**ECE 3401 Lecture 16**

**Memory & Timing Issues**

### Memories Overview

- **Memories**
  - Memory categories and timing
  - Random access memory (RAM):
    - SRAM
    - DRAM
  - Timing issues
    - Sequential system timing requirements
    - Clock skew and clock jitter
    - Clock distribution

### Memories

<table>
<thead>
<tr>
<th>Read-Write Memory</th>
<th>Read-Only Memory</th>
</tr>
</thead>
<tbody>
<tr>
<td>Volatile Memory</td>
<td>Non-volatile Memory</td>
</tr>
<tr>
<td>Random Access</td>
<td>Sequential Access</td>
</tr>
<tr>
<td>DRAM</td>
<td>FIFO</td>
</tr>
<tr>
<td>SRAM</td>
<td>EPROM</td>
</tr>
<tr>
<td></td>
<td>EEPROM</td>
</tr>
<tr>
<td></td>
<td>FLASH</td>
</tr>
</tbody>
</table>

- Volatile: need electrical power
- Nonvolatile: magnetic disk, retains its stored information after the removal of power
- Random access: memory locations can be read or written in a random order
- EPROM: erasable programmable read-only memory
- EEPROM: electrically erasable programmable read-only memory
- FLASH: memory stick, USB disk
- Access pattern: sequential access: (video memory streaming) first-in-first-out (buffer), last-in-first-out (stack), shift register, content-addressable memory

### Memory Operation Timing - Reading

- Most basic memories are asynchronous
- Storage in latches or storage of electrical charge
- No clock
- Controlled by control inputs and address, which are controlled by CPU and synchronized by its own clock
- Timing of signal changes/data observation is critical to the operation

### Memory Operation Timing - Writing

- Write cycle: the maximum time from the application of the address to the completion of all internal operations required to store a word
- Critical times measured with respect to edges of write pulse (1-0-1):
  - Address must be established at least a specified time before 1-0 and held for at least a specified time after 0-1 to avoid disturbing stored contents of other addresses
  - Data must be established at least a specified time before 0-1 and held for at least a specified time after 0-1 to write correctly
**VHDL code for ROM**

```vhdl
ARCHITECTURE behav OF rom8x4 IS
BEGIN
PROCESS(addr)
BEGIN
CASE addr IS
  when "000" => q <= "0001";
  when "001" => q <= "0000";
  when "010" => q <= "0111";
  when "011" => q <= "1101";
  when "100" => q <= "1000";
  when "101" => q <= "1100";
  when "110" => q <= "0110";
  when "111" => q <= "1011";
  when others => NULL;
END CASE;
END process;
END behav;
```

**ROMS in Xilinx library**

- ROM16x1 (16-word by 1 bit ROM)
- ROM32x1
- ROM64x1
- ROM128x1
- ROM256x1

**How do we use ROM**

- VHDL components in a VHDL design
- Graphical components in a schematic

**Random Access Memories (RAMs)**

- **Read/Write memory**
- **Types:**
  - **Static RAM (SRAM):**
    - Once a word is written at a location, it remains stored as long as power is applied to the chip, unless the same location is written again.
    - Fast speed, but their cost per bit higher.
    - Application: Caches memories in Microprocessor
  - **Dynamic RAM (DRAM):**
    - The data stored at each location must be periodically refreshed by reading it and then writing it back again, otherwise it disappears.
    - Their density is greater and their cost per bit lower, but the speed is slower.

**RAMS in Xilinx library**

- Static Block RAMs
- Single port
  - RAMB4_S1, RAMB4_S2, RAMB4_S8, RAMB4_S16
- Dual port
  - RAMB4_S1_S1, RAMB4_S1_S2, ... RAMB4_S16_S16

The difference between single port RAM and dual port RAM is that single port RAM can be accessed at one address at one time, thus you can read/write only one memory cell during each clock cycle. Dual port RAM has ability to simultaneously read and write different memory cells at different addresses.

SPRAM uses a 6 transistor basic ram cell, while the dual port ram cell uses 8 transistor cell for memory.
RAMs in Xilinx library

```vhdl
component RAMB4_S is
    attribute RamB4_S : string constant := "Ramb4_S";
    -- Data interface
    port map (DataOut : out in std_logic_vector);
    -- Control interface
    port map (DataIn : in out std_logic_vector);
    -- Address interface
    port map (Address : in std_logic_vector);
    -- Clock interface
    port map (Clock : in std_logic);
end component;
```

SRAM Cell

- Array of storage cells used to implement static RAM
- Storage Cell
  - SR Latch
  - Input “Select” for control
- Dual Rail Data
  - Inputs B and R
  - Outputs C and Q

SRAM Bit Slice

- Represents all circuitry that is required for \(2^n\) 1-bit words
  - Multiple RAM cells
  - Control Lines:
    - Word select \(i\)
    - one for each word
    - Read/Write
    - Bit Select
  - Data Lines:
    - Data in
    - Data out

Cell Arrays and Coincident Selection

- Memory arrays can be very large =>
  - Large decoders
  - Large fanouts for the input bit lines
  - The decoder size and fanouts can be reduced by approximately \(\sqrt{c}\) using a coincident selection in a 2-D array: uses two decoders, one for words and one for bits:
    - Word select becomes Row select
    - Bit select becomes Column select

See next slide for example
Making Larger Memories

- We can make larger memories from smaller ones by using the decoded higher order address bits to control CS (chip select) lines, tying all address, data, and R/W lines in parallel.
- A 16-Word by 1-Bit memory constructed using 4-Word by 1-Bit memory.

Making Wider Memories

- Tie the address and control lines in parallel and keep the data lines separate.
- Example: make a 4-word by 4-bit memory from 4, 4-word by 1-bit memories.
- Note: Both 16x1 and 4x4 memories take 4 chips and hold 16 bits of data.

**DRAM**

- Basic Principle: Storage of information on capacitors.
- Charge and discharge of capacitor to change stored value.
- Use of transistor as "switch" to:
  - Store charges
  - Charge or discharge

Dynamic RAM (Contd.)

- Circuit, hydraulic analogy, and logical model.

Dynamic RAM - Bit Slice

- C driven by 3-state drivers.
- Sense amplifier is used to change the small voltage change on C into H or L.
- In the electronics, B, C, and the sense amplifier output are connected to make destructive read into non-destructive read.