Sequential Circuits

Outline

- Bi – Stability / Meta – Stability
- Latches
- Flip – flops
- Schmitt Trigger
- Multivibrator circuits
- Counters and sequential machines

Combinational vs. Sequential Logic

(a) Combinational

\[ \text{Output} = f(\text{In}) \]

(b) Sequential

\[ \text{Output} = f(\text{In}, \text{Previous In}) \]
### Sequential Logic

2 storage mechanisms
- positive feedback
- charge-based

### Positive Feedback: Bi-Stability

Positive Feedback: Bi-Stability

### Meta-Stability

Gain should be larger than 1 in the transition region

### Outline

- Bi-Stability / Meta-Stability
- Latches
- Flip-flops
- Schmitt Trigger
- Multivibrator circuits
- Counters and sequential machines
**D Latch**

- When \( \text{CLK} = 1 \), latch is *transparent*
  - \( D \) flows through to \( Q \) like a buffer
- When \( \text{CLK} = 0 \), the latch is *opaque*
  - \( Q \) holds its old value independent of \( D \)
- transparent latch or level-sensitive latch

![D Latch Diagram]

**D Latch Design**

- Multiplexer chooses \( D \) or old \( Q \)

![D Latch Design Diagram]

**D Latch Operation**

- \( \text{CLK} = 1 \)
  - \( D \rightarrow Q \)

- \( \text{CLK} = 0 \)
  - \( D \rightarrow Q \)

![D Latch Operation Diagram]

**Latch Design**

- Pass Transistor Latch
  - **Pros**
    - Tiny
    - Low clock load
  - **Cons**
    - \( V_t \) drop
    - Nonrestoring
    - Backdriving
    - Output noise sensitivity
    - Dynamic
    - Diffusion input

Used in 1970's
Latch Design

- Transmission gate
  - No $V_t$ drop
  - Requires inverted clock

- Inverting buffer
  - Restoring
  - No backdriving
  - Fixes either
    - Output noise sensitivity
    - Or diffusion input
  - Inverted output

- Tristate feedback
  - Static
    - Backdriving risk
  - Static latches are now essential

- Buffered input
  - Fixes diffusion input
  - Noninverting
**Latch Design**

- Buffered output
  - No backdriving

- Widely used in standard cells
  - Very robust (most important)
  - Rather large
  - Rather slow
  - High clock loading

**Datapath latch**

- Smaller, faster
  - Unbuffered input

**Outline**

- Bi – Stability / Meta – Stability
- Latches

- Flip – flops
- Schmitt Trigger
- Multivibrator circuits
- Counters and sequential machines

**D Flip-flop**

- When CLK rises, D is copied to Q
- At all other times, Q holds its value
  
  *positive edge-triggered flip-flop, master-slave flip-flop*
D Flip-flop Design

- Built from master and slave D latches

D Flip-flop Operation

Flip-Flop: Timing Definitions

Maximum Clock Frequency
Flip-Flop Design

- Flip-flop is built as pair of back-to-back latches

Enable

- Enable: ignore clock when en = 0
  - Mux: increase latch D-Q delay
  - Clock Gating: increase en setup time, skew

Reset

- Force output low when reset asserted
- Synchronous vs. asynchronous

Set / Reset

- Set forces output high when enabled
- Flip-flop with asynchronous set and reset
**SR-Flip Flop**

<table>
<thead>
<tr>
<th>S</th>
<th>R</th>
<th>Q</th>
<th>(\bar{Q})</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>Q</td>
<td>Q</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>Q</td>
<td>0</td>
</tr>
<tr>
<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>
</tr>
</tbody>
</table>

**JK-Flip Flop**

<table>
<thead>
<tr>
<th>(J_n)</th>
<th>(K_n)</th>
<th>(Q_{n+1})</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>(Q_n)</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>(Q_n)</td>
</tr>
</tbody>
</table>

(a)  
(b)  
(c)  

**Other Flip-Flops**

- **Toggle Flip-Flop**
- **Delay Flip-Flop**

**Master-Slave Flip-Flop**

- **Master**  
- **Slave**  
  
- **RESET**  
- **CLEAR**
**Edge Triggered Flip-Flop**

![Flip-Flop Diagram]

**Race Condition**

- Back-to-back flops can malfunction from clock skew
  - Second flip-flop fires late
  - Sees first flip-flop change and captures its result
  - Called *hold-time failure or race condition*

![Race Condition Diagram]

**Nonoverlapping Clocks**

- Nonoverlapping clocks can prevent races
  - As long as nonoverlap exceeds clock skew
- Can be used for safe design
  - Industry manages skew more carefully instead

![Nonoverlapping Clocks Diagram]

**CMOS Clocked SR-FlipFlop**

![CMOS Clocked SR-FlipFlop Diagram]
Flip-Flop: Transistor Sizing

Charge-Based Storage

6 Transistor CMOS SR-Flip Flop

Master-Slave Flip-Flop

(a) Schematic diagram
(b) Non-overlapping clocks

Pseudo-static Latch

Overlapping Clocks Can Cause
• Race Conditions
• Undefined Signals
2 phase non-overlapping clocks

![Diagram of 2 phase non-overlapping clocks]

Flip-flop insensitive to clock overlap

![Diagram of flip-flop insensitive to clock overlap]

C²MOS avoids Race Conditions

![Diagram of C²MOS avoids Race Conditions]

2-phase dynamic flip-flop

![Diagram of 2-phase dynamic flip-flop]

C²MOS LATCH

![Diagram of C²MOS LATCH]
### Pipelining

<table>
<thead>
<tr>
<th>Clock Period</th>
<th>Adder</th>
<th>Absolute Value</th>
<th>Logarithm</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$a_1 - b_1$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>$a_2 + b_2$</td>
<td>$</td>
<td>a_2 + b_2</td>
</tr>
<tr>
<td>3</td>
<td>$a_3 - b_3$</td>
<td></td>
<td>$\log(</td>
</tr>
<tr>
<td>4</td>
<td>$a_4 + b_4$</td>
<td>$</td>
<td>a_4 + b_4</td>
</tr>
<tr>
<td>5</td>
<td>$a_5 - b_5$</td>
<td></td>
<td>$\log(</td>
</tr>
</tbody>
</table>

Non-pipelined version

Pipelined version

#### Pipelined Logic using C²MOS

What are the constraints on $F$ and $G$?

#### NORA CMOS Modules

Combinational logic

Latch

#### Doubled C²MOS Latches

Doubled n-C²MOS latch

Doubled n-C²MOS latch
### TSPC - True Single Phase Clock Logic

- Including logic into the latch
- Inserting logic between latches

### Master-Slave Flip-flops

(a) Positive edge-triggered D flip-flop
(b) Negative edge-triggered D flip-flop
(c) Positive edge-triggered D flip-flop using split-output latches

### Outline

- Bi – Stability / Meta – Stability
- Latches
- Flip – flops
- Schmitt Trigger
- Multivibrator circuits
- Counters and sequential machines

### Schmitt Trigger

- VTC with hysteresis
- Restores signal slopes
Noise Suppression using Schmitt Trigger

CMOS Schmitt Trigger

Schmitt Trigger Simulated VTC

CMOS Schmitt Trigger (2)
Outline

- Bi-Stability / Meta-Stability
- Latches
- Flip-flops
- Schmitt Trigger
- Multivibrator circuits
- Counters and sequential machines

Multivibrator Circuits

- Bistable Multivibrator
- Monostable Multivibrator
- Astable Multivibrator

Transition-Triggered Monostable

- Trigger circuit.
- Waveforms.

Monostable Trigger (RC-based)
Astable Multivibrators (Oscillators)

Ring Oscillator

simulated response of 5-stage oscillator

Voltage Controller Oscillator (VCO)

Simulated response of 5-stage oscillator

Current starved inverter

propagation delay as a function of control voltage

Relaxation Oscillator

\[ T = 2 \left( \log_3 \right) RC \]

Outline

- Bi – Stability / Meta – Stability
- Latches
- Flip – flops
- Schmitt Trigger
- Multivibrator circuits
- Counters and sequential machines
One-bit counter implementation

![Diagram of one-bit counter implementation](image)

One-bit counter operation

- All operations are performed as $s \phi_2$.
- XOR computes next value of this bit of counter.
- NAND/inverter compute carry-out.

n-bit counter structure

Sequence: $c_{out,n-1}, c_{in,n-1}, b_{n-1}, c_{out,n-2}, c_{in,n-2}, b_{n-2}, \ldots, c_{out,0}, c_{in,0}, b_0, 1$

Sequential machines

- Use memory elements to make primary output values depend on state + primary inputs.
- Varieties:
  - Mealy—outputs function of present state, inputs;
  - Moore—outputs depend only on state.
### Sequential machine definition

- Machine computes next state \( N \), primary outputs \( O \) from current state \( S \), primary inputs \( I \).
- Next-state function:
  - \( N = \delta(I, S) \).
- Output function (Mealy):
  - \( O = \lambda(I, S) \).

### FSM structure

- Primary inputs
- Combinational logic
- Memory elements
- Primary outputs
- Clock

### Summary

- **Bi-stable sequential circuits**
  - Latches (level sensitive circuits)
  - Flip-flops (edge triggered circuits)

- **Non bi-stable sequential circuits**
  - Schmitt Trigger (responds fast to a slowly changing input)
  - Multivibrator circuits
    - Monostable (only one stable state – generates pulse of predetermined width)
    - Astable (no stable states – output oscillates between two quasi stable states)