Introduction

- What is contemporary logic design?
- What will we do in CSE370?
- Class administration and logistics

© 1992 Gaetano Borriello, 1994 Steven Burns

What is contemporary logic design?

- What is design?
  - – given a specification, finding a structure that implements the requirements of the specification with some criteria such as size, cost, beauty, etc.
- What is logic design?
  - – determining the collection of digital logic components to perform a specified function and physically interconnecting them to realize the specification
- What is contemporary logic design?
  - – accomplishing the logic design with the aid of computer-aided design tools and constructing an implementation with programmable logic devices after extensive simulation

© 1992 Gaetano Borriello, 1994 Steven Burns

Motivation

- Dramatic change in the way industry does hardware design
  - larger and larger designs
  - shorter and shorter time to market
  - cheaper and cheaper products
- Pervasive use of computer-aided design tools
  - deemphasis on hand design methods (don't scale well)
  - emphasis on abstract design representation (maintainability)
  - many levels of design representation (refinement)
  - hardware design begins to look more like software design
- Emergence of rapid implementation circuit technology
  - programmable rather than discrete logic (less parts and more integration)
  - use of simulation to debug designs (before costly hardware prototyping)
- Importance of sound design methodologies
  - rules of composition (easy to compose pieces)
  - fixed design styles (rather than ad hoc)

© 1992 Gaetano Borriello, 1994 Steven Burns

What will we do in CSE370?

- Lectures
  - concepts required for logic design, starting from simple circuits
  - and working our way up
  - will follow the text rather closely with more emphasis on use of
  - programmable logic and CAD techniques
- Written assignments and reading
  - reinforce the concepts learned in lecture by exercising your
  - understanding via drill and word problems
  - reading the text for another point of view
- Laboratory assignments
  - reinforce the concepts learned in lecture by simulating designs with
  - the help of state-of-the-art CAD tools
- Hardware laboratory assignments of the old CSE370 have moved
  - to CSE467

© 1992 Gaetano Borriello, 1994 Steven Burns

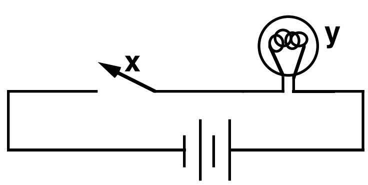
Computation: Abstract vs. Implementation

- Up to now, computation has been a mental exercise (paper, programs)
- This class is about physically implementing computation using electrons
- Basic units of computation are:
  - decision: if x then do y
  - assignment: x := y
  - data operations: x + y – 5
  - control: program counter, jump
  - we will study how each of these are implemented in hardware
  - and composed into computational structures

© 1992 Gaetano Borriello, 1994 Steven Burns

Switches as the basic element

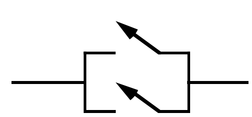
- Implementing a simple decision:



close switch (if x is asserted)
and turn on light bulb (y)

- Compose decisions into more complex ones (Boolean functions):

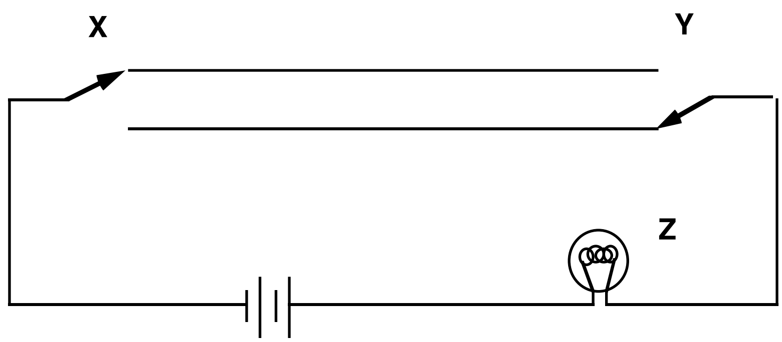
AND 

OR 

© 1992 Gaetano Borriello, 1994 Steven Burns

Constructing Logic Functions from Switches

- **EXCLUSIVE OR (XOR)**
  - **If switch X or switch Y but not both are asserted, lamp Z will light.**



Switches are drawn in the unasserted state.

© 1992 Gaetano Borriello, 1994 Steven Burns

Switching Networks

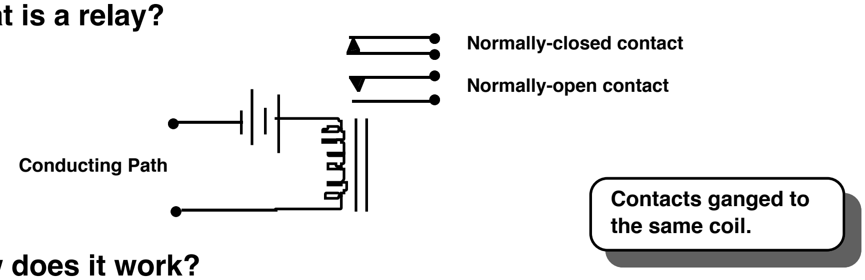
- **Switch settings determine whether or not a conducting path exists between two circuit nodes**
- **Inorder to use these networks effectively, there must be some way to use conducting paths (outputs of a network) to set other switches (inputs to a network).**
- **Now we can connect together switching networks to construct larger switching networks, i.e., there is a way to connect outputs of one network to the inputs of the next.**

© 1992 Gaetano Borriello, 1994 Steven Burns

Relay Networks

- A simple way to convert between conducting paths and switch settings is to use (electo-mechanical) relays.

- What is a relay?



- How does it work?
  - Current flowing through the coil magnetizes the iron core, closing the
  - n.o. contacts and opening the n.c. contacts. When no current flows, a
  - spring in the contacts returns them to their normal positions.

- What determines the switching speed of a relay network?

© 1992 Gaetano Borriello, 1994 Steven Burns

CMOS Transistor Networks

- Relays aren't used much anymore. (Some traffic light controllers are still electro-mechanical.)

- Modern digital systems are designed in a technology called CMOS (Complementary MOS) where MOS stands for Metal-Oxide on Semiconductor.

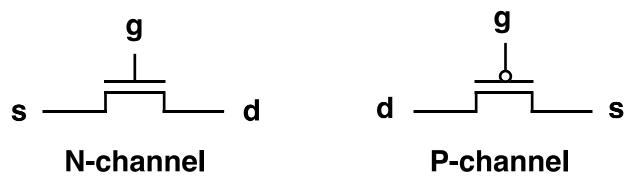
- Similar, though easier to work with than relays. MOS transistors act as voltage-controlled switches.

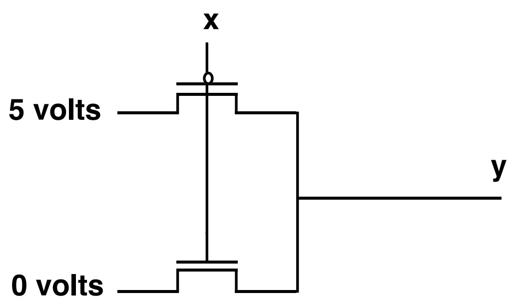
- MOS transistors have three terminals: drain, gate, and source. They act as switches in the following way. If the voltage on the gate terminal is (some amount) higher than the source terminal, a conducting path will be established between the drain and source.

- MOS transistors come in two favors: n-channel (that described above) and p-channel, where the conducting path is established if the gate terminal is at a voltage (some amount) lower than both the source terminal.

© 1992 Gaetano Borriello, 1994 Steven Burns

CMOS Networks



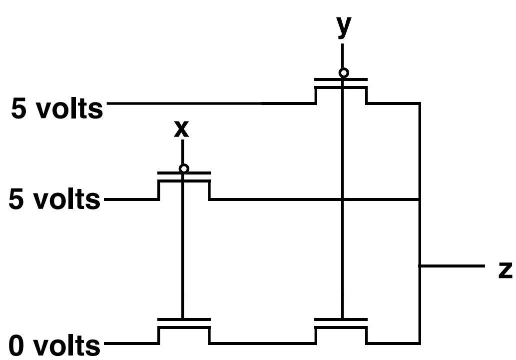


What is the relationship between x and y?

x	y
0 volts	
5 volts	

© 1992 Gaetano Borriello, 1994 Steven Burns

Two Input Networks



What is the relationship between x, y and z?

x	y	z
0 volts	0 volts	
0 volts	5 volts	
5 volts	0 volts	
5 volts	5 volts	

© 1992 Gaetano Borriello, 1994 Steven Burns

Speed of CMOS Networks

- What influences the speed of CMOS networks?

© 1992 Gaetano Borriello, 1994 Steven Burns

Representations of Digital Designs

- Physical devices (transistors, relays)
- Switches
- Truth tables
- Boolean algebra
- Gates
- Waveforms
- Finite state behavior
- Concurrent abstract specifications

scope of CSE370

© 1992 Gaetano Borriello, 1994 Steven Burns

Digital vs. Analog

- We'll find it convenient to think of digital systems as having only discrete, digital, output values.
- In reality, real electronic components will exhibit continuous, analog, behavior.
- Why do we make this abstraction?
    -
    -
- Why does it work?
    -
    -

© 1992 Gaetano Borriello, 1994 Steven Burns

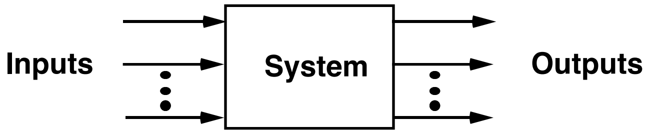
Mapping from Physical World to Binary World

Technology	State 0	State 1
Relay logic	Circuit Open	Circuit Closed
CMOS logic	0.0-1.5 volts	3.5-5.0 volts
Transistor Transistor Logic (TTL)	0.0-0.8 volts	2.0-5.0 volts
Fiber Optics	Light off	Light on
Dynamic Memory	Discharged capacitor	Charged capacitor
Nonvolatile memory (erasable)	Trapped electrons	No trapped electrons
Programable Read-only Memory	Fuse blown	Fuse intact
Bubble memory	No magnetic bubble	Bubble present
Magnetic disk	No flux reversal	Flush reversal
Compact disc	No pit	Pit

© 1992 Gaetano Borriello, 1994 Steven Burns

Combinational Versus Sequential Digital Circuits

- **A simple model of a digital system is a unit with inputs and outputs:**



- **Combinational means "memory-less". A digital circuit is combinational if its output values only depend on its input values.**
- **Sequential systems exhibit behavoirs (output values) that depend not only on the current input values, but also on previous input values.**
- **In reality, all real circuits are sequential because the outputs do not change instananeously after an input change. (Why, and why is it then sequential?)**
- **A fundamental abstraction of digital design is to reason (mostly) about steady-state behaviors, that is, what are the outputs after sufficient time has elapsed for the system to make its required changes.**

© 1992 Gaetano Borriello, 1994 Steven Burns

Synchronous Sequential Digital Systems

- **After sufficient time has elapsed, the outputs of a combinational circuit depend only on the current inputs.**
- **Sequential circuits have memory; even after waiting for the transient activity to finish.**
- **The steady-state abstraction is so useful that most designers use a form of it when constructing sequential circuits:**
  - The sequential character of a system is represented as the "state" of the system.
  - Changes in system state only occur at specific times, controlled by an external "clock'.
  - The "clock period", the time that elapses between state changes, is sufficiently lengthy that the system reaches a steady-state before the next state change.

© 1992 Gaetano Borriello, 1994 Steven Burns

Synchronous Sequential Digital Circuit

- Using this abstraction, combinational circuit design techniques can be used to construct sequential circuits.
- We'll spend the next few weeks discussing combinational circuit design, and then go on to sequential circuit design.

© 1992 Gaetano Borriello, 1994 Steven Burns

Switches, Truth Tables, Algebra, Gates, Waveforms, Blocks, and Programs

- There are several levels at which to express and reason about a computation.
- Switches, truth tables, Boolean algebra, and gates are useful in representing steady-state combinational objects.
- Waveforms can be used to expression dynamic (non-steady-state or transient) behaviors of a system.
- Interconnected blocks of smaller, better understood, components are useful to describe large, complex systems.
- High-level, program-like, specifications are also helpful for complex systems.

© 1992 Gaetano Borriello, 1994 Steven Burns