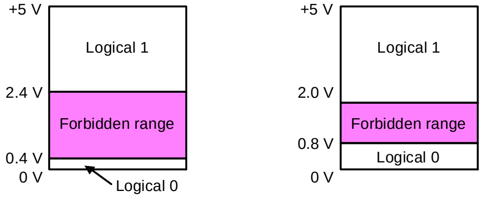
Chapter 2: Gates, Circuits, and Combinational Logic

Dr. Tim McGuire
Sam Houston State University
Based on notes by
Miles Murdocca

Analog and Digital Systems

- **An *analog circuit* can have any value between its maximum and minimum limits**
- **A *digital circuit* (at least in concept) has one of a fixed number of values and changes from one value to another instantaneously**
  - **Digital electronic circuits use a binary system, with two values (0 and 1)**
  - **Ideally, if a computer runs off 5V, a 0 (false, low, off) value would be represented by 0.0 V and 1 (true, high, on) by +5.0V**
    - **This is TTL (which is common but being replaced by faster and cooler devices)**
    - **We can't unfortunately, construct devices with such precision, so we assign *ranges* of values to represent 0 and 1**

Assignments of Logical 0 and Logical 1 to Voltage Ranges



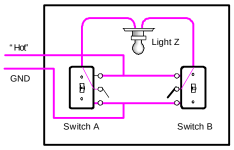
(a) At the output of a logic gate

(b) At the input to a logic gate

Truth Tables

- **Developed in 1854 by George Boole**
- **Further developed by Claude Shannon (Bell Labs)**
- **Outputs are computed for all possible input combinations (how many input combinations are there?**

Consider a room with two light switches. How must they work[†]?



Inputs		Output
A	B	Z
0	0	0
0	1	1
1	0	1
1	1	0

[†]Don't show this to your electrician, or wire your house this way. This circuit definitely violates the electric code. The practical circuit never leaves the lines to the light "hot" when the light is turned off. Can you figure how?

Truth Tables Showing All Possible Functions of Two Binary Variables

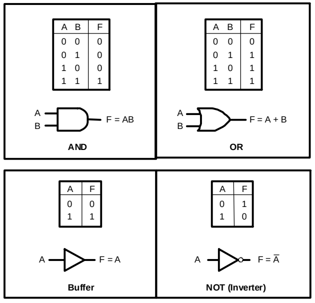
A	B	False	AND	$A\overline{B}$	A	$\overline{A}B$	B	XOR	OR
0	0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	1	1	1	1
1	0	0	0	1	1	0	0	1	1
1	1	0	1	0	1	0	1	0	1

A	B	NOR	XNOR	$\overline{B}$	$A + \overline{B}$	$\overline{A}$	$\overline{A} + B$	NAND	True
0	0	1	1	1	1	1	1	1	1
0	1	0	0	0	0	1	1	1	1
1	0	0	0	1	1	0	0	1	1
1	1	0	1	0	1	0	1	0	1

- **The more frequently used functions have names: AND, XOR, OR, NOR, XNOR, and NAND. (Always use upper-case spelling.)**

Logic Gates and Their Symbols

Logic Gate Symbols for *AND*, *OR*, Buffer, and *NOT* Boolean functions



A	B	F
0	0	0
0	1	0
1	0	0
1	1	1

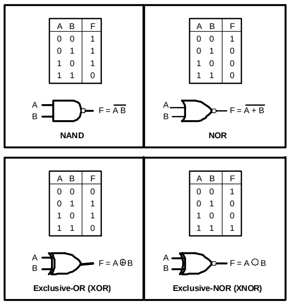
A	B	F
0	0	0
0	1	1
1	0	1
1	1	1

A	F
0	0
1	1

A	F
0	1
1	0

- **Note the use of the "inversion bubble."**
- **Be careful about the "nose" of the gate when drawing AND vs. OR.**

Logic Gate Symbols for *NAND*, *NOR*, *XOR*, and *XNOR* Boolean functions



A	B	F
0	0	1
0	1	1
1	0	1
1	1	0

NAND

A	B	F
0	0	1
0	1	0
1	0	0
1	1	0

NOR

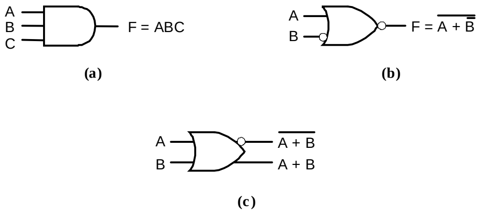
A	B	F
0	0	0
0	1	1
1	0	1
1	1	0

Exclusive-OR (XOR)

A	B	F
0	0	1
0	1	0
1	0	0
1	1	1

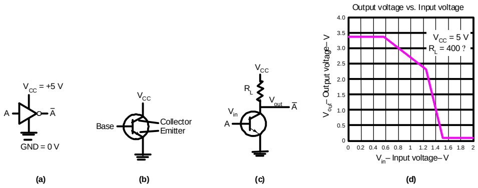
Exclusive-NOR (XNOR)

Variations of Basic Logic Gate Symbols



(a) 3 inputs (b) A negated input (c) Complementary outputs

The Inverter at the Transistor Level



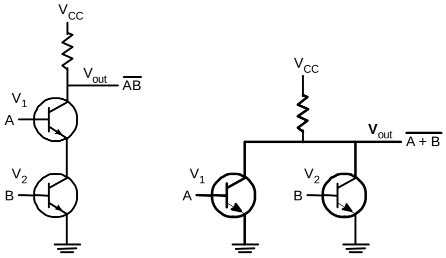
Power terminals for an inverter made visible

Transistor symbol

A transistor used as an inverter

Inverter transfer function

Transistor Circuits



(a) A two-input NAND gate (b) A two-input NOR gate

The Basic Properties of Boolean Algebra

Relationship	Dual	Property
$AB = BA$	$A + B = B + A$	Commutative
$A(B + C) = AB + AC$	$A + BC = (A + B)(A + C)$	Distributive
$1A = A$	$0 + A = A$	Identity
$A\bar{A} = 0$	$A + \bar{A} = 1$	Inverse
$0A = 0$	$1 + A = 1$	Null
$AA = A$	$A + A = A$	Idempotence
$A(BC) = (AB)C$	$A + (B + C) = (A + B) + C$	Associative
$\bar{\bar{A}} = A$		Complement
$\overline{AB} = \bar{A} + \bar{B}$	$\overline{A + B} = \bar{A}\bar{B}$	DeMorgan's Theorem
$AB + \bar{A}C + BC = AB + \bar{A}C$	$(\bar{A} + B)(A + C)(B + C) = (A + B)(\bar{A} + C)$	Consensus Theorem

Principle of duality: The dual of a Boolean function is gotten by replacing AND with OR and OR with AND, constant 1s by 0s, and 0s by 1s

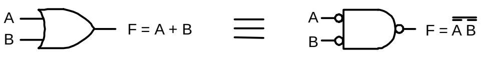
Postulates

Theorems

DeMorgan's Theorem

A	*B*	$\overline{AB}$	= $\bar{A} + \bar{B}$	$\overline{A + B}$	= $\bar{A}\,\bar{B}$
0	0	1	1	1	1
0	1	1	1	0	0
1	0	1	1	0	0
1	1	0	0	0	0

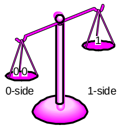
DeMorgan's theorem: $A + B = \overline{\overline{A + B}} = \overline{\bar{A}\,\bar{B}}$



The Sum-of-Products (SOP) Form

Truth Table for the *Majority Function*

Minterm Index	A	B	C	F
0	0	0	0	0
1	0	0	1	0
2	0	1	0	0
3	0	1	1	1
4	1	0	0	0
5	1	0	1	1
6	1	1	0	1
7	1	1	1	1



A balance tips to the left or right depending on whether there are more 0's or 1's.

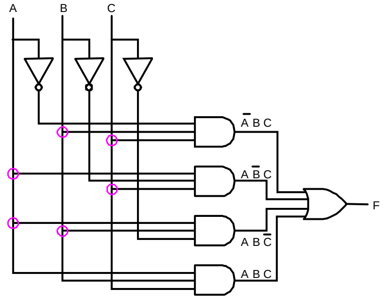
- **Transform the function into a two-level AND-OR equation**
- **Implement the function with an arrangement of logic gates from the set {AND, OR, NOT}**
- ***M* is true when A = 0, B = 1, and C = 1, or when A = 1, B = 0, and C = 1, and so on for the remaining cases.**
- **Represent logic equations by using the sum-of-products (SOP) form**

The SOP Form of the Majority Gate

- **The SOP form for the 3-input majority gate is:**
- **M = $\overline{A}BC + A\overline{B}C + AB\overline{C} + ABC$ = m3 + m5 +m6 +m7 = ? ?(3, 5, 6, 7)**
- **Each of the 2^n terms are called minterms, running from 0 to 2^n - 1**

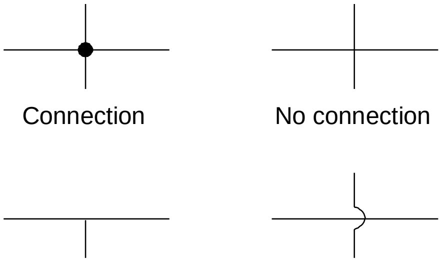
- **Note the relationship between minterm number and Boolean value.**
- **Discuss: common-sense interpretation of equation.**

A Two-Level AND-OR Circuit Implements the Majority Function



Discuss: what is the gate count?

Four Notations Used at Circuit Intersections



Connection

No connection

Connection

No connection

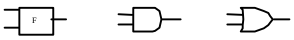
Positive versus Negative Logic

•Positive logic: truth, or assertion is represented by logic 1, higher voltage; falsity, de- or unassertion, logic 0, is represented by lower voltage.
•Negative logic: truth, or assertion is represented by logic 0 , lower voltage; falsity, de- or unassertion, logic 1, is represented by lower voltage

Gate Logic: Positive vs. Negative Logic

Normal Convention: Postive Logic/Active High
Low Voltage = 0; High Voltage = 1

Alternative Convention sometimes used: Negative Logic/Active Low



Voltage Truth Table

A	B	F
low	low	low
low	high	low
high	low	low
high	high	high

Positive Logic

A	B	F
0	0	0
0	1	0
1	0	0
1	1	1

Negative Logic

A	B	F
1	1	1
1	0	1
0	1	1
0	0	0

Behavior in terms of Electrical Levels

Two Alternative Interpretations
Positive Logic AND
Negative Logic OR

Dual Operations

Positive and Negative Logic Assignments

Voltage levels

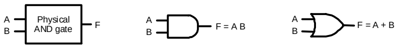
A	B	F
low	low	low
low	high	low
high	low	low
high	high	high

Positive logic levels

A	B	F
0	0	0
0	1	0
1	0	0
1	1	1

Negative logic levels

A	B	F
1	1	1
1	0	1
0	1	1
0	0	0



Voltage levels

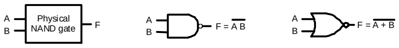
A	B	F
low	low	high
low	high	high
high	low	high
high	high	low

Positive logic levels

A	B	F
0	0	1
0	1	1
1	0	1
1	1	0

Negative logic levels

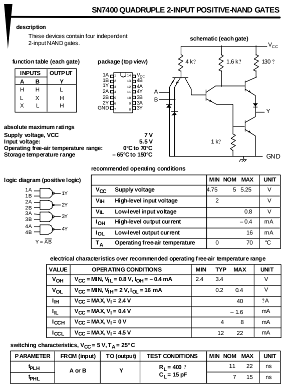
A	B	F
1	1	0
1	0	0
0	1	0
0	0	1



Digital Components

- High-level digital circuit designs are normally made using collections of logic gates referred to as *components*, rather than using individual logic gates. The majority function can be viewed as a component.
- Levels of integration (numbers of gates) in an integrated circuit (IC) can be roughly considered as:
  - Small-scale integration (SSI): 10–100 gates.
  - Medium-scale integration (MSI): 100–1000 gates.
  - Large-scale integration (LSI): 1000–10,000 logic gates.
  - Very large scale integration (VLSI): 10,000–upward.
- These levels are approximate, but the distinctions are useful in comparing the relative complexity of circuits.
- Let us consider several useful MSI components.

Simplified Data Sheet for 7400 NAND gate



SN7400 QUADRUPLE 2-INPUT POSITIVE-NAND GATES

description

These devices contain four independent 2-input NAND gates.

function table (each gate)

INPUTS		OUTPUT
A	B	Y
H	H	L
L	X	H
X	L	H

package (top view)

1A 1
1B 2
1Y 3
2A 4
2B 5
2Y 6
GND 7
8 3Y
9 3A
10 3B
11 4Y
12 4A
13 4B
14 VCC

schematic (each gate)

VCC
4 k?
1.6 k?
130 ?
A
B
Y
1 k?
GND

absolute maximum ratings

Supply voltage, VCC	7 V
Input voltage:	5.5 V
Operating free-air temperature range:	0°C to 70°C
Storage temperature range	– 65°C to 150°C

logic diagram (positive logic)

$Y = \overline{AB}$

recommended operating conditions

		MIN	NOM	MAX	UNIT
V_{CC}	Supply voltage	4.75	5	5.25	V
V_{IH}	High-level input voltage	2			V
V_{IL}	Low-level input voltage			0.8	V
I_{OH}	High-level output current			– 0.4	mA
I_{OL}	Low-level output current			16	mA
T_A	Operating free-air temperature	0		70	°C

electrical characteristics over recommended operating free-air temperature range

VALUE	OPERATING CONDITIONS	MIN	TYP	MAX	UNIT
V_{OH}	V_{CC} = MIN, V_{IL} = 0.8 V, I_{OH} = – 0.4 mA	2.4	3.4		V
V_{OL}	V_{CC} = MIN, V_{IH} = 2 V, I_{OL} = 16 mA		0.2	0.4	V
I_{IH}	V_{CC} = MAX, V_I = 2.4 V			40	?A
I_{IL}	V_{CC} = MAX, V_I = 0.4 V			– 1.6	mA
I_{CCH}	V_{CC} = MAX, V_I = 0 V		4	8	mA
I_{CCL}	V_{CC} = MAX, V_I = 4.5 V		12	22	mA

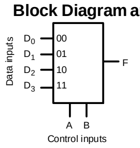
switching characteristics, V_{CC} = 5 V, T_A = 25° C

PARAMETER	FROM (input)	TO (output)	TEST CONDITIONS	MIN	NOM	MAX	UNIT
t_{PLH}	A or B	Y	R_L = 400 ? C_L = 15 pF		11	22	ns
t_{PHL}					7	15	ns

The Multiplexer (MUX)

Block Diagram and Truth Table

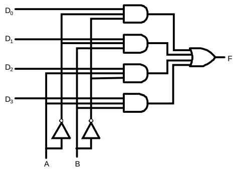
This is a 4-to-1 Multiplexer



A	B	F
0	0	D_0
0	1	D_1
1	0	D_2
1	1	D_3

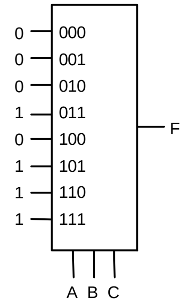
$F = \overline{A}\,\overline{B}\,D_0 + \overline{A}\,B\,D_1 + A\,\overline{B}\,D_2 + A\,B\,D_3$

AND-OR Circuit Implementation



An 8-1 MUX Can Implement the Majority Function

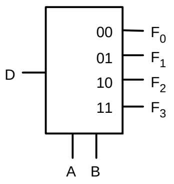
A	B	C	M
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1



Principle: Use the 3 MUX control inputs to select (one at a time) the 8 data inputs

The Demultiplexer (DEMUX)

Block Diagram and Truth Table



D	A	B	F_0	F_1	F_2	F_3
0	0	0	0	0	0	0
0	0	1	0	0	0	0
0	1	0	0	0	0	0
0	1	1	0	0	0	0
1	0	0	1	0	0	0
1	0	1	0	1	0	0
1	1	0	0	0	1	0
1	1	1	0	0	0	1

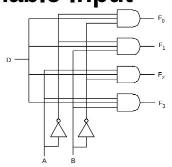
$F_0 = D\,\overline{A}\,\overline{B}$ $F_2 = D\,A\,\overline{B}$

$F_1 = D\,\overline{A}\,B$ $F_3 = D\,A\,B$

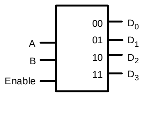
The Demultiplexer Is a Decoder with an Enable Input

A Circuit for a 1-4 DEMUX

Compare to Decoder on next slide



Block Diagram and Truth Table

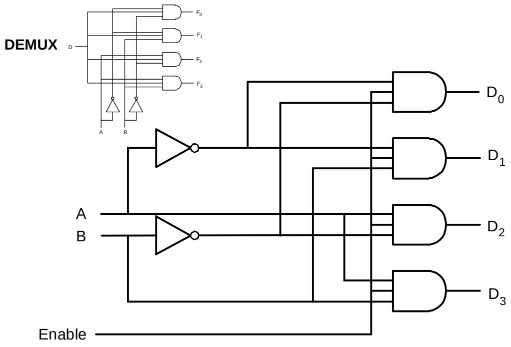


Enable = 1					
A	B	D_0	D_1	D_2	D_3
0	0	1	0	0	0
0	1	0	1	0	0
1	0	0	0	1	0
1	1	0	0	0	1

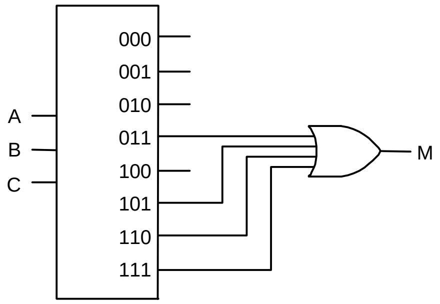
Enable = 0					
A	B	D_0	D_1	D_2	D_3
0	0	0	0	0	0
0	1	0	0	0	0
1	0	0	0	0	0
1	1	0	0	0	0

$D_0 = \overline{A}\,\overline{B}$ $D_1 = \overline{A}\,B$ $D_2 = A\,\overline{B}$ $D_3 = A\,B$

An AND Circuit for a 2-4 Decoder



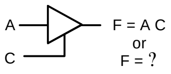
A 3-to-8 Decoder Used to Implement the Majority Function



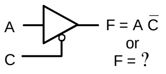
Tri-State Buffers

C	A	F
0	0	?
0	1	?
1	0	0
1	1	1

C	A	F
0	0	0
0	1	1
1	0	?
1	1	?



Tri-state buffer

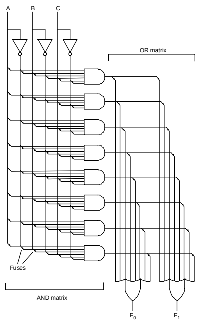


Tri-state buffer, inverted control

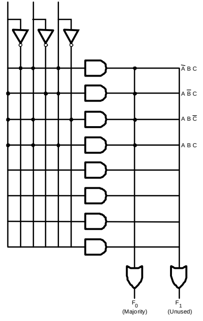
- There is a third state: high impedance. This means the gate output is essentially disconnected from the circuit.
- This state is indicated by ? in the figure.

A Programmable Logic Array

- A PLA is a customizable AND matrix followed by a customizable OR matrix



Simplified Representation of a PLA


A
B
C
$\overline{A}\,B\,C$
$A\,\overline{B}\,C$
$A\,B\,\overline{C}$
$A\,B\,C$
F_0
(Majority)
F_1
(Unused)