Combinational Logic (mostly review!)

- Logic functions, truth tables, and switches
  - NOT, AND, OR, NAND, NOR, XOR, . . .
  - Minimal set
- Axioms and theorems of Boolean algebra
  - Proofs by re-writing
  - Proofs by perfect induction
- Gate logic
  - Networks of Boolean functions
  - Time behavior
- Canonical forms
  - Two-level
  - Incompletely specified functions
- Two-level minimization

Possible Logic Functions of Two Variables

- 16 possible functions of 2 input variables:
  - 2**(2**n) functions of n inputs

X	Y	16 possible functions (F0–F15)															
0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
0	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
1	0	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
1	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1

0 | X and Y | X | Y | X xor Y | X or Y | X nor Y
not (X or Y) | X = Y | not Y | not X | X nand Y
not (X and Y) | 1

Cost of Different Logic Functions

- Some are easier, others harder, to implement
  - Each has a cost associated with the number of switches needed
  - 0 (F0) and 1 (F15): require 0 switches, directly connect output to low/high
  - X (F3) and Y (F5): require 0 switches, output is one of inputs
  - X' (F12) and Y' (F10): require 2 switches for "inverter" or NOT-gate
  - X nor Y (F4) and X nand Y (F14): require 4 switches
  - X or Y (F7) and X and Y (F1): require 6 switches
  - X = Y (F9) and X ⊕ Y (F6): require 16 switches

  - Because NOT, NOR, and NAND are the cheapest they are the functions we implement the most in practice