Lecture 21: Combinational Circuits

- Integrated Circuits
- Combinational Circuits
  - Multiplexer
  - Demultiplexer
  - Decoder
  - Adders
  - ALU

Integrated Circuits

- Circuits use modules that contain multiple gates packaged together, rather than individual gates.
- These are called Integrated Circuits (ICs, chips)
  - SSI (small scale integration): 1-10 gates/chip
  - MSI (medium scale integration): 10-100 gates/chip
  - LSI (large scale integration): 100 – 100,000 gates/chip
  - VLSI (very large scale integration): more than 100,000 gates/chip

Integrated Circuits, cont.

- Current technology could put 5 million NAND gates on a chip!
- But… that chip would need 15,000,002 pins.
- With standard pin spacing, an 18km long chip.
- Instead, circuits are designed with a high gate/pin ratio.

• TTL example (older Tannenbaum)
Combinational Circuits

- Def: a set of interconnected gates whose output at any time is a function of the input at that time.
- The appearance of input is followed almost immediately by output, with only gate delays.
  - gate delays – how long it takes the signal to propagate through the gate
- We’ll look at some useful combinational circuits.

Multiplexer (MUX)

- A circuit that goes from many inputs to one output.
  - $2^n$ input lines (data)
  - 1 output line
  - $n$ select (control) lines
- The select lines are used to pick one of the input lines to directly output to the output line.
- At any time, one of the inputs is selected to be passed to the output.

MUX Diagram

- S1 and S0 are connected to AND gates in such a way that for any combination of S1 and S2, 3 of the AND gates will output 0.
- The 4th AND gate will output the value of the selected input line.
- So, 3 inputs to the OR-gate will always be 0, and the output of the OR-gate will equal the value of the selected input gate.
Multiplexer Uses

- Device controllers – several terminals connected to a MUX with a single line to the CPU. The MUX is used to select which terminal should transmit.
- Parallel-to-serial data converter – if we want to transmit an 8-bit character over a telephone line, where each bit needs to be transmitted one at a time, each bit could be put on the input of the MUX, then the control lines can sequentially be used to send each bit.

Demultiplexer

- Reverse of Multiplexer:
  - 1 input line
  - $2^n$ output lines
  - n control lines
- Control lines choose which of the output lines will get the input bit (the rest of the output lines will get 0)

Demultiplexer, cont.

Decoder

- A circuit that asserts one output line, depending on a pattern of input lines.
  - n input lines
  - $2^n$ output lines
- In this circuit, inputs are the select lines. The line they select gets a one, all other lines get zero.
  (recall, in a demultiplexer, the selected line got a 0 or 1 depending on what was on the data line)
Decoder, cont.

Decoder Uses

- Address decoding
  - Suppose you wish to construct a 1K-byte memory using 4 256x8-bit RAM chips. Want a single unified address space.

<table>
<thead>
<tr>
<th>Address</th>
<th>Chip</th>
</tr>
</thead>
<tbody>
<tr>
<td>000-0FF</td>
<td>0</td>
</tr>
<tr>
<td>100-1FF</td>
<td>1</td>
</tr>
<tr>
<td>200-2FF</td>
<td>2</td>
</tr>
<tr>
<td>300-3FF</td>
<td>3</td>
</tr>
</tbody>
</table>

- Each chip needs 8 address lines (256 bits). These are supplied by the low-order 8-bits of the address.
- High-order 2 bits (of the 10-bit address) are used to select 1 of 4 chips.

Adders

- Truth table for 1-bit addition:

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>Sum</th>
<th>Carry</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

A + B
-----
C S

- This can be drawn using our previous method:

- Or, we could notice that the Sum is the XOR of A and B:

- This circuit is known as a Half-Adder.
Half-Adder vs. Full Adder

- To be useful for arithmetic, need to also consider carry-in:

  1011
  + 0011
  -------
  10

  half-adder computes this correctly
  half-adder wouldn’t calculate this correctly because it doesn’t consider the carry-in from the previous bit.

Full-Adder

- For multiple-bit addition, need a full adder.
- Truth table:

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>CarryIn</th>
<th>Sum</th>
<th>CarryOut</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</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>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

Full-Adder, cont.

- For sixteen bit words, wire together 16 1-bit full adders.
  - Wire CarryIn for lowest bit to zero.
  - CarryIn for the remaining bits should be wired to the CarryOut of the previous bit.

Arithmetic Logic Units

- Most computers have a single circuit for performing AND, OR, and sum of two words.
- For n-bit words, built from n identical circuits or individual bit positions.
- These are known as 1-bit ALUs or bit slices.
ALU

- Figure 3-19 from Tannenbaum.