7. Basic processor design – Sequential logic


Prof. Valeria Bertacco & Prof. Scott Mahlke

EECS Department
University of Michigan in Ann Arbor, USA

© V. Bertacco & S. Mahlke, 2007
The material in this presentation cannot be copied in any form without our written permission
Announcements

- New professor today – Scott Mahlke (mall key)
  - For the next 3 classes, then Valeria for Exam 1 review and Exam 1, then me for the next 1 and a half months

- Homework 2 – Due today (Jan 30)
  - Homework 3 available on course website

- Project 1 – Due Friday (Feb 2)

- Exam 1
  - Review session – In class, Thursday Feb 8
  - Exam itself – In class, Tuesday Feb 13
  - Open book / open notes → Covers through finite state machines

- Regular discussion sections this week – cancelled
  - GSIs will have extra office hours during discussion slots in the Media Union
Lecture schedule

1. Combinational Logic:
   • Basics of electronics; logic gates, muxes, decoders

2. Sequential Logic:
   • clocks and data storage

3. ALU design
   • Building an adding circuit

4. State Machines
   • Building a simple processor
Levels of abstraction

- Quantum level, solid state physics
- Conductors, Insulators, Semiconductors.
- Doping silicon to make diodes and transistors.
- Building simple gates, boolean logic, and truth tables
- Combinational logic: muxes, decoders
- Clocks
- Sequential logic: latches, memory
- State machines
- Processor Control: Machine instructions
- Computer Architecture: Defining a set of instructions
Let’s look at the following circuit

What is the value of Q if \( \overline{R} \) is 1 and \( \overline{S} \) is 0?

\[
\begin{array}{cccc}
\overline{R} = 1 & \overline{Q} = 0 & Q = 1 & \overline{Q} = 0 \\
\overline{S} = 0 & Q = 1 & & \\
\end{array}
\]
Building the Truth Table

What is the value of Q if \( \bar{R} \) is 0 and \( \bar{S} \) is 1?

\[
\begin{array}{cccccc}
\bar{S} & \bar{R} & Q & \bar{Q} \\
0 & 1 & 1 & 0 \\
1 & 0 & 0 & 1 \\
\end{array}
\]
For a Basic Memory Cell

What is the value of Q if \( \overline{R} \) is 1 and \( \overline{S} \) is 1?

As long as \( R \) and \( S \) remain 1 then the value \( Q \) (and \( \overline{Q} \)) will remain unchanged. This value is stored in this circuit. **This is a basic memory cell.**
With unstable inputs 0,0

What is the value of $Q$ if $\bar{R}$ is 0 and $\bar{S}$ is 0?

<table>
<thead>
<tr>
<th>$\bar{S}$</th>
<th>$R$</th>
<th>$Q$</th>
<th>$\bar{Q}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>undef</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>Q</td>
<td>$\bar{Q}$</td>
</tr>
</tbody>
</table>
Transparent D Latch

Gating signal

<table>
<thead>
<tr>
<th>D</th>
<th>G</th>
<th>Q</th>
<th>Q</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>Q</td>
<td>Q</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>Q</td>
<td>Q</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

Set state is retained

Next state is set
Adding a clock to the mix

- We can design more interesting circuits if we have a clock signal.
- The use of a clock enables a sequential circuit to **predictably** change state (and store information).
- A clock signal alternates between 0 and 1 states at a fixed frequency (e.g. 100MHz)
- What should the clock frequency be?
Clocks

- Clock signal
  - Periodic pulse
  - Generated using oscillating crystal or ring oscillator
  - Distributed throughout chip using clock distribution net

- With clock signals we can create a new class of circuits called **sequential**
  - Output determined by inputs & previous state
Edge Triggered D Flip-flop

Latching starts

Data “sealed” inside flip-flop

Data should remain valid and stable during latching!
Q?

Data

Clock

Q
Why edge-triggered flip-flops?

In edge-triggered flip-flops, the latching edge provides convenient abstraction of “instantaneous” change of state.
Tri-state Logic

- The output of a gate can be any of three different states: one, zero or not connected
  - Need to disconnect the circuit. How?

Bus Driver

Implemented as a single transistor
A Static Memory Cell
A 4-Bit Register
Addressing Memory Arrays

One of the $2^4$ enable lines is turned ON

Replaces 16x1 MUX

16 X 1 BIT MEMORY
A Scheme with Fewer Components

16 X 1 BIT MEMORY
Putting it in a package

18-bit X 2M MEMORY (36 MBits)
Other Memories

- **Static RAM**
  - Built from sequential circuits
    - Takes 4-6 transistors to store 1 bit
    - Fast access (< 1 ns access possible)

- **Dynamic RAM**
  - Built using a single transistor and a capacitor
    - 1’s must be refreshed often to retain value
    - Slower access than static RAM
    - Much more dense layout than static RAM

- **ROM, PROM, Flash memory, etc.** Later