CL=CSCI 160

CLASS 1

History

1854 - George Boole

He wrote:

'An Investigation of the Laws of thought' = mathematical methods to study the field of LOGIC.

Algebra of Propositions - Values:

True False

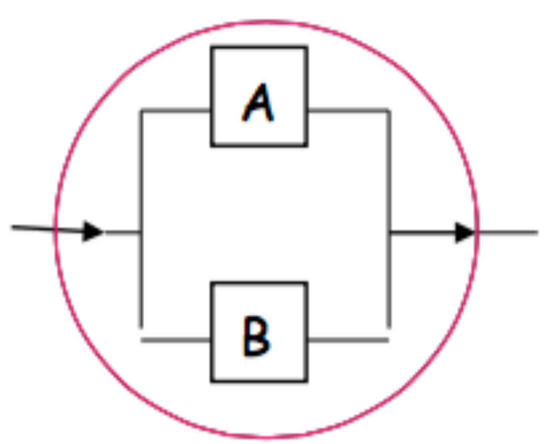
Operations:	not,	or,	and
Symbols:	$\overline{\ }$, '	$+$	$\cdot$
	$\neg$	V	$\wedge$
	$\sim$	|	&, *

Charles Pierce: XIX – made the connection with ELECTRICITY.

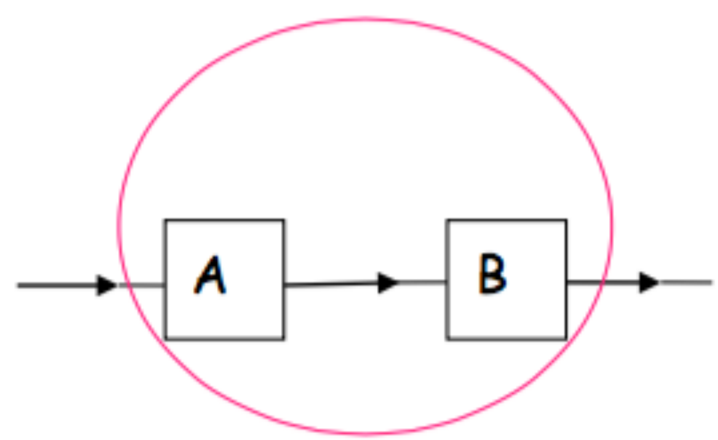
A Boolean Variable = Black Box in a circuit.

Two values → Conducts / Does not conduct } Electricity

A or B



A and B



1937 - Claude E. Shannon

In his Master's thesis at MIT he devised ways of using symbolic logic to improve electrical switching circuits.

He put the theoretical basis for the entire set of operations that would be used to design the electronic digital computers.

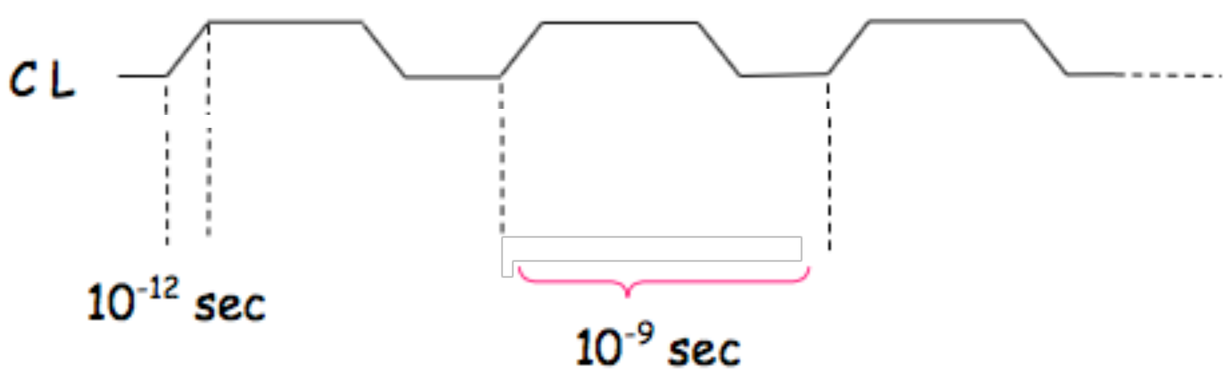
This course is about: **Hardware**

Key word: integrated circuit (packages); need to know:

1. How to design them.
2. How to combine them.

2-State Device

C L = clock



10^{-9} sec = 1 nanosec

10^{-12} sec=1 picosec

10^{-15} = femto

10^{-18} = atto

10^{12} = terra
10^{9} = giga

Binary numbers

Basis, or radix will be denoted by r.

Most used: r = 2 (binary), and r = 16, or any power of 2.

Conversions

Base 10 to base 2 Integers:

$$379_{10} = 2^8 + 2^6 + 2^5 + 2^4 + 2^3 + 2^1 + 2^0$$

$$= 1\,0\,1\,1\,1\,1\,0\,1\,1_2$$

$$\begin{array}{r} 379 \\ -256 \\ \hline 123 \\ 64 \\ \hline 59 \\ 32 \\ \hline 27 \\ 16 \\ \hline 11 \\ 8 \\ \hline 3 \end{array}$$

Remainder Method (integers):

Divide by 2 and take remainders.

379	1
189	1
94	0
47	1
23	1
11	1
5	1
2	0
1	1

Read binary number $\boxed{\text{from bottom up.}}$

We obtain the same binary number as above: $1\,0\,1\,1\,1\,1\,0\,1\,1_2$

Fractions: Use powers of 2 with **negative** exponents.

NOTE:

$2^{-1} = 1/2 = .5$

$2^{-2} = 1/\ 2^{2} = .25$

$2^{-3} = .125$

$2^{-4} = .0625$

First method of conversion:

$0.379_{10} = 2^{-2} + 2^{-3}$

.25
.129
.125
.004

4 bit prec

$= .0110_{2}$

1-4 Remainder Method for Integers - Justification

Why does the remainder method work? Let's take part of the number considered earlier:

$59_{10} = 1\,1\,1\,0\,1\,1_2 = 2^5 + 2^4 + 2^3 + 2^1 + 2^0$

$= 2(2^4 + 2^3 + 2^2 + 2^0) + 1$

We divide by 2 and record quotient and remainder:

Repeat until quotient = 0

$29 = 2(2^3 + 2^2 + 2^1) + 1$

$14 = 2(2^2 + 2^1 + 2^0) + 0$

$7 = 2(2^1 + 2^0) + 1$

$3 = 2(2^0) + 1$

$1 = 1$

59	1
29	1
14	0
7	1
3	1
1	1
0	

HW 1-1: Find a method similar to the remainder method for the integer numbers that applies to fractional numbers.
(As in converting $.379_{10} = .???_2$)

Review: Arithmetic operations in bases 10 and 2.

Hexadecimal digits:

Hexadecimals (base 16)

0 - 0

...

9 - 9

10 - A

11 - B

12 - C

13 - D

14 - E

15 - F

We have: $16 = 2^4$, i.e. 16 is a power of 2, and $16^2 = (2^4)^2 = 2^8$

To convert from base 16 to base 10 we simply do: $324_{16} = 3*16^2 + 2*16 + 4$

The easy part is converting hex to and from binary.

Ex: Convert **binary** number 1 0 1 1 0 0 1 1 0 1. 1 0 1 1 0 1 **to hex**

$0\,0\,1\,0\;1\,1\,0\,0\;1\,1\,0\,1.\,1\,0\,1\,1\;0\,1\,0\,0_2 = 2CD.B4_{hex}$

Hex to binary: $FAD.DAD_{hex} = 1111\,1010\,1101.1101\,1010\,1101_2$

Addition / Subtraction

HW 1-2: Consider the two decimal numbers:

$M = 3892.74_{10}$

$N = 9341.65_{10}$

Convert them to bases 2 and 16, and then add & subtract them in those bases.

HW 1-A

Which is the largest binary number that can be expressed with 15 bits? What are the equivalent decimal and hexadecimal numbers?

HW 1-B

Consider a system that contains 32K bytes. Assume we are using byte addressing, that is assume that each byte will need to have its own address, and therefore we will need 32K different addresses. For convenience, all addresses will have the same number ***n***, of bits, and ***n*** should be as small as possible.

What is the value of ***n*** ?

HW 1-C

The numbers in each of the following equalities are all expressed in the same base, ***r***. Determine this radix ***r*** in each case for the following operations to be correct.

(a) 14/2 = 5
(b) 54/4 = 13

Examples

Addition:

Binary:

$$\begin{array}{r} {}^{1\;\;1\,1\;\;1\;\;} \\ 1\,1\;1\,1\;\;+ \\ 1\,0\;0\,1\;\; \\ \hline 1\,1\;0\,0\,0\;\; \end{array}$$

Subtraction:

Decimal:

$$\begin{array}{r} {}^{1\;\;} \\ 2\,4\;\;- \\ 9\;\; \\ \hline 1\,5\;\; \end{array}$$

Attention: No crossing of digits while subtracting

Binary:

$$\begin{array}{r} {}^{1\;\;1\;\;\;\;} \\ 1\;0\;0\;\;- \\ 1\;1\;\; \\ \hline 0\;1\;\; \end{array}$$