of a gate is the minimum of the numbers calculated for its LOW-state and HIGH-state fanouts. Overall Fanout = Min(Fanout_{LOW}, Fanout_{HIGH})

CMOS (Complementary MOS) Logic Family

CMOS logic performs logic functions using a combination of MOSFETs (*Metal-Oxide-Semiconductor Field-Effect Transistors*).

A MOS transistor can be modeled as a 3-terminal device that acts like a voltage-controlled resistor.

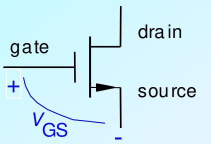


The voltage from Gate to Source (V_{GS}) controls the resistance between Drain and Source (R_{DS}).

- If the transistor is OFF, $R_{DS} \geq 1M\Omega$ ($10^6\Omega$)
- If the transistor is ON, $R_{DS} \leq 10\Omega$

There are two types of MOS transistors.

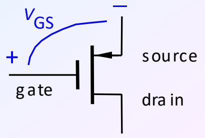
a) n-channel MOS: NMOS.



Increase V_{GS} → decrease R_{DS}

Normally: $V_{GS} \geq 0V$

b) p-channel MOS: PMOS.



Decrease V_{GS} → decrease R_{DS}

Normally: $V_{GS} \leq 0V$

CMOS Gate Implementation

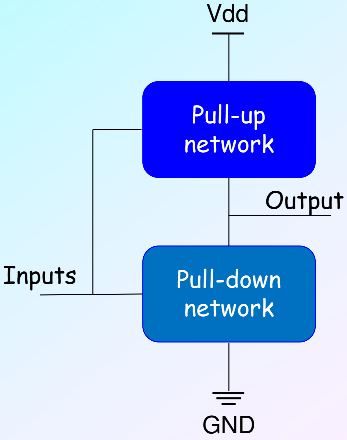
A CMOS logic gate consists of two parts:

A. Pull-UP network: p-tree for pull-up

- It has to be connected to Vdd to pull up the output.
- Build using pMOS connected in parallel or in series
- Turns on when the function is true

B. Pull-DOWN network: n-tree for pull-down

- It has to be connected to GND to pull down the output.
- Build using nMOS connected in parallel or in series
- Turns on when the function is false



These two networks are operationally complements:

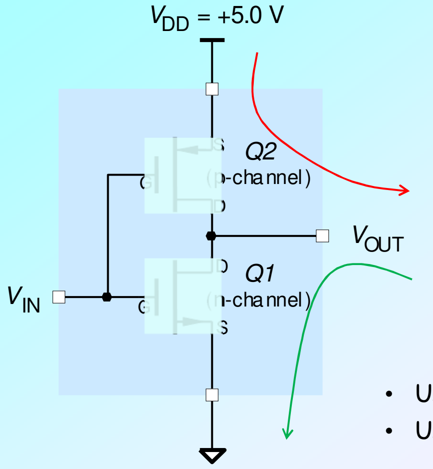
- Pull-up network is complement of pull-down
- Parallel -> series, series -> parallel

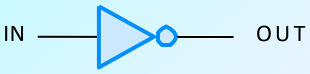
CMOS NOT Gate

Remember: complementary and symmetrical pairs of NMOS and PMOS transistors are used to form CMOS logic.

For each NMOS transistor, there is a PMOS transistor in the circuit.

Example: CMOS NOT Gate



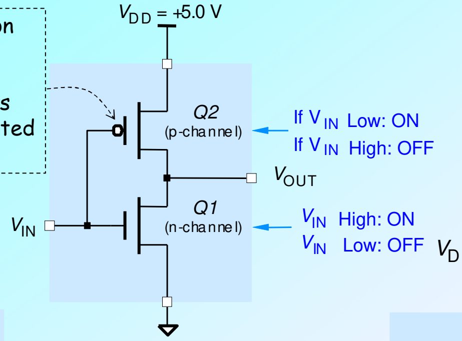


V_{IN}	Q1	Q2	V_{OUT}
0.0 (L)	off	on	5.0 (H)
5.0 (H)	on	off	0.0 (L)

- Use pMOS to produce logic '1' -> Pull UP
- Use nMOS to produce logic '0' -> Pull DOWN

CMOS NOT Gate Switch Model

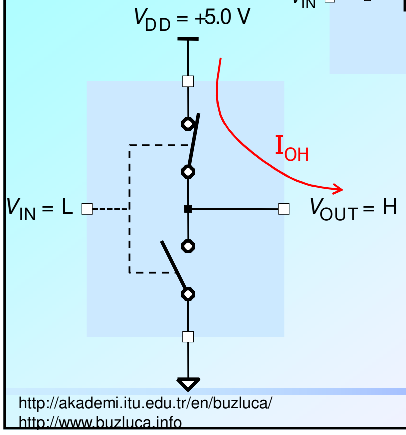
Alternative representation of a pMOS transistor to reflect the fact that it is ON when a LOW voltage is applied to its Gate (indicated by the inversion bubble)



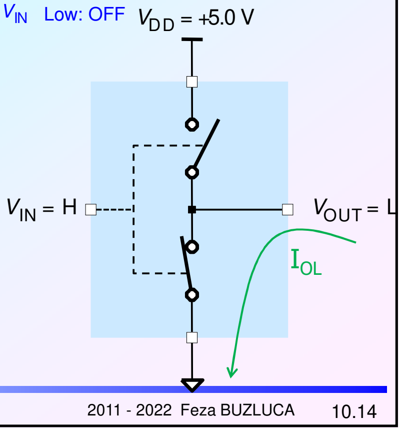
If V_{IN} Low: ON
If V_{IN} High: OFF

V_{IN} High: ON
V_{IN} Low: OFF

$\overline{0}$ = 1 State:

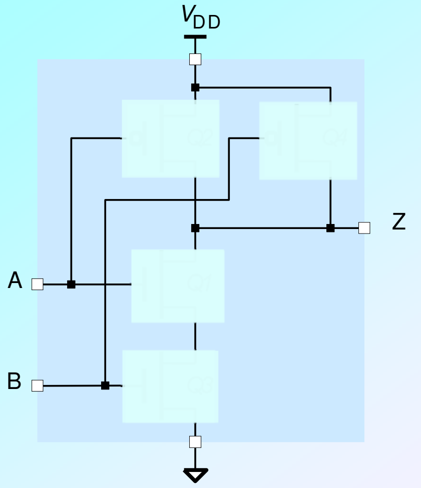


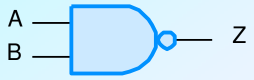
$\overline{1}$ = 0 State:



CMOS NAND Gate

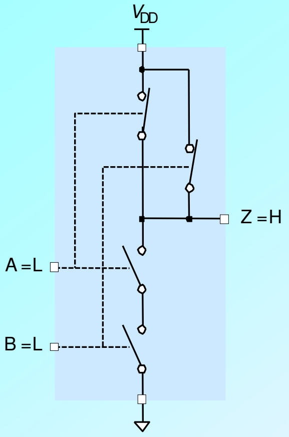
- Pull-up network: pMOS transistors connected in parallel.
- Pull-down network: nMOS transistors connected in series



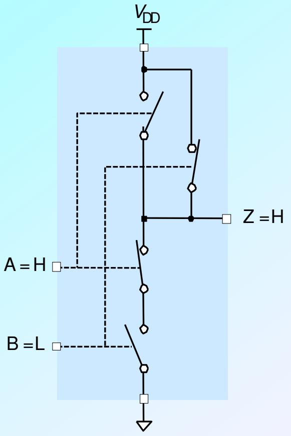


A	B	Q1	Q2	Q3	Q4	Z

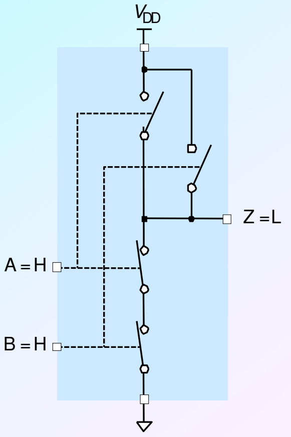
CMOS NAND Gate Switch Model



0 NAND 0 = 1



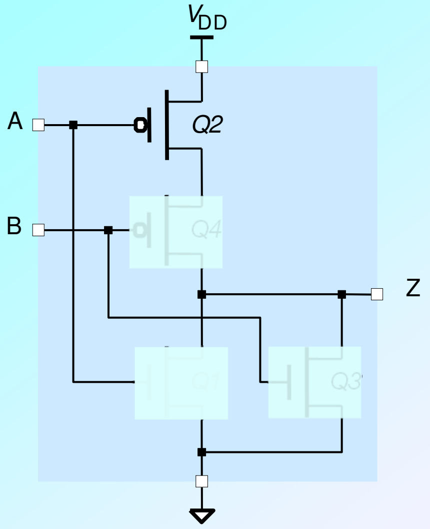
0 NAND 1 = 1

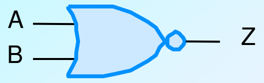


1 NAND 1 = 0

CMOS NOR Gate

- Pull-up network: pMOS transistors connected in series.
- Pull-down network: nMOS transistors connected in parallel.

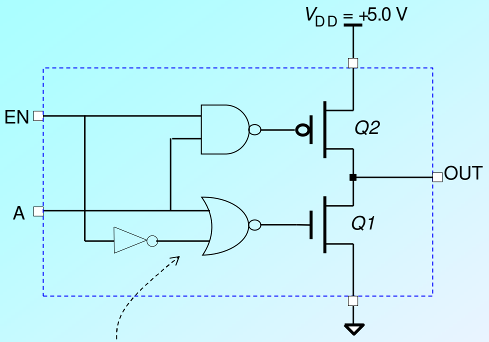




A	B	Q1	Q2	Q3	Q4	Z
L	L	off	on	off	on	H
L	H	off	on	on	off	L
H	L	on	off	off	on	L
H	H	on	off	on	off	L

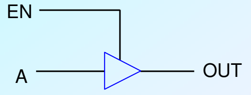
CMOS Three-State Buffer

Remember that if an output is in **high-impedance (Hi-Z)** state (also called the **third state**), the output behaves as if it is not even connected to the circuit.



To simplify the diagram, the internal NAND, NOR, and NOT functions are represented by their own symbols rather than by their constituent transistors.
These elements actually consist of 10 transistors.

Three-state buffer:



IF EN=HIGH THEN OUT = A
IF EN=LOW THEN OUT = Hi-Z

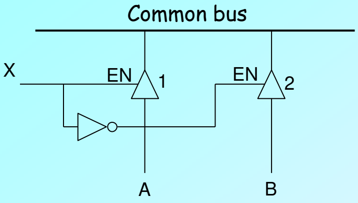
EN	A	Q1	Q2	OUT
L	L	off	off	Hi-Z
L	H	off	off	Hi-Z
H	L	on	off	L
H	H	off	on	H

Three-State Common Bus

Several three-state outputs can be wired together to form a three-state common bus.

At any given moment, only one unit is enabled to drive the bus.

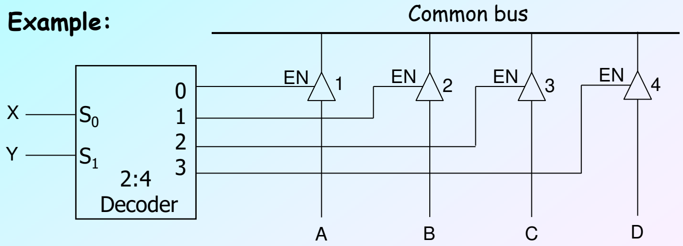
Example:



If X=0, buffer #2 drives the bus. B is on bus.

If X=1, buffer #1 drives the bus. A is on bus.

Example:

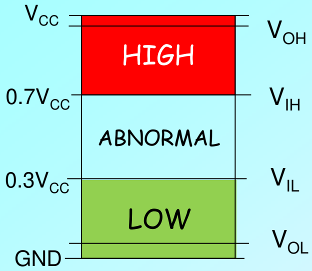


Y	X	Bus
0	0	A
0	1	B
1	0	C
1	1	D

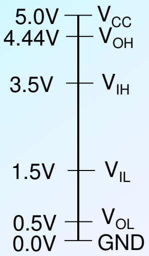
CMOS Logic Levels

CMOS circuits can use power supplies with a voltage less than 5 Volts.

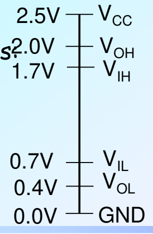
Logic levels change according to the voltage of the power supply (V_{CC}).



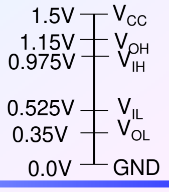
5-V CMOS families:



2.5-V CMOS families:



1.5-V CMOS families:


1.5V
VCC
1.15V
VOH
0.975V
VIH
0.525V
VIL
0.35V
VOL
0.0V
GND