ECE 545—Digital System Design with VHDL
Lecture 1

Digital Logic Refresher
Part B – Sequential Logic Building Blocks
Lecture Roadmap – Sequential Logic

- Sequential Logic Building Blocks
  - Flip-Flops, Latches
  - Registers, Shift Registers
  - Counters
  - RAM
Textbook References

- **Sequential Logic Review**
  - Stephen Brown and Zvonko Vranesic, *Fundamentals of Digital Logic with VHDL Design, 2\textsuperscript{nd} or 3\textsuperscript{rd} Edition*
    - Chapter 7 Flip-flops, Registers, Counters, and a Simple Processors
      - (7.3-7.4, 7.8-7.11 only)
    - OR your undergraduate digital logic textbook (chapters on sequential logic)
Sequential Logic Building Blocks

some slides modified from:
S. Dandamudi, “Fundamentals of Computer Organization and Design”
Introduction to Sequential Logic

• Output depends on the current input and the internal state
• Past inputs effects the internal state
• Sequential circuits consist typically of
  • Storage elements (flip-flop, latch, register, RAM, etc.)
  • Combinational logic
Main components of a typical synchronous sequential circuit (synchronous = uses a clock to keep circuits in lock step)
State-Holding Memory Elements

• Latch versus Flip Flop
  • Latches are level-sensitive: whenever clock is high, latch is transparent
  • Flip-flops are edge-sensitive: data passes through (i.e. data is sampled) only on a rising (or falling) edge of the clock
  • Latches cheaper to implement than flip-flops
  • Flip-flops are easier to design with than latches
• In this course, primarily use D flip-flops
D Latch vs. D Flip-Flop

Latch transparent when clock is high

“Samples” D on rising edge of clock
D latch

Graphical symbol

Truth table

<table>
<thead>
<tr>
<th>Clock</th>
<th>D</th>
<th>Q(t+1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>Q(t)</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

Timing diagram
D flip-flop

Graphical symbol

Truth table

<table>
<thead>
<tr>
<th>Clk</th>
<th>D</th>
<th>Q(t+1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>↑</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>↑</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>-</td>
<td>Q(t)</td>
</tr>
<tr>
<td>1</td>
<td>-</td>
<td>Q(t)</td>
</tr>
</tbody>
</table>

Timing diagram

Clock

D

Q

Time
D Flip-Flop with Asynchronous Set and Reset

- Bubble on the symbol means “active-low”
  - When Set = 0, set Q to 1
  - When Set = 1, do nothing
  - When Reset = 0, set Q to 0
  - When Reset = 1, do nothing
- “Set” and “Reset” also known as “Preset” and “Clear” respectively
- In this circuit, Set and Reset are asynchronous
  - Q changes immediately when preset or clear are active, regardless of clock
D Flip-Flop with Synchronous Reset

- Asynchronous active-low Reset: Q immediately clears to 0
- Synchronous active-low Reset: Q clears to 0 on rising-edge of clock
Register

- In typical nomenclature, a register is a name for a collection of flip-flops used to hold a bus.
- All flip-flops of a register share the same clock and control signals.
Shift Register

(a) Circuit

<table>
<thead>
<tr>
<th></th>
<th>Sin</th>
<th>Q₃</th>
<th>Q₂</th>
<th>Q₁</th>
<th>Q₀ = Q</th>
</tr>
</thead>
<tbody>
<tr>
<td>t₀</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>t₁</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>t₂</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>t₃</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>t₄</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>t₅</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>t₆</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>t₇</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>
4-bit Shift Registers: symbols

a) with Enable

\[
\begin{array}{c}
4 \\
\hline
D \\
\hline
\text{Sin} \\
\hline
\text{Clock} \\
\hline
\text{Enable} \\
\hline
\text{Q} \\
\hline
4
\end{array}
\]

b) with Enable and Parallel Load

\[
\begin{array}{c}
4 \\
\hline
D \\
\hline
\text{Load} \\
\hline
\text{Sin} \\
\hline
\text{Clock} \\
\hline
\text{Enable} \\
\hline
\text{Q} \\
\hline
4
\end{array}
\]
Shift Register with Enable: internal structure
Shift Register with Parallel Load: internal structure

Load

D(3)  Sin

Clock

Enable

Q(3)  Q(2)  Q(1)  Q(0)
Synchronous Up Counter

- **Enable (synchronous):** when high enables the counter, when low counter holds its value
- **Load (synchronous):** when load = 1, load the desired value into the counter
- **Output carry:** indicates when the counter “rolls over”
- **D3 downto D0, Q3 downto Q0** is how to interpret MSB to LSB
Random Access Memory (RAM)

• More efficient than registers for storing large amounts of data
• Can read and write to RAM
• Addressable memory
• RAM dimensions are:
  • (number of words) x (bits per word)
• Address is m bits, data is n bits
  • $2^m \times n$-bit RAM
• Example: address is 5 bits, data is 8 bits
  • 32 x 8 RAM
• Write Enable (WE)
  • When set, writing takes place at the next rising edge of the clock
Dual-Port RAM

- Two sets of input ports
  - \{DINA, ADDRA, WEA\}
  - \{DINB, ADDRB, WEB\}
- Two corresponding outputs
  - DOUTA
  - DOUTB
- One memory matrix
- Possible operations:
  - Read from two memory locations
  - Write to two different memory locations
  - Read from a memory location and write to a memory location (different or the same)