Digital Logic Design Introduction

EECE 256

Dr. Sidney Fels
Steven Oldridge

Introduction

- Introduction to Digital Design
  - or, how I learned to love 0 and 1
- Follow text somewhat closely but augment
- Use the LIP structure
  - 2 weeks on, 1 week off
- Office hours:
  - Tues and Thurs, 2:30-2:30pm
- Two instructors available
  - Sid Fels and Steve Oldridge
  - plus TAs

Introduction

- Assignments handed out online
  - are expected to be completed
  - solutions provided shortly after assignment given out
  - not marked
- Quizzes during selected Friday tutorials and will be graded
- Grades:
  - Quizzes: 15%
  - Midterm: 25%
  - Final: 60%

Overview of Topics

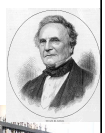
1. History of Digital Computing
   - why digital design is important
2. Digital Systems and Binary Numbers
   - how to represent and work with numbers
3. Boolean Algebra and Logic Gates
   - the basics functions with bits
4. Gate-level Minimization
   - how to make it simpler
5. Combinational Logic
   - let's make it more complex to do more

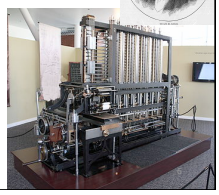
Overview of Topics

5. Synchronous Sequential Logic Design
   - adding a memory element so the system has state
6. Registers and Counters
   - useful state machines
7. Memory and Programmable Logic
   - larger scale state representations
8. Design using Different Digital Components
   - bringing it all together
9. Some HDL approaches to Digital Design
   - we'll use some HDL throughout term so you have an introduction to it
   - you have Verilog simulator with your text too

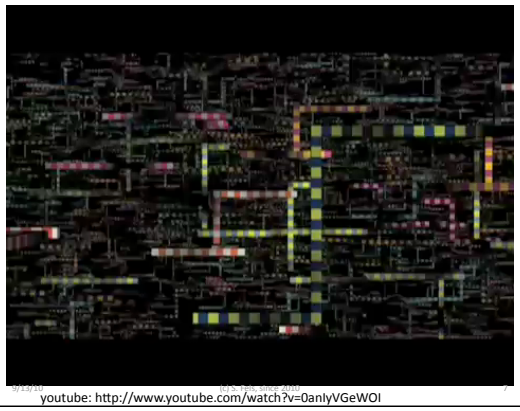
Digital Systems and Binary Numbers

- First computing device, called Difference Engine by Charles Babbage (1822)
  - Mechanical with crank power
  - Used base 10 as its number system
  - Calculate mathematical tables





permission: wikiMedia commons,



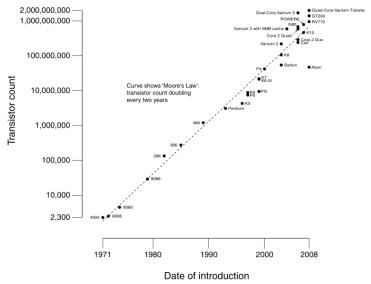
youtube: http://www.youtube.com/watch?v=0anlyVGeWOI

Meanwhile, some theory developments…

- George Boole (1854)
  - demonstrates that logic is math
  - creates notion of Boolean Algebra
    - mathematics of binary numbers
- Claude Shannon (1937)
  - showed two-valued Boolean Algebra
    - called switching algebra
  - opened the door for digital design
- the foundations of our current computers
- and, the starting point of our course

Where we are today

CPU Transistor Counts 1971-2008 & Moore's Law



permission: Creative Commons ASL, wgsimon

Where this course fits:

1. Problem statement
   - specifications, design requirements
2. Behaviour description
   - algorithms
   - flowcharts
   - state transition diagrams
3. Boolean logic and state
   - logic equations
   - digital circuit schematics
4. Hardware Implementation
   - TTL gates (AND, OR, NOT, XOR…)
   - Modules (counters, shift registers, …)
   - Programmable logic

Where this course fits:

1. Problem statement
   - specifications, design requirements
2. Behaviour description
   - algorithms
   - flowcharts
   - state transition diagrams
3. Boolean logic and state
   - logic equations
   - digital circuit schematics
4. Hardware Implementation
   - TTL gates (AND, OR, NOT, XOR…)
   - Modules (counters, shift registers, …)
   - Programmable logic

First half

Where this course fits:

1. Problem statement
   - specifications, design requirements
2. Behaviour description
   - algorithms
   - flowcharts
   - state transition diagrams
3. Boolean logic and state
   - logic equations
   - digital circuit schematics
4. Hardware Implementation
   - TTL gates (AND, OR, NOT, XOR…)
   - Modules (counters, shift registers, …)
   - Programmable logic

Second half

Where this course fits:

1. Problem statement
   - specifications, design requirements
2. Behaviour description
   - algorithms
   - flowcharts
   - state transition diagrams
3. Boolean logic and state -> HDL description
   - logic equations -> code
   - digital circuit schematics -> code
4. Hardware Implementation
   - ~~TTL gates (AND, OR, NOT, XOR…)~~
   - ~~Modules (counters, shift registers, …)~~
   - Programmable logic
   - ASIC

} HDL helps here;
- programming language
  * i.e., verilog, VHDL
- helps reduce error
- helps with complexity

Let's begin…

- Questions