Binary Numbers

Part 7c of
“Electronics and Telecommunications”
A Fairfield University E-Course
Powered by LearnLinc

Module: Digital Electronics
(in two parts)

- Text: "Digital Logic Tutorial ," Ken Bigelow, http://www.play-hookey.com/digital/
- References:
  - "Electronics Tutorial", part 10 (Thanks to Alex Pounds) http://doctord.dyndns.org:8000/courses/Topics/Electronics/Alex_Pounds/Index.htm
- Contents:
  - 7 – Digital Electronics 1
    - 5 on-line sessions plus one lab and a quiz
  - 8 – Digital Electronics 2
    - 5 on-line sessions plus one lab and a quiz
- Mastery Test part 4 follows this Module

Section 7: Digital Electronics 1

- Logic gates and Boolean algebra
- Truth Tables
- Binary numbers
- Memory
- Flip-Flops

Section 8: Digital Electronics 2

- Clocks and Counters
- Shift Registers
- Decoders
- Multiplexers & Demultiplexers
- Sampling

- **MT4**

Section 7 Schedule

Session 7a	03/05	Introduction: Binary, Logic Gates and Boolean	Alex Pounds: Part 10 "Ken B": Home, Basic Gates, & Boolean Algebra
Session 7b	03/10	Logic Gates and Truth Tables	Alex Pounds: Part 10 "Ken B": Derived Gates, Xor
Session 7c	**03/12**	**Binary numbers**	**"Keb B": Binary Addition** **"Vinay ": Binary Numbers**
Session 7d	03/17	Memory: Registers, RAM & ROM	"Ken B": RS Nand Latch, Clocked RS Latch, D Latch
Session 7e (Lab - 03/22, Sat.)	03/19	Pulses, Clocks and Flip-Flops	"Ken B": RS Flip-Flop, JK Flip-Flop, D Flip-Flop, Flip-Flop Symbols
Session 7f (Quiz 7 due 03/30)	03/24	Review for Quiz 7	
Session 7g	03/31	Quiz Results	

Review

- Binary
  - 1, “True”, “On”, “High” (5 volts in electronics)
  - 0, “False”, “Off”, “Low” (0 volts in electronics)
- Basic Logic Gates
  - AND, OR, NOT
- Derived Logic Gates
  - NAND, NOR, XOR
- Truth Tables:
  - Enumerate outputs for all input combinations
- Boolean Algebra
  - Named Variables: True or False
  - Expressions: Equations describing relationships

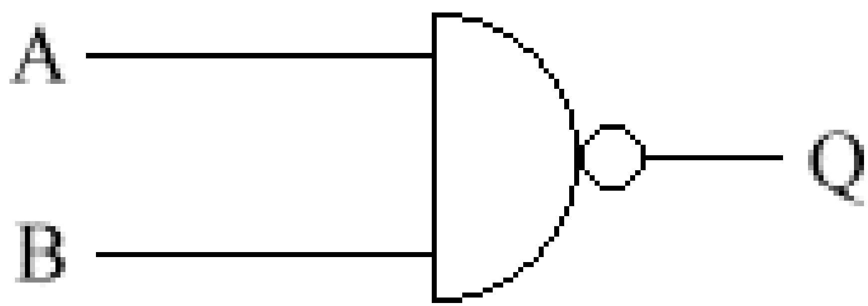
Derived Logic Gates

- Derived gates are those made out of simple combinations of the basic gates.
- Common derived functions
  – NAND: inverted AND
  – NOR: inverted OR
  – XOR: the exclusive or A or B but not A and B
- These derived gates are the ones seen most often.

NAND Gate

- Q is False when both A AND B are True and True otherwise
  - $Q = \overline{(A*B)} = (A*B)'$
  - It can have any number of inputs
  - Note that this is an AND followed by a NOT

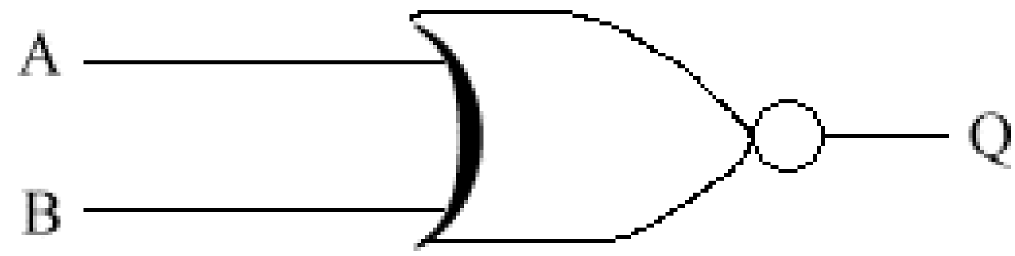
A	B	Q
0	0	1
0	1	1
1	0	1
1	1	0



NOR Gate

- Q is False when either A or B is True
  - $Q = \overline{(A+B)} = (A+B)'$
  - It can have any number of inputs
  - Note that this is an OR followed by a NOT

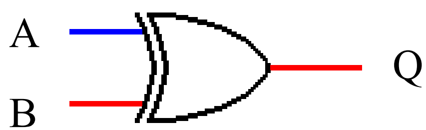
A	B	Q
0	0	1
0	1	0
1	0	0
1	1	0

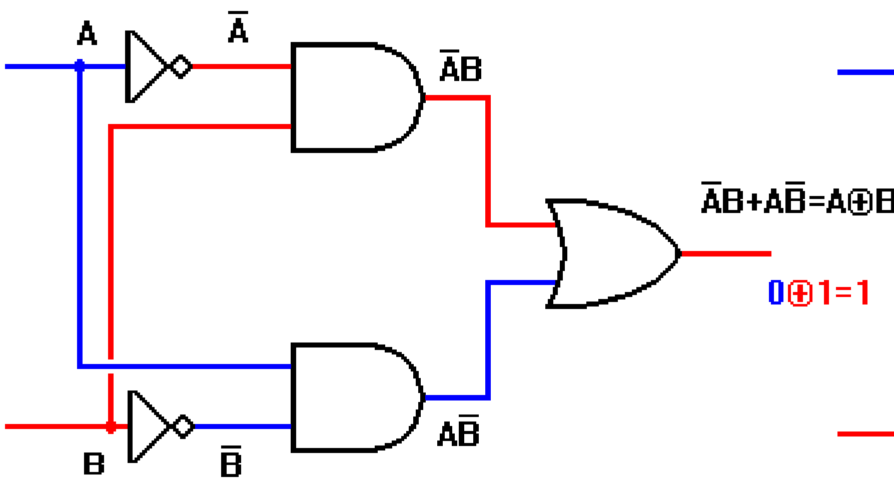


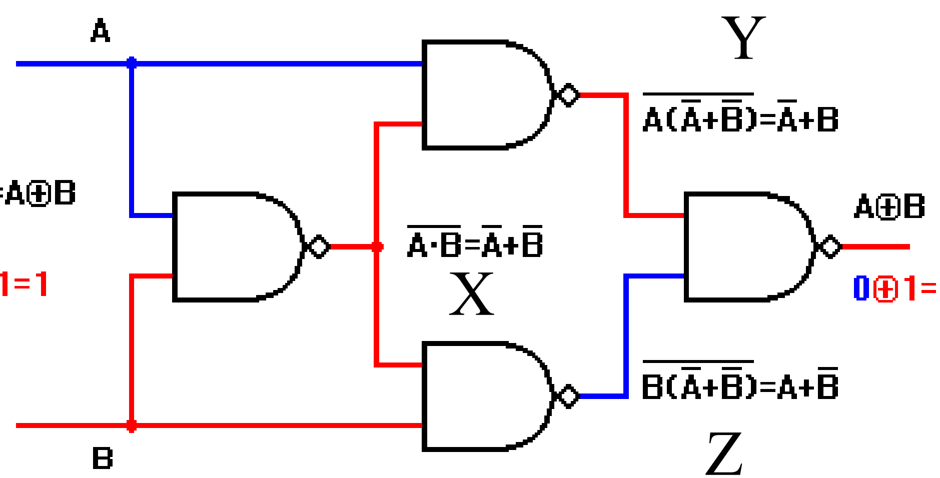
XOR: The Exclusive OR

- Q is True when either A OR B is True, but not when both A AND B are True
  - $Q = \bar{A}*B + A*\bar{B} = A \oplus B$

A	B	X	Y	Z	Q
0	0	1	1	1	0
0	1	1	1	0	1
1	0	1	0	1	1
1	1	0	1	1	0







Number Systems

- Decimal Numbers (we have 10 fingers)
  - $2705 = 2*10^3 + 7*10^2 + 0*10^1 + 5*10^0$
  - Zero is a place holder (an Arab invention)
  - Replaced Roman Numerals (MCMXVIII=1943)
- Binary Numbers
  - Based on powers of 2 (the “base” or “radix”)
  - $1010 = 1*2^3 + 0*2^2 + 1*2^1 + 0*2^0 = 10$ decimal
  - k bits can count up to $2^k – 1$ (2^k values including zero)
    - 8-bits ⇨ 256 values, 16-bits ⇨ 65536 values (64k binary)
    - 10-bits ⇨ 1024 values (1k binary)
    - 20-bits ⇨ 1,048,576 values (1 meg binary)
  - Well suited for our 2-valued digital logic (computers)

Definitions

- Bit: the unit of information
- Nibble: 4-bits
- Byte: 8-bits
- Word: the unit of storage in a computer (32 bits in a Pentium)

Negative Numbers

- The first (leftmost, “most significant”) bit is the “SIGN” bit
  - 0 means positive (half of the values)
  - 1 means negative (half of the values)
- “Sign-Magnitude” numbers (not used often)
  - Remaining bits give the magnitude
    0110 = +3, 1110 = -3
- Two’s complement negative numbers
  - First complement all bits (One’s complement)
  - Add one
     5 = 0101, its one’s complement is 1010 so
    -5 = 1011 in two’s complement

Adding Binary Numbers

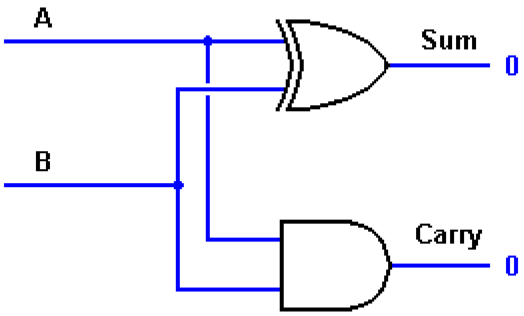
- Let's do an example:
  17 = 00010001 (eight bits)
  11 = 00001011
  28 = 00011100 (watch out for "carries")
  16 + 8 + 4
- Another example
  17 = 00010001
  -5 = 11111011 (two's complement again)
  12 = 00001100 (the "overflow" is ignored)
  8 + 4
- Note that subtraction is done by adding the twos complement of the "subtrahend"

Binary Number Simulation

- We'll go to http://vwop.port5.com/beginner/bhextut.html (Vinay's site) to see binary numbers in action

Half Adder using Logic Gates

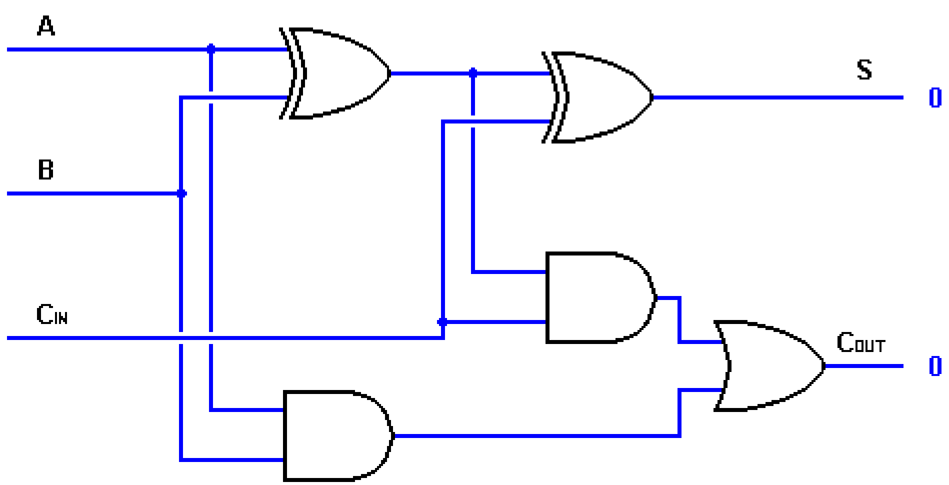
- An XOR gives us the result of adding two bits
- The AND gate gives us the "carry"
- What about the carry from the next lower result?



A	B	S	C
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

A Full Adder

- The “Full Adder” can do the job.



A	B	Cin	S	Cout
0	0	0	0	0
0	1	0	1	0
1	0	0	1	0
1	1	0	0	1
0	0	1	1	0
0	1	1	0	1
1	0	1	0	1
1	1	1	1	1

Adder Simulation

- We'll again go to www.play-hookey.com/digital to see adders in action

Section 7 Schedule

Session 7a	03/05	Introduction: Binary, Logic Gates and Boolean	"Hookey": Home, Basic Gates, & Boolean Algebra Alex Pounds: Part 10
Session 7b	03/10	Logic Gates and Truth Tables	"Hookey": Derived Gates, Xor
Session 7c	03/12	Binary numbers	"Hookey": Binary Addition "Vinay ": Binary Numbers
Session 7d	**03/17**	**Memory: Registers, RAM & ROM**	**"Hookey": RS Nand Latch, Clocked RS Latch, D Latch, Notes**
Session 7e (Lab - 03/22, Sat.)	03/19	Pulses, Clocks and Flip-Flops	"Hookey": RS Flip-Flop, JK Flip-Flop, D Flip-Flop, Flip-Flop Symbols,
Session 7f (Quiz 7 due 03/30)	03/24	Review for Quiz 7	
Session 7g	03/31	Quiz Results	