Lesson 20: Introduction to Logic Design

Prof. John Wenskovitch

10/24/2014
Last Time
• Addition and multiplication with floating-point numbers.

Today
• How does the CPU interpret 0s and 1s?
• What kinds of logic gates can we build with transistors to perform computations?
• How can we show the potential outputs of logical circuits?
Layered Design Approach

• Logic design is done using logic gates
• Often, we design a desired function using high-level languages at a somewhat higher level than 0/1 logic
• Bottom-up design
The Transistor

- Two types: N-type and P-type
Building an Inverter

```
    "P"-type TR
   /   \          "1"
  /     \        /  \\
A ----- Y        "0"
   "N"-type TR
```

A diagram showing the connections and labels for an inverter circuit.
$A = 1$
A = 0
Abstracting – A “Not” Gate
Other Logic Gates

2-input AND
\[ Y = A \land B \]

2-input OR
\[ Y = A \lor B \]

2-input NAND
\[ Y = \overline{A \land B} \]

2-input NOR
\[ Y = \overline{A \lor B} \]
Boolean Algebra

• George Boole (1815-1864) – mathematician and philosopher; inverter of Boolean algebra, the basis of all computer arithmetic

• Binary values: \{0,1\}
• Two binary operators: \text{AND ( } \times, \cdot \text{)}, \text{OR ( } + \text{)}
• One unary operator: \text{NOT ( } \sim \text{)}
## Truth Tables

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>A \cdot B</th>
<th>A + B</th>
<th>\sim A</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>
Boolean Algebra

• **Idempotent Law**
  - \( a \cdot a = a + a = a \)

• **Commutative Law**
  - \( a \cdot b = b \cdot a \)
  - \( a + b = b + a \)

• **Associative Law**
  - \( a \cdot (b \cdot c) = (a \cdot b) \cdot c \)
  - \( a + (b + c) = (a + b) + c \)

• **Distributive Law**
  - \( a \cdot (b + c) = a \cdot b + a \cdot c \)
  - \( a + (b \cdot c) = (a + b) \cdot (a + c) \)
Boolean Algebra

• **De Morgan’s Laws** (Augustus De Morgan)
  - \(\neg(a \cdot b) = \neg a + \neg b\)
  - \(\neg(a + b) = \neg a \cdot \neg b\)

• **Miscellaneous Others**
  - \(a + (a \cdot b) = a\)
  - \(a \cdot (a + b) = a\)
  - \(\neg\neg a = a\)
  - \(a + \neg a = 1\)
  - \(a \cdot (\neg a) = 0\)

  “It is not true that I ate the sandwich and the soup”
  ...is the same as...
  “I didn’t eat the sandwich or I didn’t eat the soup.”
## Truth Tables

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>A \cdot B</th>
<th>A + B</th>
<th>\sim (A + B)</th>
<th>(A \cdot B) + \sim (A + B)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>
Any Questions?