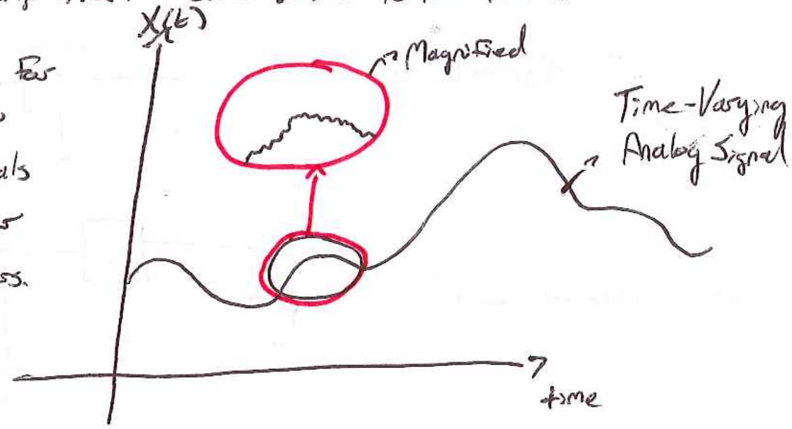
~~####~~ 4/21/2015 - LECTURE

Analog and Digital Systems

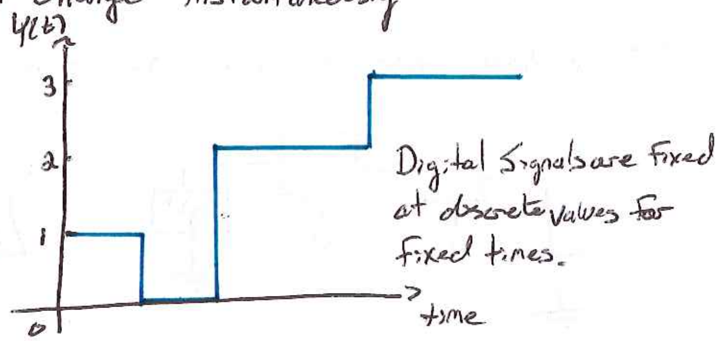
- Computers are constructed of digital logic circuits. Digital circuits may be described by Boolean algebra with the use of Karnough Maps.

- Analog circuits represent physical quantities in terms of voltages or currents.
  - Analog variables can take on an infinite number of values in a specific range of limits. X is a variable that is continuous in value and can change its value by only small amounts. X cannot change instantaneously or jump from one value to the next.

Examples:
- Old RCA jacks for Composite video
- Pure audio signals
- aneroid barometer
- Mercury Thermometers.



- Digital Circuits represent discrete values ~~[illegible]~~ at discrete times i.e. a discrete variable in both time and value. A digital variable Y can only take on discrete values or levels at discrete times and can change instantaneously



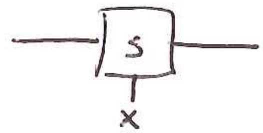
Two most common digital symbols: 0 and 1 (False and True) This is binary.
State is either 1 or 0. Each logic state has an inverse.
Common variable names: START, STOP, RESET, COUNT, ADD.

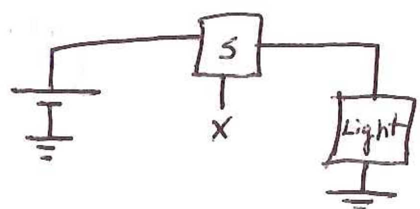
If a high voltage causes an action, then it is active-high. If a low-voltage causes an action then it is active low.

Positive Logic: 0 is false or low state, 1 is true or high state.

Common household switches 

Switches generally have 2 states (a binary operation). If the switch is controlled by an input x. Switch open is $x=0$, switch closed is $x=1$

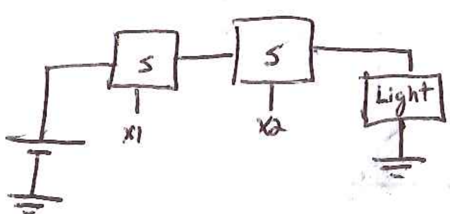
Symbol for a switch: 

light controlled switch 

Current flows when switch is closed on $x=1$.

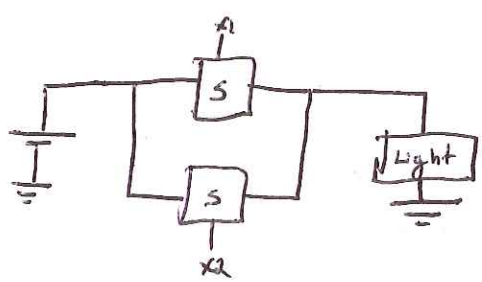
The output (light) is affected by the input x, if the light is on, $L=1$, if off, $L=0$
Therefor L is a function of the input x. $L=1$ if $x=1$, $L=0$ if $x=0$. Thus, $L(x)=x$ is a logical expression with an input variable

If we have two switches in series with separate inputs x_1 + x_2: $L(x_1,x_2)=x_1 \cdot x_2$
where $L=1$ if $x_1+x_2=1$
$L=0$ otherwise



Logical AND

$\boxed{L(x_1,x_2)=x_1 \cdot x_2}$

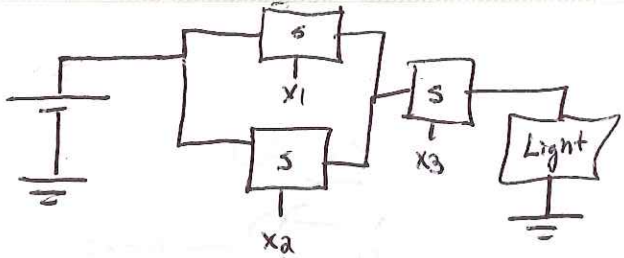


Logical OR

$\boxed{L(x_1,x_2)=x_1+x_2}$

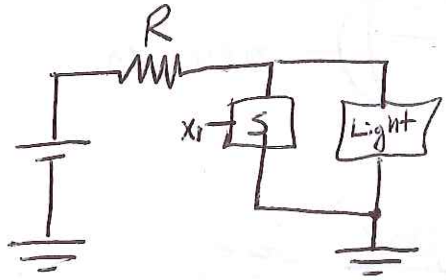
2

Series parallel connection



$$\boxed{L(x_1,x_2,x_3) = (x_1+x_2)\cdot x_3}$$

Inverting Circuit:



when switch is closed, current flows through short-circuit. Thus we have a NOT operation

$$\boxed{L(x) = \bar{x}}$$

L=1 if x=0

L=0 if x=1

Common representations of inversion/NOT

$$\boxed{\bar{x} = x' = !x = \sim x = \text{Not } x}$$

if $f(x_1,x_2) = x_1+x_2$

$\bar{f}(x_1,x_2) = \overline{x_1+x_2}$

Truth Tables of common Logic operations

x_1	x_2	AND $x_1\cdot x_2$	OR x_1+x_2
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	1

we can extend logic operators to more than 2 variables:

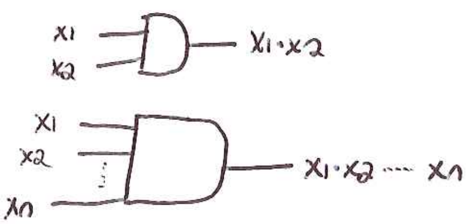
x_1	x_2	x_3	$x_1\cdot x_2\cdot x_3$	$x_1+x_2+x_3$
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	0	1
1	0	0	0	1
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

with binary and n inputs, we have 2^n possible states.

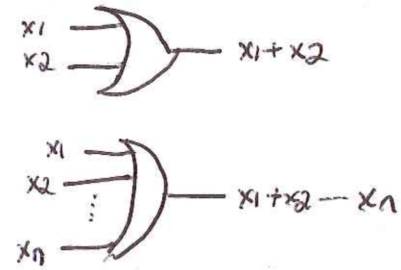
For AND operations only one instance results in 1; For OR operators, all but one instance results in 1.

Logic Gates

AND Gate

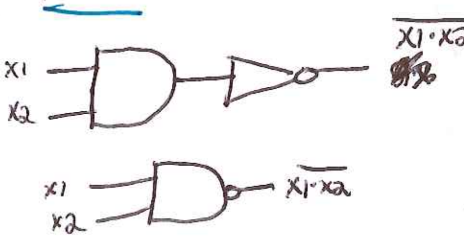


OR Gate

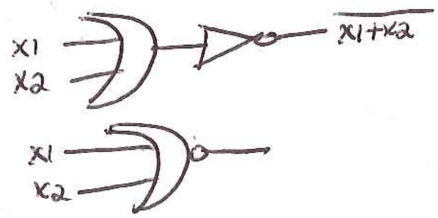


NOT Gate

NAND Gate



NOR Gate



XOR Gate

Exclusive Or

XNOR Gate

NOT Exclusive Or

XNOR Truth Table

x_1	x_2	Y
0	0	1
0	1	0
1	0	0
1	1	1

$\boxed{x_1 \text{ XOR } x_2 = x_1 \oplus x_2}$

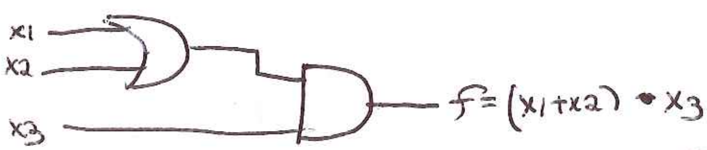
$\oplus$ = XOR

XOR Truth Table

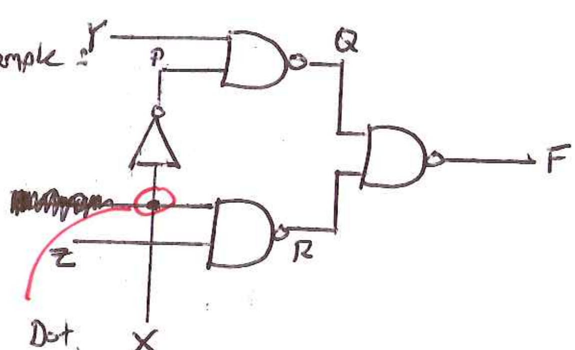
x_1	x_2	Y
0	0	0
0	1	1
1	0	1
1	1	0

XOR is similar to Modulus 2

Logic Gates together create networks and functions (mixed logic)



Example:



Dot means connection

Inputs			Intermediate Values			Output
X	Y	Z	$P = \overline{X}$	$Q = \overline{P \cdot Y}$	$R = \overline{X \cdot Z}$	$F = \overline{Q \cdot R}$
0	0	0	1	1	1	0
0	0	1	1	1	1	0
0	1	0	1	0	1	1
0	1	1	1	0	1	1
1	0	0	0	1	1	0
1	0	1	0	1	0	1
1	1	0	0	1	1	0
1	1	1	0	1	0	1

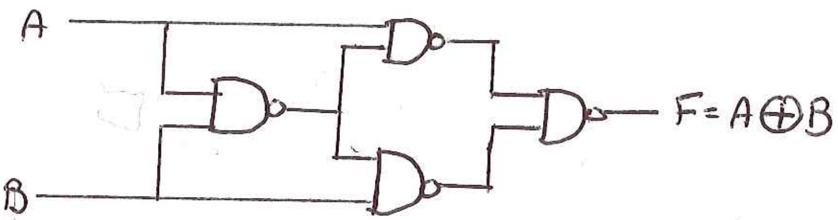
Gates and circuits can also compare words, but each gate and circuit does comparison on individual bits of words

word A : 11011100
word B : 01100101

if $C = A \cdot B$ | →

$$\begin{array}{r} 11011100 \\ 01100101 \\ \hline 01000100 \end{array} = C = A \cdot B$$

XOR circuit made of NAND Gates



Binary Addition and Subtraction

First let's look at Binary and decimal relationship

Consider 1234 in decimal. This means we have

1,	1000'	10^3
2,	100's	10^2
3,	10's	10^1
4,	1's	10^0

Decimal system is base10 (0-9)

Binary is base 2 (0-1), or radix 2.

$2^0 = 1$
$2^1 = 2$
$2^2 = 4$
$2^3 = 8$
$2^4 = 16$
$2^5 = 32$
$2^6 = 64$
$2^7 = 128$
$2^8 = 256$
$2^9 = 512$
$2^{10} = 1024$
$2^{11} = 2048$
$2^{12} = 4096$
$2^{13} = 8192$

$86_{10} = 86$ base 10 $= 86 \to 8\times10^1 + 86\times10^0 = 86$

86 in base 2 $= 1\times2^6 + 0\times2^5 + 1\times2^4 + 0\times2^3 + 1\times2^2 + 1\times2^1 + 0\times2^0$

$= 64 + 0 + 16 + 0 + 4 + 2 + 0$

$= 86_{10} = 1010110_2$

Key rules for addition and subtraction:

1+0 = 1	1−0 = 1
1+1 = 10	10−1 = 1
1+1+1 = 11	11−1 = 10

$$\begin{array}{r} {}^{1} \\ 101 \\ +\,101 \\ \hline 1010 \end{array} \qquad \begin{array}{r} {}^{11} \\ 1011 \\ +\,1011 \\ \hline 10110 \end{array} \qquad \begin{array}{r} 111 \\ -\,10 \\ \hline 101 \end{array} \qquad \begin{array}{r} 101 \\ \times\,11 \\ \hline 101 \\ 1010 \\ \hline 1111 \end{array}$$

$$\begin{array}{r} 11 \\ 11\,\overline{)\,1011} \\ -11 \\ \hline 101 \\ -11 \\ \hline 10 \end{array} \leftarrow \text{Remainder}$$

Boolean Algebra + Axioms

Axioms

$0 \cdot 0 = 0$

$1 + 1 = 1$

$1 \cdot 1 = 1$

$0 + 0 = 0$

$0 \cdot 1 = 1 \cdot 0 = 0$

$1 + 0 = 0 + 1 = 1$

if $x = 0$, $\bar{x} = 1$

if $x = 1$, $\bar{x} = 0$

Single variable Theorems

$x \cdot 0 = 0$

$x + 1 = 1$

$x \cdot 1 = x$

$x + 0 = x$

$x \cdot x = x$

$x + x = x$

$x \cdot \bar{x} = 0$

$x + \bar{x} = 1$

$\bar{\bar{x}} = x$

Variable Properties

$x \cdot y = y \cdot x$, $x + y = y + x$ } commutative

$x \cdot (y \cdot z) = (x \cdot y) \cdot z$, $x + (y + z) = (x + y) + z$ } Associative

$x \cdot (y + z) = x \cdot y + x \cdot z$, $x + y \cdot z = (x + z)(x + y)$ } Distributive

$x + x \cdot y = x$, $x \cdot (x + y) = x$ } absorption

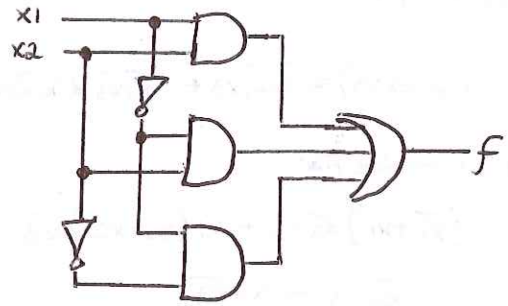
$x \cdot y + x \cdot \bar{y} = x$, $(x + y) \cdot (x + \bar{y}) = x$ } Combining

$\overline{x \cdot y} = \bar{x} + \bar{y}$, $\overline{x + y} = \bar{x} \cdot \bar{y}$ } De Morgan's Theorem

Proof of De Morgan's Law/Theorem

x	y	$x \cdot y$	$\overline{x \cdot y}$	$\bar{x}$	$\bar{y}$	$\bar{x} + \bar{y}$
0	0	0	1	1	1	1
0	1	0	1	1	0	1
1	0	0	1	0	1	1
1	1	1	0	0	0	0

Gate and Logic Synthesis



Some function f

x_1	x_2	$f(x_1, x_2)$
0	0	1
0	1	1
1	0	0
1	1	1

$\bar{x_1} \cdot \bar{x_2} = f$

$\bar{x_1} \cdot x_2 = f$

$\bar{x_1} \cdot x_2 = f$

$x_1 x_2 = f$

$\therefore f(x_1, x_2) = x_1 x_2 + \bar{x_1}\bar{x_2} + \bar{x_1} x_2$

not simplest form though!

Let's use some theorems for reduction

$$\therefore f(x_1, x_2) = x_1 x_2 + \bar{x_1}\bar{x_2} + \bar{x_1} x_2 + \bar{x_1} x_2$$

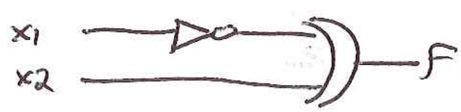
$$= x_1 x_2 + \bar{x_1} x_2 + \bar{x_1}\bar{x_2} + \bar{x_1} x_2$$

$$= (x_1 + \bar{x_1}) x_2 + \bar{x_1}(\bar{x_2} + x_2)$$

$$\boxed{= x_2 + \bar{x_1}}$$

$x_1 + \bar{x_1} = 1$

$x_2 + \bar{x_2} = 1$



much simpler and costs less!

From truth table we can synthesize expressions for rows of $f = 1$ or for rows of $f = 0$.

Minterm is a sum of products

Maxterm is a product of sums

MINTERMS + MAXTERMS

Row Number	x_1	x_2	x_3	Minterm	Maxterm
0	0	0	0	$m_0 = \bar{x_1}\bar{x_2}\bar{x_3}$	$M_0 = x_1 + x_2 + x_3$
1	0	0	1	$m_1 = \bar{x_1}\bar{x_2} x_3$	$M_1 = x_1 + x_2 + \bar{x_3}$
2	0	1	0	$m_2 = \bar{x_1} x_2 \bar{x_3}$	$M_2 = x_1 + \bar{x_2} + x_3$
3	0	1	1	$m_3 = \bar{x_1} x_2 x_3$	$M_3 = x_1 + \bar{x_2} + \bar{x_3}$
4	1	0	0	$m_4 = x_1 \bar{x_2}\bar{x_3}$	$M_4 = \bar{x_1} + x_2 + x_3$
5	1	0	1	$m_5 = x_1 \bar{x_2} x_3$	$M_5 = \bar{x_1} + x_2 + \bar{x_3}$
6	1	1	0	$m_6 = x_1 x_2 \bar{x_3}$	$M_6 = \bar{x_1} + \bar{x_2} + x_3$
7	1	1	1	$m_7 = x_1 x_2 x_4$	$M_7 = \bar{x_1} + \bar{x_2} + \bar{x_3}$

Row Number	x_1	x_2	x_3	$f(x_1,x_2,x_3)$
0	0	0	0	0
1	0	0	1	1
2	0	1	0	0
3	0	1	1	0
4	1	0	0	1
5	1	0	1	1
6	1	1	0	1
7	1	1	1	0

MSOP w/ minterm uses $f(x_1,x_2,x_3)$ with 1's

$$\therefore f(x_1,x_2,x_3) = \bar{x}_1\bar{x}_2x_3 + x_1\bar{x}_2\bar{x}_3 + x_1\bar{x}_2x_3 + x_1x_2\bar{x}_3$$

with reduction:

$$(\bar{x}_1 + x_1)\bar{x}_2x_3 + x_1(\bar{x}_2 + x_2)\bar{x}_3$$

$$\bar{x}_2x_3 + x_1\bar{x}_3$$

lower number of gates than originally thought

MPOS w/ maxterms uses $f(x_1x_2x_3)$ with 0's

$$f(x_1,x_2,x_3) = (x_1+x_2+x_3)(x_1+\bar{x}_2+x_3)(x_1+\bar{x}_2+\bar{x}_3)(\bar{x}_1+\bar{x}_2+\bar{x}_3)$$

w/ commutative + associative properties

$$f = ((x_1+x_3)+x_2)((x_1+x_3)+\bar{x}_2)(x_1+(\bar{x}_2+\bar{x}_3))(\bar{x}_1+(\bar{x}_2+\bar{x}_3))$$

$$= (x_1+x_3)(\bar{x}_2+\bar{x}_3)$$