CS271 WEEK 9 LECTURES

Procedures
- may have parameters, calling mechanism present, return method or mechanism also present
- Control passed to proc

Macros
- Macros are not called using call
- may have parameters
- Entire Macro code is subbed into where the invocation is.
- Similar to constant ↑
- Must be defined BEFORE invocation
- Declared: macroname MACRO [p1, p2...]
  statements (opcodes)
  ENDM
  (optional to pass params)
- Invoked as the macro name.

Macros and Labels
- MASM requires that you use labels like this:
  LOCAL label as opposed to label

Macros
- execute faster than procedures as they are in-line.
- Calling a macro too many times results in FAT code... Alot of extra code.

Recursion
- requires base case and recursive part
  - Outline
    - Get vars out of stack
    - Check base case
    - Push vars to stack
    - Call function
    - Pop baseptr ← Quit
      and ret n | n = var_size * vars

Stack Overflow
- too much recursion that messes up (overflows) the stack.

Digital Logic
- Discrete circuitry where only values are 1, 0

Logical Operators
- AND

F	F	F
F	T	F
T	F	F
T	T	T

- OR

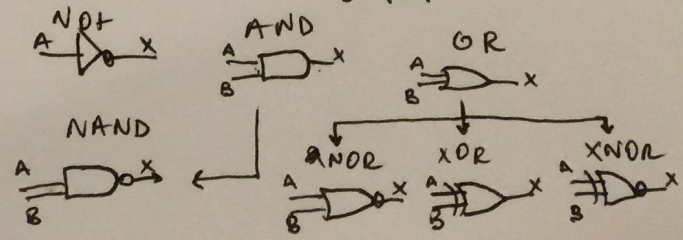
F	F	F
F	T	T
T	F	T
T	T	T

- NOT

F	T
T	F

Boolean Operators
- NOT A: $\overline{A}$
- A AND B: AB
- A OR B: $A+B$
- A XOR B: $A \oplus B$
- A NAND B: $\overline{AB}$
- A NOR B: $\overline{A+B}$
- A XNOR B: $\overline{A \oplus B}$

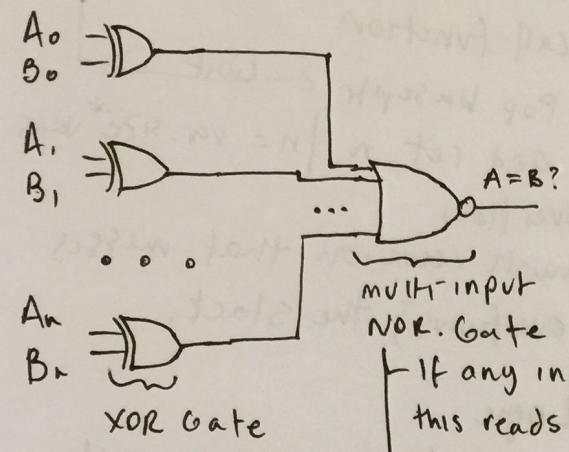
Internal Circuitry
- Gates (ON: 1: 5v ; OFF: 0: 0.5v)
- Combinations of gates are circuits
- There are Gates for



Boolean functions

- for a function with n- inputs, there are 2^n outputs

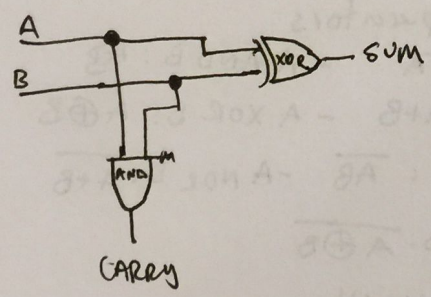
n- bit comparator



Adder (s)

- Addition involves carry bits and sum bits.

HALF ADDER



FULL ADDER