CS 3410: Computer System Organization and Programming

Anne Bracy
Computer Science
Cornell University

The slides are the product of many rounds of teaching CS 3410 by Professors Weatherspoon, Bala, Bracy, and Sirer.
“Sometimes it is the people that no one imagines anything of who do the things that no one can imagine”

--quote from the movie The Imitation Game
“Can machines think?”

-- Alan Turing, 1950

Computing Machinery and Intelligence
The Bombe used by the Allies to break the German Enigma machine during World War II

Enigma machine
Used by the Germans during World War II to encrypt and exchange secret messages
Turing Machine
1936

= abstract model for CPU that can simulate any algorithm
Who are you?

Demographics
Introduce yourself to the people next to you

“Sometimes it is the people that no one imagines anything of who do the things that no one can imagine.”

Turing Award Winners?
Course Objective

Understand the HW / SW interface software
  • How a processor works
  • How a computer is organized

Establish a foundation for building applications
  • How to write a good program
    – Good = correct, fast, and secure
  • How to understand where the world is going

Understand technology (past, present, future)
```c
#include <stdio.h>

int main() {
    printf("Hello world!\n");
    return 0;
}
```

How does it work?
I’m glad you asked...

15 weeks later and you’ll know!

“I know Kung Fu.”
Compilers & Assemblers

```c
int x = 10;
x = 2 * x + 15;
```

```
r0 = 0
```

```mips
addi r5, r0, 10  \(\rightarrow\) r5 = r0 + 10
muli r5, r5, 2  \(\rightarrow\) r5 = r5 * 2
addi r5, r5, 15  \(\rightarrow\) r5 = r5 + 15
```

```
op = addi  r0  r5  10
  001000000000010100000000000001010
  0000000000000101000101010000100000
  00100000101001010000000000001111

op = addi  r5  r5  15
```

Everything is a number!
How to Design a Simple Processor

```
<table>
<thead>
<tr>
<th>PC:</th>
<th>Addi</th>
<th>R5, R0, 10</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PC:</td>
<td>Mul</td>
<td>R5, R5, 2</td>
</tr>
<tr>
<td>04</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PC:</td>
<td>Addi</td>
<td>R5, R5, 15</td>
</tr>
<tr>
<td>08</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```
Instruction Set Architecture

ISA

• abstract interface between hardware and the lowest level software

• user portion of the instruction set plus the operating system interfaces used by application programmers
Covered in this course
Where did it begin?

Electrical Switch

- On/Off
- Binary

Transistor

The first transistor on a workbench at AT&T Bell Labs in 1947
Moore’s Law

1965

• # of transistors integrated on a die doubles every 18-24 months (i.e., grows exponentially with time)

Amazingly visionary

• 2300 transistors, 1 MHz clock (Intel 4004) - 1971
• 16 Million transistors (Ultra Sparc III)
• 42 Million transistors, 2 GHz clock (Intel Xeon) – 2001
• 55 Million transistors, 3 GHz, 130nm technology, 250mm² die (Intel Pentium 4) – 2004
• 290+ Million transistors, 3 GHz (Intel Core 2 Duo) – 2007
• 721 Million transistors, 2 GHz (Nehalem) - 2009
• 1.4 Billion transistors, 3.4 GHz Intel Haswell (Quad core) – 2013
Microprocessor Transistor Counts 1971-2011 & Moore’s Law

- curve shows transistor count doubling every two years
The first transistor
- One workbench at AT&T Bell Labs
- 1947
- Bardeen, Brattain, and Shockley

An Intel Haswell
- 1.4 billion transistors
- 177 square millimeters
- Four processing cores

What are we doing with all these transistors?
Computer System Organization
Why take this course?

• Basic knowledge needed for *all* other areas of CS: operating systems, compilers, ...
• Levels are not independent
  hardware design ↔ software design ↔ performance
• Crossing boundaries is hard but important
  device drivers
• Good design techniques
  abstraction, layering, pipelining, parallel vs. serial, ...
• Understand where the world is going

*The Mysteries of Computing will be revealed!*