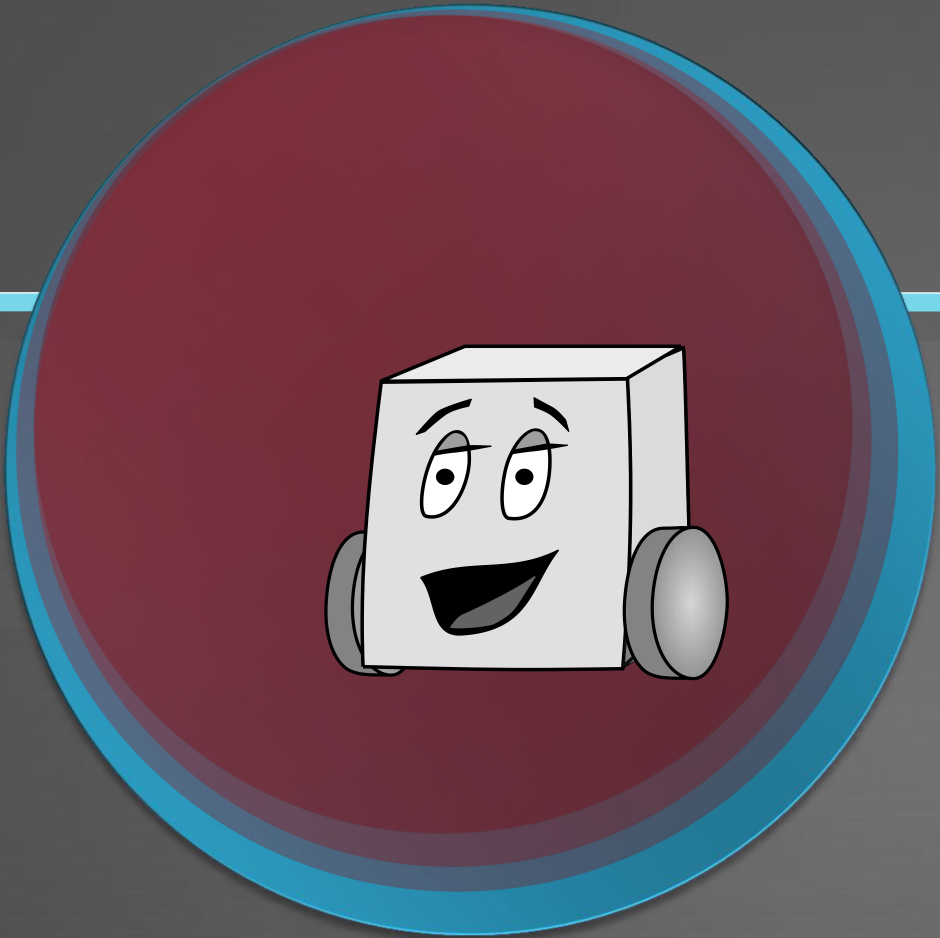


E11 Lecture 1: The Big Picture & Digital Systems

Profs. David Money Harris & Sarah Harris

Fall 2011

Outline

- Course Goals
- Syllabus
- From Zero to One
- Boolean Logic
- Number Systems

Course Goals

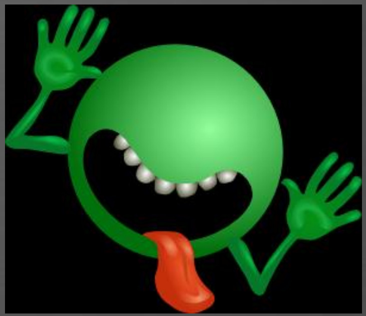
- Hands-on interdisciplinary introduction to what engineers and computer scientists do
  - Mechanical Engineering
  - Electrical Engineering
  - Computer Engineering
  - Computer Science
  - Design
  - Controls

Course Goals (Part 2)

- Give students a tastes of what engineers and computer scientists do to help make informed major decisions Provide practical skills including:
  - Machine shop
  - 3D CAD and printing
  - Soldering
  - C programming
  - Sensors & actuators
  - Analog & digital interfacing
  - Modeling
  - Embedded control systems

Course Goals (Part 3)

- Whet students' appetite to learn more advanced topics
- Develop skills:
  - Design – build – test – debug
  - Teamwork
  - Presentations
  - Technical writing
- Just plain fun!



www.clker.com

The Teaching Team

- Profs. David Money Harris & Sarah Harris
- Unusual course with a big component of peer teaching
  - Three upper-class lab section instructors:
    - Eric Zhang
    - Greg Fong
    - Brad Perfect & Katie Vinnedge
  - Six sophomore lab assistants who took the course Fall 10
    - Becca Thomas & Vijay Ramakrishnan
    - Tyler Robinson
    - Alistair Dobke
    - Stephen Pinto
  - Tutors
    - Jeremy Usatine & Josh Vasquez

Schedule

Week	Tue	Thurs	Lab	Problem Set (Due Tuesdays in class)
0: 8/29	Big Picture, Digital Systems	C Programming I	0: Shop safety briefing	
1: 9/5	Arduino Board	C Programming II	1: Arduino Board	
2: 9/12	Design Representation	C Programming III	2: 3D CAD & Printing	Programming 1: Welcome to Arduino
3: 9/19	Gold Codes	C Programming IV	3: Machine Shop	Programming 2: Music & Memory Game
4: 9/26	Analog Circuits	Analog Circuit Analysis	4: Robot Assembly	Programming 3: Gold Code Generation
5: 10/3	Diodes & Transistors	DEs, Capacitors & Inductors	5: Motors & Sensors	Hardware 4: Volts & Amps & Ohms, Oh My!
6: 10/10	Feedback Control	Motors	6: Line-Following Robot	Programming 5: Reaction Timer & Light Tag
7: 10/17	Fall Break	Line Following Race	Break week: no lab	
8: 10/24	Game Kickoff ; Team Dynamics	Mechanical Performance	Robot Design I	Programming 6: Gold Code Detection
9: 10/31	Robot Navigation	Debugging	Robot Design II	Hardware 7: 1st Order Circuits & Transistors
10: 11/7	Scrimmage	Batteries	Robot Design III	Hardware 8: Motors
11: 11/14	Guest Lecture	Robotics Show & Tell	Robot Design IV	
12: 11/21	Capture the Flag Game (5:30 pm in Galileo)	Thanksgiving: no class	No lab	
13: 11/28	Technical Writing	Presentation Skills	Technical Writing	
14: 12/5	Peer Editing	Engineering Outlook	Presentations	Project Report (due Thursday 12/8)

Grading

- Pass/fail. To Pass:
  - Regularly attend class and labs
  - Complete all but one of the weekly labs
  - Complete all but one of the homework assignments
  - Deploy an operational autonomous vehicle to play Capture the Flag
  - Make a presentation about the vehicle
  - Complete a final report documenting your vehicle

Collaboration Policy

- Labs 1-5:
  - Done on your own
  - You are welcome consult your instructors and classmates
- Lab 6 & Final Project:
  - Done with a partner
- Problem Sets:
  - Done on your own or with a partner
  - Both of you should be engaged in all aspects
  - OK to discuss with other students after making an effort yourself

From Zero to One

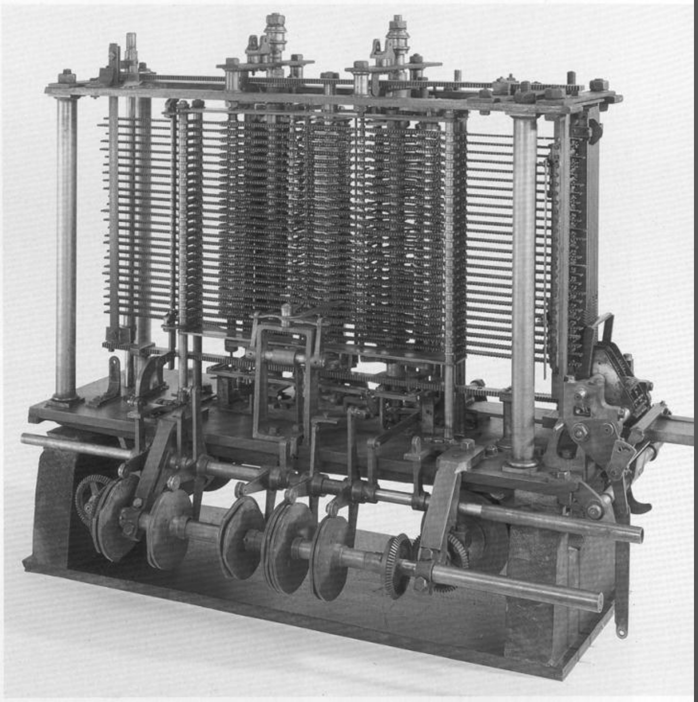
- We’ll be building digital systems
  - Simple building blocks: 0 and 1
    - 0 = FALSE
    - 1 = TRUE
  - Robustly assemble them into complex systems
  - (Much more on this in E85 and CS60 and E155)

Digital Abstraction

- 0 and 1 could be represented by any physical quantity
  - voltage
  - current
  - position of a mass
  - electron spin
  - reflectivity
  - magnetic polarity
  - water flow
- Most of these properties are continuous
- Only consider a discrete subset of the values

The Analytical Engine

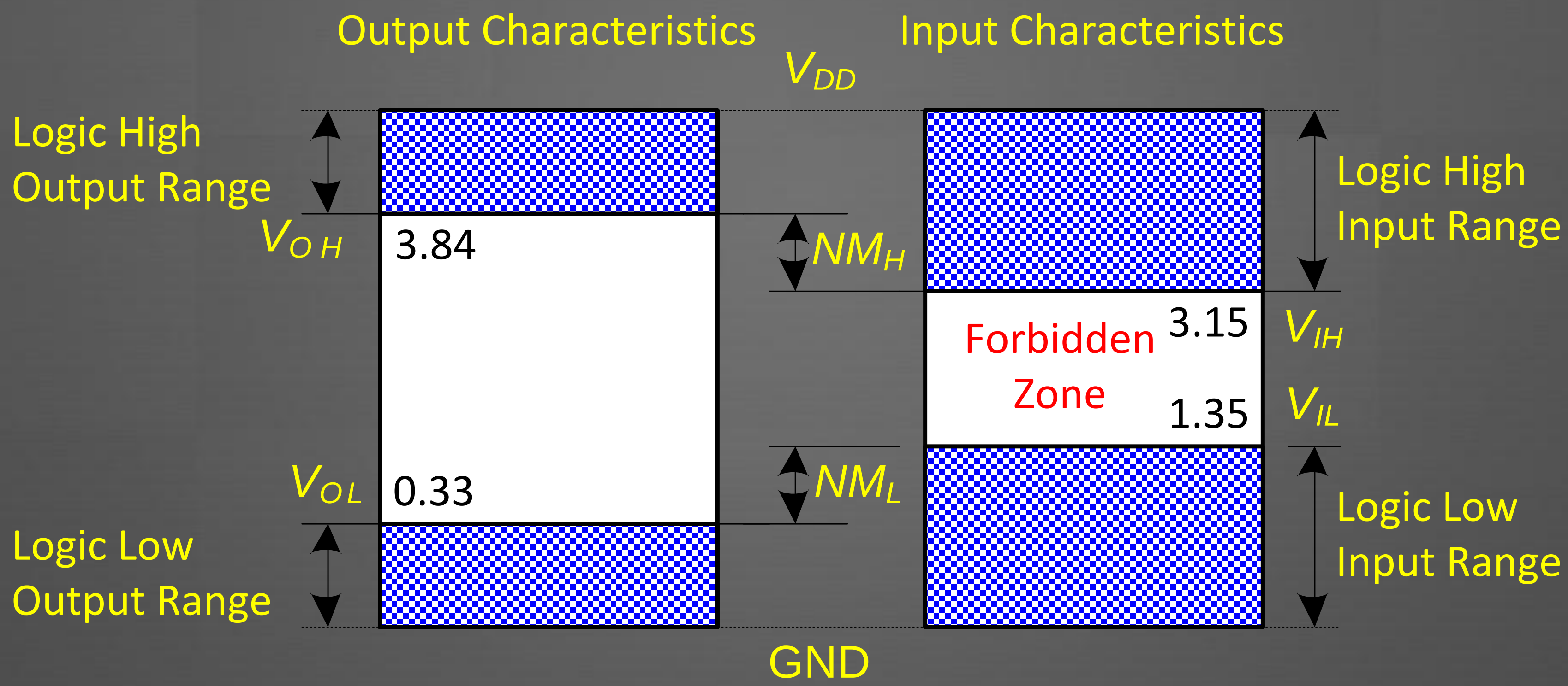
- Designed by Charles Babbage from 1834 – 1871
- Considered to be the first digital computer
- Built from mechanical gears, where each gear represented a discrete value (0-9)
- Babbage died before completion



High and Low Voltages

- Most digital systems today use voltage to process 0 and 1
  - 0 = low voltage
  - 1 = high voltage
- Power supply voltage: VDD (or VCC)
  - Formerly 5 V standard
  - Decreased toward 3.3, 2.5, 1.8, 1.2, 1.0, …
  - We'll use VDD = 5 V
- Ground = 0 V

Logic Levels



Bits & Binary Numbers

- A 0 or 1 represents one of two states
  - Hence, it is called a *bi*nary digi*t*, or bit
- N bits can represent one of ☐ states
- Write as binary numbers
  - 000…000 = 0
  - 111…111 = 2^N-1
- Leftmost bit is called most significant bit (weight = 2^{N-1})
- Rightmost bit is called least significant bit (weight = 1)

Number Systems

- **Decimal Numbers**

1000's column
100's column
10's column
1's column

$$5374_{10} = 5 \times 10^3 + 3 \times 10^2 + 7 \times 10^1 + 4 \times 10^0$$

five thousands | three hundreds | seven tens | four ones

- **Binary Numbers**

8's column
4's column
2's column
1's column

$$1101_2 = 1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 = 13_{10}$$

one eight | one four | no two | one one

Powers of 2

- $2^0 =$
- $2^1 =$
- $2^2 =$
- $2^3 =$
- $2^4 =$
- $2^5 =$
- $2^6 =$
- $2^7 =$
- $2^8 =$
- $2^9 =$
- $2^{10} =$
- $2^{11} =$
- $2^{12} =$
- $2^{13} =$
- $2^{14} =$
- $2^{15} =$

More Powers of 2

- 2^{10} = 1 kilo ≈ 1000 (1024)
- 2^{20} = 1 mega ≈ 1 million (1,048,576)
- 2^{30} = 1 giga ≈ 1 billion (1,073,741,824)
- 2^{40} = 1 tera ≈ 1 trillion
- 2^{50} = 1 peta ≈ 1 quadrillion

Number Conversion

- Decimal to binary conversion:
  - Convert 10011_2 to decimal
  -
- Decimal to binary conversion:
  - Convert 47_{10} to binary
  -

Addition

- **Decimal**

```
   11  ← carries
   3734
+  5168
-------
   8902
```

- **Binary**

```
   11  ← carries
   1011
+  0011
-------
   1110
```

Addition Examples

```
   1001          1011
+  0101       +  0110
--------      --------
```

Signed Numbers

- How could we represent negative numbers in binary?
  - "Two's complement" number system
  - Most significant bit has a weight of -2^{N-1}
- Examples: 5-bit two's complement numbers
  - -6_{10} =
  - 00011_2 =
  - 10011_2 =

Boolean Logic

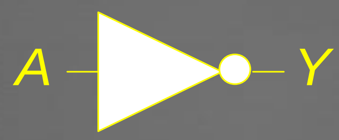
- Digital systems operate on 0’s and 1’s to produce more 0’s and 1’s
  - Called Boolean Logic

Charles Boole 1815-1864

- Born to working class parents
- Taught himself mathematics and joined the faculty of Queen's College in Ireland.
- Wrote *An Investigation of the Laws of Thought* (1854)
- Introduced binary variables
- Introduced the three fundamental logic operations: AND, OR, and NOT.

NOT Gate

NOT

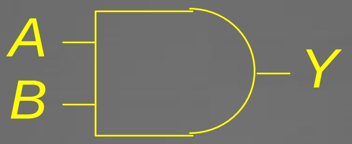


$$Y = \overline{A}$$

A	Y
0	
1	

AND Gate

AND

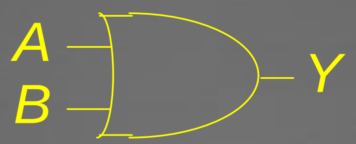


$$Y = AB$$

A	B	Y
0	0	
0	1	
1	0	
1	1	

OR Gate

OR

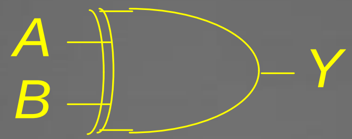


$$Y = A + B$$

A	B	Y
0	0	
0	1	
1	0	
1	1	

XOR Gate

XOR



$Y = A \oplus B$

A	B	Y
0	0	
0	1	
1	0	
1	1	