Computing via boolean logic.

COS 116: 3/8/2011

Sanjeev Arora

Recap: Boolean Logic Example

Ed goes to the party if
Dan does not and Stella does.

Choose "Boolean variables" for 3 events:

E: Ed goes to party
D: Dan goes to party
S: Stella goes to party

Each is either TRUE or FALSE

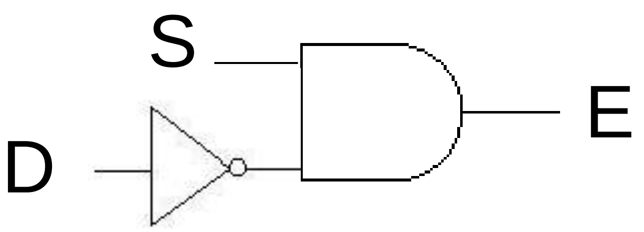
E = S **AND** (**NOT** D)

Alternately: E = S **AND** $\overline{\mathrm{D}}$

Three Equivalent Representations

Boolean Expression $E = S \text{ AND } \overline{D}$

Boolean Circuit



Truth table:
Value of E for every possible D, S.
TRUE=1; FALSE= 0.

D	S	E
0	0	0
0	1	1
1	0	0
1	1	0

Boolean "algebra"

A **AND** B written as $A \cdot B$ A **OR** B written as $A + B$

$0 \cdot 0 = 0$

$0 \cdot 1 = 0$

$1 \cdot 1 = 1$

$0 + 0 = 0$

$1 + 0 = 1$

$1 + 1 = 1$

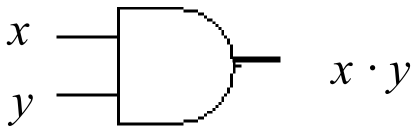
Funny arithmetic

Will provide readings on this…

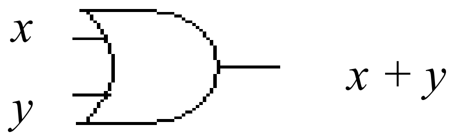
Boolean gates

Shannon (1939)

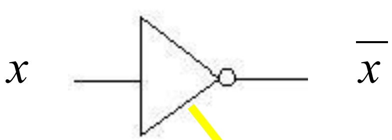
High voltage = 1
Low voltage = 0



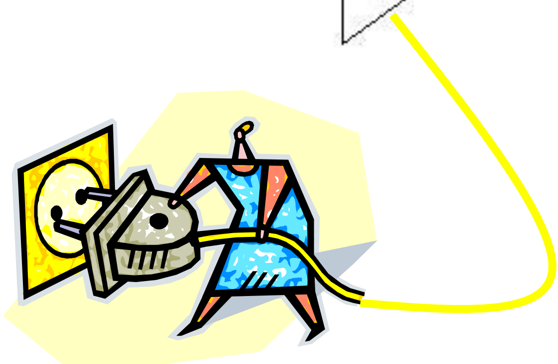
Output voltage is high
if **both** of the input voltages are high;
otherwise output voltage low.



Output voltage is high
if **either** of the input voltages are high;
otherwise output voltage low.



Output voltage is high
if the input voltage is low;
otherwise output voltage low.



(implicit extra wires for power)

Claude Shannon (1916-2001)

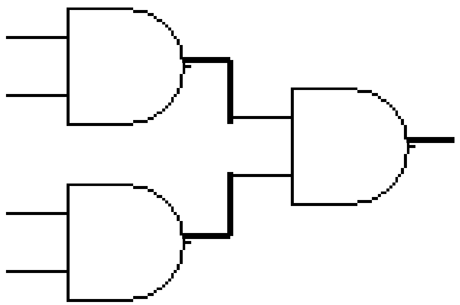
Founder of many fields
(circuits, information theory, artificial intelligence…)

A Symbolic Analysis of Relay and Switching Circuits, [1938]

With “Theseus” mouse

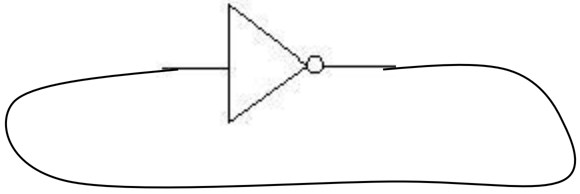
Combinational circuit

- Boolean gates connected by wires



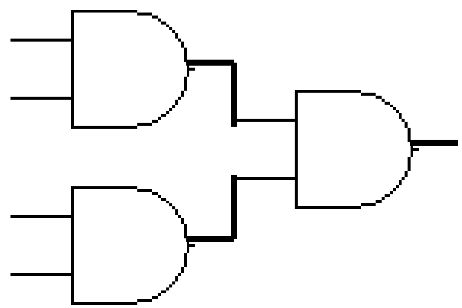
Wires: transmit voltage
(and hence value)

- Important: no cycles allowed

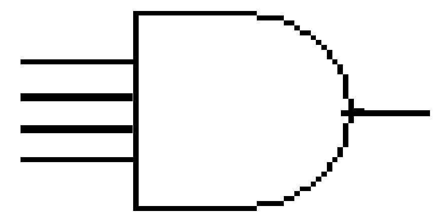


Examples

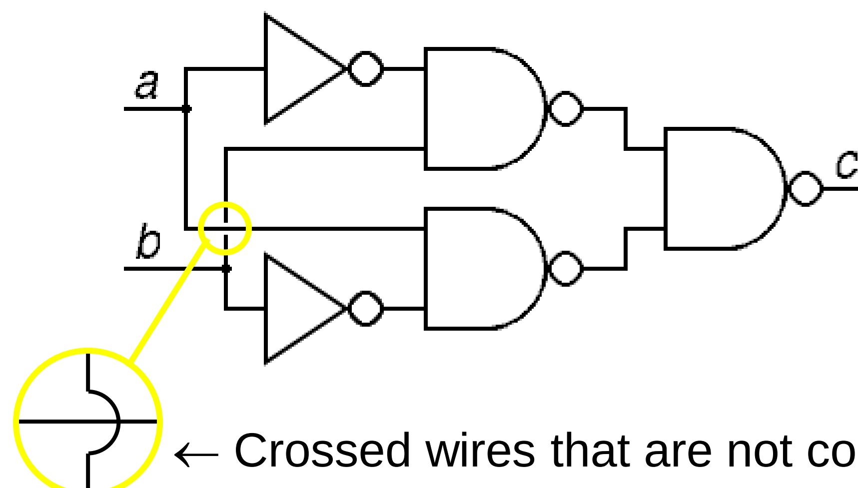
4-way AND



(Sometimes we use this for shorthand)



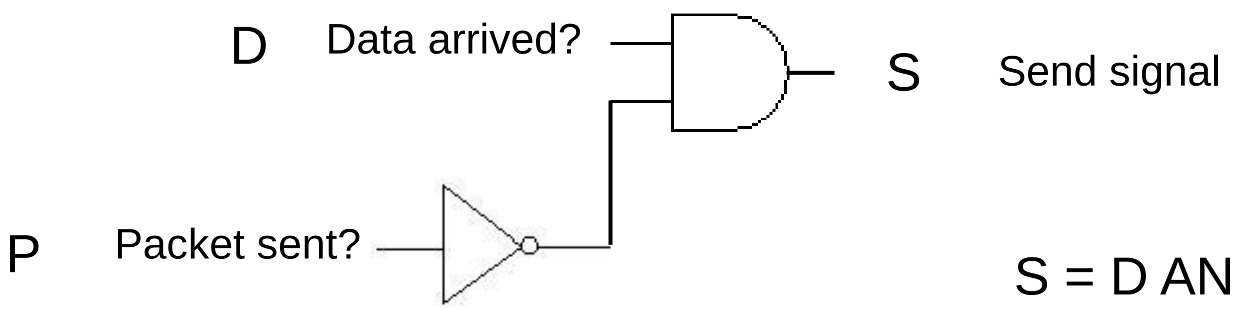
More complicated example



← Crossed wires that are not connected are sometimes drawn like this.

Combinational circuits and control

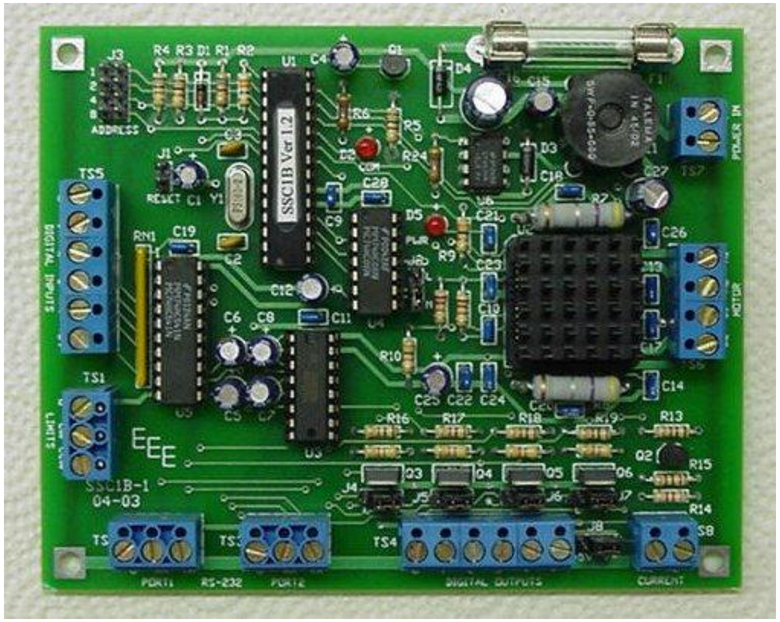
- "If data has arrived and packet has not been sent, send a signal"



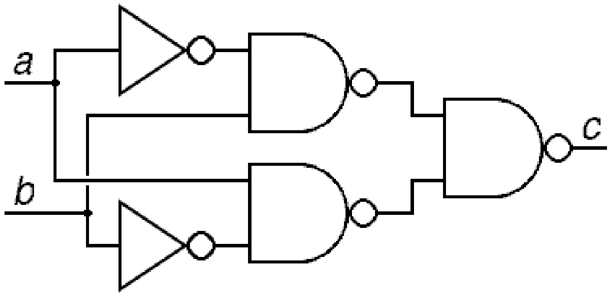
S = D AND (NOT P)

Circuits compute functions

- Every combinational circuit computes a Boolean function of its inputs



Inputs



Outputs

Ben Revisited

Ben only rides to class if he overslept, but even then if it is raining he'll walk and show up late (he hates to bike in the rain). But if there's an exam that day he'll bike if he overslept, even in the rain.

B: Ben Bikes

R: It is raining

E: There is an exam today

O: Ben overslept

How to write a boolean expression for B in terms of R, E, O?

Truth table → Boolean expression

Use **OR** of all input combinations that lead to TRUE

$$B = O\cdot\overline{R}\cdot\overline{E} + O\cdot\overline{R}\cdot E + O\cdot R\cdot E$$

O	R	E	B
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	1

Note:
AND, OR, and NOT gates suffice to implement every Boolean function!

Expression simplification

- Some simple rules:

$x + \overline{x} = 1$

$x \cdot 1 = x$

$x \cdot 0 = 0$

$x + 0 = x$

$x + 1 = 1$

$x + x = x \cdot x = x$

$x \cdot (y + z) = x \cdot y + x \cdot z$

$x + (y \cdot z) = (x+y) \cdot (x+z)$

$x \cdot y + x \cdot \overline{y}$

$= x \cdot (y + \overline{y})$

$= x \cdot 1$

$= x$

De Morgan's Laws:

$\overline{x \cdot y} = \overline{x} + \overline{y}$

$\overline{x + y} = \overline{x} \cdot \overline{y}$

In groups of three try to simplify: $O \cdot \overline{R} \cdot \overline{E} + O \cdot \overline{R} \cdot E + O \cdot R \cdot E$

Simplifying Ben's circuit

- $B = O \cdot \overline{R} \cdot \overline{E} + O \cdot \overline{R} \cdot E + O \cdot R \cdot E$

$$= O \cdot (\bar{R} \cdot \bar{E} + \bar{R} \cdot E + R \cdot E)$$

$$= O \cdot (\bar{R} \cdot (\bar{E} + E) + R \cdot E)$$

$$= O \cdot (\bar{R} + R \cdot E)$$

…

$$= O \cdot (\bar{R} + E)$$

Boole's reworking of Clarke's "proof" of existence of God (see handout)

- General idea: Try to prove that Boolean expressions $E_1, E_2, \ldots, E_k$ cannot simultaneously be true

- <u>Method</u>: Show $E_1 \cdot E_2 \cdot \ldots \cdot E_k = 0$

- What exactly does Clarke's "proof" prove? How convincing is such a proof to you?

Also: Do Google search for "Proof of God's Existence."

The Kalam argument for god's existence (arose in many world traditions)

- Whatever that begins to exist has a cause.
- The universe began to exist. If there is no original cause (i.e., God) then there must be an infinite chain of causal events, which is impossible.

Does this remind you of other issues studied in the course?

Sizes of representations

- For *k* variables:

k	10	20	30
2^k	1024	1048576	1073741824

For an arbitrary function,
expect roughly half of X's to be 1
(for 30 inputs roughly 1/2 billion!)

Tools for reducing size:
(a) circuit optimization (b) modular design

A	B	…	X
0	0	…	0
0	0	…	0
0	1	…	0
0	1	…	1
…	…	…	…
…	…	…	…
1	1	…	1

2^k

$k + 1$

Combinational circuit for binary addition?

25	11001
+29	11101
54	110110

- Want to design a circuit to add any two *N*-bit integers (say N =64).

Is the truth table method useful? Ideas?

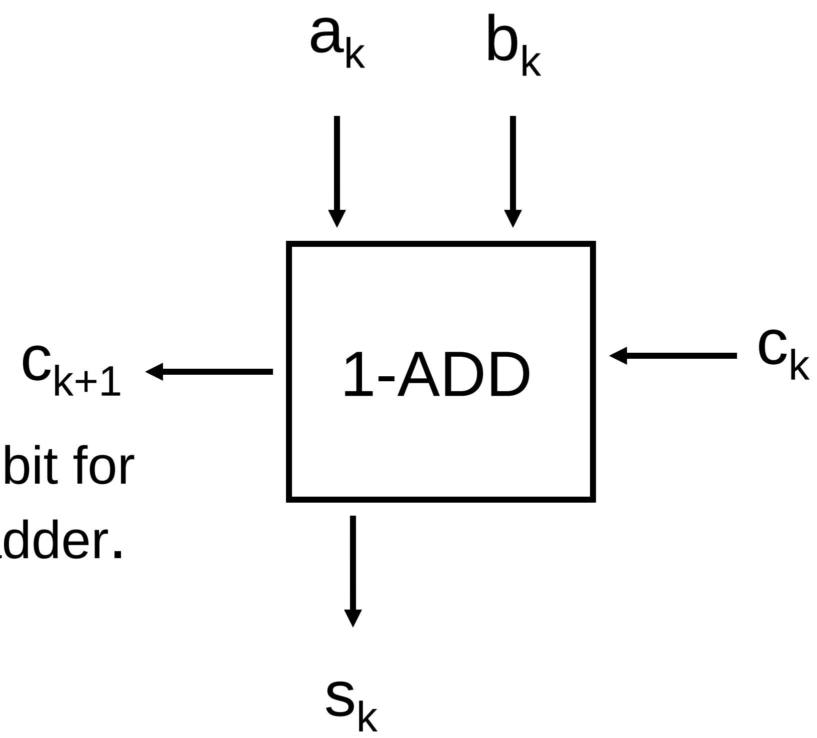
Modular design

Have small number
of basic components.

Put them together to achieve
desired functionality

Basic principle of modern industrial design;
recurring theme in next few lectures.

1-bit adder



(Carry from previous adder)

Carry bit for next adder.

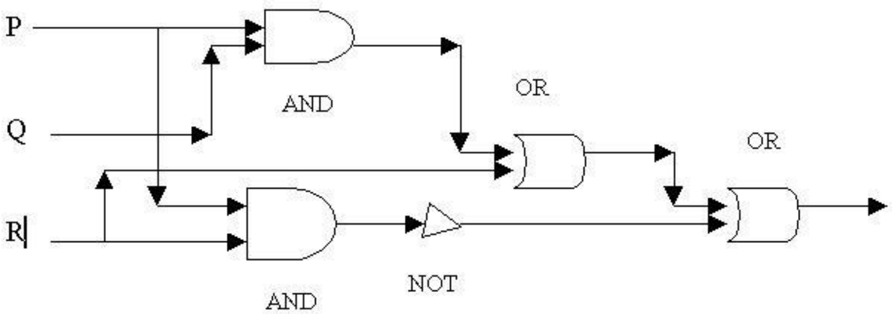
Hand in on Mar 22: Truth table, circuit for 1-bit adder.

Modular Design for boolean circuits

An N-bit adder using N 1-bit adders
(will do Mar 22)

Something to think about:
How hard is Circuit Verification?

- Given a circuit, decide if it is “trivial” (no matter the input, it either always outputs 1 or always outputs 0)



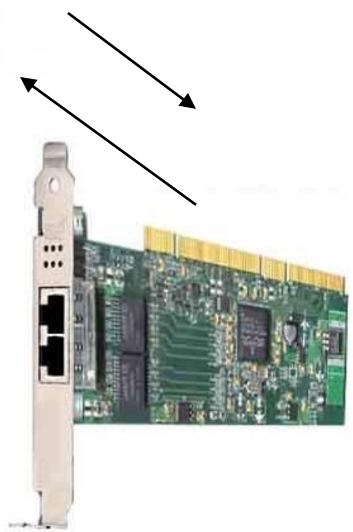
- Alternative statement: Decide if there is any setting of the inputs that makes the circuit evaluate to 1.

Time required?

Beyond combinational circuits …

- Need 2-way communication (must allow cycles!)

CPU



Ethernet card

- Need memory (scratchpad)

Will study next time.

(1) A waveform is sampled at a rate of 4 Hz and 4-bit samples. The following is the sequence of samples.
Draw the waveform (hand it in).
Is your answer unique? If not,
Draw another waveform consistent with the samples.

10, 11, 11, 11, 5, 0, 8, 11, 11, 11, 15.

(2) Suppose variable i has value 4 and j has the value 8. What values do they have after the following instruction is executed: $i \leftarrow j/i$?

(3)Your computer runs at 3Ghz. How may operations per sec (roughly) does it do?