Homework

• Reading
  – Tokheim, Chapter 12-1 through 12-4
• Machine Projects
  – MP4 due at start of next class
• Labs
  – Continue with your assigned section
Read Only Memory (ROM)

- PC uses it to hold BIOS for system and I/O drivers
- Slow (100-200 nanoseconds)
- Various forms:
  - Read Only Memory (ROM)
  - Programmable Read Only Memory (PROM)
  - Erasable Programmable Read Only Memory (EPROM)
  - Electrically Erasable Programmable ROM (EEPROM)
  - Flash memory
Read Only Memory (ROM)

- Diode Matrix Design

Presence of a diode means a 1, absence of a diode = 0
Programmable ROM

- Diode Matrix Design

Starts with all diodes/all ones. Burn out diodes where value needs to be zero.
Random Access Memory (RAM)

- Static RAM = an array of full flip-flops
- Simple interface and fast (10-20 nsec) but more costly (2 - 4X) than ROM

- Small capacity compared to Dynamic RAMs
Dynamic RAM (DRAM)

- Square array of simple one transistor memory cells
- Capacitance at input to transistor remembers a 0 or 1

- Reading contents of the cell is destructive
- Over time, the charge leaks away (in milliseconds)
Dynamic RAM (DRAM)

• For both reasons, HW must perform a memory refresh
• Regularly reading/writing on a row and column basis
• Advantages are:
  – Cells are simple
  – Uses less power
• Disadvantage:
  – Slower (20 - 30 nsec access time)
  – Total cycle time is 2X due to refresh after read
• Bottom line
  – 2-4X less chip area and 2-4X less power
  – Interleaving and access in column or page mode
Example Layouts of Memory Parts

- **Organized 1 Meg x 1 or Organized 128K x 8**

  - **Address Bits**

    - (1 Meg Bits)
    - One Data Bit Per Chip
    - Need 32 Chips
    - For a 32 bit bus

    ![Diagram 1]

    Number of Data Bit Pins Attached to Data Bus

    ![Diagram 2]
Full Address Decoding

• Each location within a memory component responds to only one unique address
• Need to use all the address lines in decoding
• May choose not to fill-in some portions of the address space with real memory (empty spots)
Addressing Memory Components

- **CPU**
  - $A_{23}$
  - $A_{12}$
  - $A_{11}$
  - $A_{01}$
  - $A_{00}$

- **M0** 4K x 8
  - $D_0$-$D_7$
  - W/R#

- **M1** 4K x 8
  - $D_0$-$D_7$
  - W/R#

- **Combinational Logic**
  - $CS0#$
  - $CS1#$
Memory Map and Comb. Logic

<table>
<thead>
<tr>
<th></th>
<th>CS0</th>
<th>CS1</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>00 0000</td>
<td>00 0000</td>
</tr>
<tr>
<td></td>
<td>00 0FFF</td>
<td>00 0FFF</td>
</tr>
<tr>
<td></td>
<td>00 1000</td>
<td>00 1000</td>
</tr>
<tr>
<td></td>
<td>00 1FFF</td>
<td>00 1FFF</td>
</tr>
<tr>
<td></td>
<td>00 2000</td>
<td>00 2000</td>
</tr>
</tbody>
</table>

* Empty Space (Bus Error If Accessed)

<table>
<thead>
<tr>
<th></th>
<th>FF FFFF</th>
</tr>
</thead>
</table>

A23 A22 A21 A20 A19 A18 A17 A16 A15 A14 A13 A12 A11 A10 A09 A08

<table>
<thead>
<tr>
<th>CS0</th>
<th>X X X X</th>
</tr>
</thead>
<tbody>
<tr>
<td>00 0000</td>
<td>0 0 0 0 0 0 0 0 0 0 0 0 0</td>
</tr>
<tr>
<td>CS1</td>
<td>1 X X X X</td>
</tr>
<tr>
<td>00 0000</td>
<td>0 0 0 0 0 0 0 0 0 0 0 0 0 1</td>
</tr>
</tbody>
</table>

A23 A22 A21 A20 A19 A18 A17 A16 A15 A14 A13 A12 A11 A10 A09 A08

<table>
<thead>
<tr>
<th></th>
<th>CS1#</th>
<th>CS0#</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Partial Address Decoding

• Simpler and less expensive
• Some address lines are NOT used in the address decoding process to generate chip enable signals
• Many groups of addresses can map to the same physical memory chip
Partial Address Decoding

CPU

A_{23}
A_{12}
A_{11}
A_{01}
A_{00}

M0
4K x 8

M1
4K x 8

D_0-D_7

D_0-D_7

D_0-D_7

R/W*

R/W#

R/W#

CS0#

CS1#
Memory Map for Partial Decoding

CS0 is repeated 2,048 times in the memory space 00 0000 to 7F FFFF

CS1 is repeated 2,048 times in the memory space 80 0000 to FF FFFF
Comb. Logic for CS0 and CS1
Mixed Address Decoding

• A mixture or a compromise between partial and full address decoding
• Divide the memory space into a number of fully decoded blocks, generally of equal size
• Use high-order address bits to select the block and low-order address bits to select a sub-block
Comb. Logic for CS0, CS1, CS2, & CS3

Logic Diagram for CS0# and CS1#
Similar to Diagram on Slide 11
Except A_{22} and A_{12} are swapped
Memory Map for Mixed Decoding

CS0
Empty
CS1
Empty
CS2
CS3
Empty
CS3

00 0000
00 0FFF
00 1000
3F FFFF
40 0000
40 0FFF
40 1000
7F FFFF
80 0000
FF FFFF

CS0/CS1 each decoded only once in the memory space 00 0000 to 7F FFFF

CS2/CS3 are repeated 1,024 times in the memory space 80 0000 to FF FFFF