Digital Integrated Circuits
A Design Perspective
Designing Sequential Logic Circuits
Design Sequential Logic Circuits

- Introduction
- Timing
- Static Latches and Registers
- Dynamic Latches and Registers
Sequencing

- **Combinational logic**
  - output depends on current inputs
- **Sequential logic**
  - output depends on current and previous inputs
  - Requires separating previous, current, future
  - Called *state* or *tokens*
  - Ex: FSM, pipeline

![Finite State Machine](image1)

![Pipeline](image2)
Sequencing Cont.

- If tokens moved through pipeline at constant speed, no sequencing elements would be necessary.
- Ex: fiber-optic cable
  - Light pulses (tokens) are sent down cable
  - Next pulse sent before first reaches end of cable
  - No need for hardware to separate pulses
  - But dispersion sets min time between pulses
- This is called wave pipelining in circuits.
- In most circuits, dispersion is high
  - Delay fast tokens so they don’t catch slow ones.
Sequencing Overhead

• Use flip-flops to delay fast tokens so they move through exactly one stage each cycle.
• Inevitably adds some delay to the slow tokens
• Makes circuit slower than just the logic delay
  • Called sequencing overhead
• Some people call this clocking overhead
  • But it applies to asynchronous circuits too
  • Inevitable side effect of maintaining sequence
Sequencing Elements

- **Latch**: Level sensitive
  - a.k.a. transparent latch, D latch
- **Flip-flop**: edge triggered
  - A.k.a. master-slave flip-flop, D flip-flop, D register
- **Timing Diagrams**
  - Transparent
  - Opaque
  - Edge-triggered
Naming Conventions

• In our text:
  • a latch is level sensitive
  • a register is edge-triggered

• There are many different naming conventions
  • For instance, many books call edge-triggered elements flip-flops
  • This leads to confusion however
Latch versus Register

- **Latch**
  - stores data when clock is low

- **Register**
  - stores data when clock rises

**Diagram:**

- Latch: D → Q when Clk is low.
- Register: D → Q when Clk rises.

**Clock Waveforms:**

- Clock (Clk) pulses.
- Data (D) and output (Q) waveforms show the behavior of each component.

**Sequential Logic:**

- Digital IC

**Page:** 8/76
Latches

**Positive Latch**

\[ In \xrightarrow{D} Q \xrightarrow{G} Out \]

\[ CLK \]

\[ clk \]

**Negative Latch**

\[ In \xrightarrow{D} Q \xrightarrow{G} Out \]

\[ CLK \]

\[ clk \]
Timing Definitions

Setup time: the time that the data inputs must be valid before clock transition

Hold time: the time that the data must remain valid after the clock transition

Propagate delay time

Sequential logic
Characterizing Timing

Register

Latch

$t_{C2Q}$

$t_{D2Q}$
Sequencing Methods

- Flip-flops
- 2-Phase Latches
- Pulsed Latches
Design Sequential Logic Circuits

- Introduction
- Timing
- Static Latches and Registers
- Dynamic Latches and Registers
Timing Diagrams

Contamination and Propagation Delays

<table>
<thead>
<tr>
<th>Delay Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$t_{pd}$</td>
<td>Logic Prop. Delay</td>
</tr>
<tr>
<td>$t_{cd}$</td>
<td>Logic Cont. Delay</td>
</tr>
<tr>
<td>$t_{pcq}$</td>
<td>Latch/Flop Clk-Q Prop Delay</td>
</tr>
<tr>
<td>$t_{ccq}$</td>
<td>Latch/Flop Clk-Q Cont. Delay</td>
</tr>
<tr>
<td>$t_{pdq}$</td>
<td>Latch D-Q Prop Delay</td>
</tr>
<tr>
<td>$t_{pcq}$</td>
<td>Latch D-Q Cont. Delay</td>
</tr>
<tr>
<td>$t_{setup}$</td>
<td>Latch/Flop Setup Time</td>
</tr>
<tr>
<td>$t_{hold}$</td>
<td>Latch/Flop Hold Time</td>
</tr>
</tbody>
</table>

Sequentional Logic

Digital IC
Max-Delay: Flip-Flops

\[ t_{pd} < T_c - (t_{\text{setup}} + t_{pcq}) \]

sequencing overhead

\[ \text{clk} \quad \rightarrow \quad F_1 \hat{\rightarrow} \quad Q_1 \rightarrow \text{Combinational Logic} \rightarrow D_2 \rightarrow F_2 \quad \text{clk} \]

\[ T_c \]

\[ \text{clk} \quad \rightarrow \quad t_{pcq} \rightarrow \quad Q_1 \rightarrow \quad t_{pd} \rightarrow \quad t_{\text{setup}} \rightarrow \quad D_2 \]
Min-Delay: Flip-Flops

Race condition
Hold time failure
min-delay failure

\[ t_{cd} \geq t_{hold} - t_{ccq} \]
Clock skew

• Definition
  • The spatial variation in arrival time of a clock transition on an integrated circuit

• Sources
  • Static mismatches in the clock paths and differences in the clock load
Positive/negative skew

φ

Data

$CL \ R \ CL \ R \ CL \ R$

Positive skew

$CL \ R \ CL \ R \ CL \ R$

φ

Data

Negative skew
Clock skew about FF

$$t_{pdq} \leq T_c - \left( t_{pcq} + t_{setup} + t_{skew} \right)$$

*Positive skew for better condition*
Clock skew about FF

\[ t_{cd} \geq t_{\text{hold}} - t_{\text{ccq}} + t_{\text{skew}} \]

Positive skew for worse condition
Clock tree

sequentional logic
Design Sequential Logic Circuits

- Introduction
- Timing
- Static Latches and Registers
- Dynamic Latches and Registers
Static vs Dynamic Storage

• Static storage
  • preserve state as long as the power is on
  • have positive feedback (regeneration) with an internal connection between the output and the input
  • useful when updates are infrequent (clock gating)

• Dynamic storage
  • store state on parasitic capacitors
  • only hold state for short periods of time (milliseconds)
  • require periodic refresh
  • usually simpler, so higher speed and lower power
Static Latches and Registers

- The bistability principle
- Multiplexer-based latches
- Master-slave edge-triggered register
- Low-voltage static latches
- Static SR Flip-flops-writing data by pure force
If the gain in the transient region is larger than 1, only A and B are stable operation points. C is a metastable operation point.
Meta-Stability

Gain should be larger than 1 in the transition region
How to change the stored value

- Cutting the feedback loop
- Overpowering the feedback loop
SR Latch

<table>
<thead>
<tr>
<th>S</th>
<th>R</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>1</td>
<td>0</td>
<td>1</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>

Break the feedback
Clocked D Latch

D

Q

!Q

clock

transparent mode

clock

hold mode

Digital IC
Static Latches and Registers

- The bistability principle
- Multiplexer-based latches
- Master-slave edge-triggered register
- Low-voltage static latches
- Static SR Flip-flops-writing data by pure force
Mux-Based Latches

**Negative latch**
(transparent when CLK= 0)

**Positive latch**
(transparent when CLK= 1)

\[ Q = Clk \cdot Q + \overline{Clk} \cdot In \]

\[ Q = \overline{Clk} \cdot Q + Clk \cdot In \]
Writing into a Static Latch

*Use the clock as a decoupling signal, that distinguishes between the transparent and opaque states*

Converting into a MUX

Forcing the state (can implement as NMOS-only)
Mux-Based Latch

NMOS only

Non-overlapping clocks

Degraded high voltage
Static Latches and Registers

- The bistability principle
- Multiplexer-based latches
- Master-slave edge-triggered register
- Low-voltage static latches
- Static SR Flip-flops-writing data by pure force
Master-Slave (Edge-Triggered) Register

Two opposite latches trigger on edge
Also called master-slave latch pair
Master-Slave Register

Multiplexer-based latch pair

How to define setup time?

How to define propagation delay?

sequentioal logic
Clk-Q Delay

The diagram illustrates the Clk-Q Delay with time in nsec on the x-axis and Volts on the y-axis. The waveforms for CLK, D, and Q are shown, with labels for time delays $t_{cq(lh)}$ and $t_{cq(hl)}$. The graph is part of a discussion on sequentinal logic.
Setup Time

\[ (a) \ T_{\text{setup}} = 0.21 \text{ nsec} \]

\[ (b) \ T_{\text{setup}} = 0.20 \text{ nsec} \]
Reduced Clock Load
Master-Slave Register

should be designed more weak

ratioed

sequential logic
PowerPC Flipflop

- Master transparent
- Slave hold

Clock (clk) and complement (\!\text{clk}) waveforms:

- clk:
  - 0 → 1
  - 1 → 0

- \!\text{clk}:
  - 0 → 1
  - 1 → 0

Sequential logic diagram with state transitions:

- D → 0 → 1 → 0
- Q → 0 → 1

Digital IC
Avoiding Clock Overlap

(a) Schematic diagram

(b) Overlapping clock pairs
race and hazard

- First phase signal A and Second phase signal P3, meet with each one at gate G4 and have race

because \( P_3 \) after \( A \), exist a moment of “0” time, hazard happen at \( P_4 \)

there is an error signal for NOR
Why hazard happen

- Hazard is driven by race
- Race condition
  - when at least two input signal change to different direction
  - two changing input come at different time
- $A$ and $P_3$ are race signal

Hazard condition is $B=0$ let $A$ and $P_3$ meet, $B=1$ not
# Hazard examples

\[ P = A + \bar{A} \]

### 0-hazard

Stable state is “1”, 0-1, so 0 is unstable hazard state

### 1-hazard

Stable state is “0”, 1-0, so 1 is unstable hazard state

\[ P = A\bar{A} \]

<table>
<thead>
<tr>
<th>Expression</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>( P_1 = AB + \bar{A}C )</td>
<td>When ( B=C=1 ), ( P_1=A+\bar{A}, ) ”0” hazard happens</td>
</tr>
<tr>
<td>( P_2 = (A + B)(\bar{A} + C) )</td>
<td>When ( B=C=0 ), ( P_2=A+\bar{A}, ) ”1” hazard happens</td>
</tr>
<tr>
<td>( P_3 = \bar{A}B + A\bar{C} + \bar{B}C )</td>
<td>When ( B=1/C=0 ), ( P_3=A+\bar{A}, ) ”0” hazard happens</td>
</tr>
<tr>
<td>( P_4 = A\bar{B} + \bar{A}C + \bar{B}C )</td>
<td>When ( C=1/A=0 ), ( P_4=B+\bar{B}, ) ”0” hazard happens</td>
</tr>
<tr>
<td>( P_5 = A\bar{B} + \bar{A}C + \bar{B}C )</td>
<td>When ( A=1/B=0 ), ( P_5=C+\bar{C}, ) ”0” hazard happens</td>
</tr>
</tbody>
</table>
Design Sequential Logic Circuits

- Introduction
- Timing
- Static Latches and Registers
- Dynamic Latches and Registers
Dynamic ET Flipflop

\[ T_1 \quad I_1 \quad Q_M \quad T_2 \quad I_2 \]

\[ D \quad !clk \quad clk \quad 1 \quad !clk \quad clk \quad Q \]

\[ C_1 \quad C_2 \]

\[ t_{su} = \]
\[ t_{hold} = \]
\[ t_{c-q} = \]

sequentional logic
Dynamic ET Flipflop

master

\[ \text{D} \]

\[ !\text{clk} \]

\[ \text{clk} \]

\[ T_1 \]

\[ C_1 \]

\[ I \]

\[ Q_M \]

\[ T_2 \]

\[ C_2 \]

\[ !\text{clk} \]

\[ \text{clk} \]

slave

\[ \text{Q} \]

\[ I_2 \]

\[ !\text{clk} \]

\[ \text{clk} \]

master \text{ transparent}

slave \text{ hold}

\[ t_{su} = t_{pd_{tx}} \]

\[ t_{hold} = \text{zero} \]

\[ t_{c-q} = 2 \cdot t_{pd_{inv}} + t_{pd_{tx}} \]

\[ \text{clk} \]

\[ !\text{clk} \]

master \text{ hold}

slave \text{ transparent}
Dynamic ET FF Race Conditions

0-0 overlap race condition
\[ t_{\text{overlap0-0}} < t_{T1} + t_{l1} + t_{T2} \]

1-1 overlap race condition
\[ t_{\text{overlap1-1}} < t_{\text{hold}} \]
Fix 1: Dynamic Two-Phase ET FF

### Diagram

- **D**
- **clk1**
- **clk2**
- 
- **!clk1**
- **!clk2**
- **C1**
- **C2**
- **T1**
- **T2**
- **I1**
- **I2**
- **Q**
- **Q_M**

### Timing Diagrams

- **clk1**
- **clk2**
- **t_{non_overlap}**

### Notes

- **master transparent**
- **slave hold**
- **master hold**
- **slave transparent**

**sequential logic**

Digital IC
Fix 2: \( \text{C}^2\text{MOS (Clocked CMOS) ET} \) Flipflop

- A clock-skew insensitive FF
C²MOS (Clocked CMOS) ET Flipflop

- A clock-skew insensitive FF

![C²MOS ET Flipflop Diagram]

- Master transparent
- Slave hold

![Clock Waveforms]

- Master hold
- Slave transparent
C\textsuperscript{2}MOS FF 0-0 Overlap Case

If clock fall slope is too slow

Clock-skew insensitive as long as the rise and fall times of the clock edges are sufficiently small
C^2MOS FF 1-1 Overlap Case

1-1 overlap constraint

\[ t_{\text{overlap 1-1}} < t_{\text{hold}} \]

UNSAFE
C²MOS Transient Response

For a 0.1 ns clock

For a 3 ns clock (race condition exists)
Pipelining using $C^2$MOS

aka NORA (NO RAce) Logic

What are the constraints on F and G?
Fix 3: True Single Phase Clocked (TSPC) Latches

Negative Latch

Positive Latch

$\textit{hold when clk} = 1$

$\textit{transparent when clk} = 0$

$\textit{transparent when clk} = 1$

$\textit{hold when clk} = 0$
Embedding Logic in TSPC Latch
TSPC ET FF

Sequential logic
TSPC ET FF

master transparent
slave hold

clk

master hold
slave transparent

Digital IC
Simplified TSPC ET FF
Simplified TSPC ET FF

master transparent
slave hold

clk

master hold
slave transparent
Sizing Issues in Simplified TSPC ET FF

Transistor sizing

Original width
\( M_4, M_5 = 0.5\mu m \)
\( M_7, M_8 = 2\mu m \)

Modified width
\( M_4, M_5 = 1\mu m \)
\( M_7, M_8 = 1\mu m \)
Split-Output TSPC Latches

Positive Latch

transparent when clk = 1
hold when clk = 0

When In = 0, A = V_{DD} - V_{Tn}

Negative Latch

hold when clk = 1
transparent when clk = 0

When In = 1, A = | V_{Tp} |
Split-Output TSPC ET FF
# Flipflop Comparison Chart

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>#clk Id</th>
<th>#tr</th>
<th>t\textsubscript{su}</th>
<th>t\textsubscript{hold}</th>
<th>T\textsubscript{c-q}</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mux</td>
<td>Static</td>
<td>8 (clk-!clk)</td>
<td>20</td>
<td>3t\textsubscript{pinv}+t\textsubscript{ptx}</td>
<td>0</td>
<td>t\textsubscript{pinv}+t\textsubscript{ptx}</td>
</tr>
<tr>
<td>PowerPC</td>
<td>Static</td>
<td>8 (clk-!clk)</td>
<td>16</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2-phase</td>
<td>Ps-Static</td>
<td>8 (clk1-clk2)</td>
<td>16</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>T-gate</td>
<td>Dynamic</td>
<td>4 (clk-!clk)</td>
<td>8</td>
<td>t\textsubscript{ptx}</td>
<td>t\textsubscript{o1-1}</td>
<td>2t\textsubscript{pinv}+t\textsubscript{ptx}</td>
</tr>
<tr>
<td>C²MOS</td>
<td>Dynamic</td>
<td>4 (clk-!clk)</td>
<td>8</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TSPC</td>
<td>Dynamic</td>
<td>4 (clk)</td>
<td>11</td>
<td>t\textsubscript{pinv}</td>
<td>t\textsubscript{pinv}</td>
<td>3t\textsubscript{pinv}</td>
</tr>
<tr>
<td>S-O TSPC</td>
<td>Dynamic</td>
<td>2 (clk)</td>
<td>10</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AMD K6</td>
<td>Dynamic</td>
<td>5 (clk)</td>
<td>19</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SA 100</td>
<td>SenseAmp</td>
<td>3 (clk)</td>
<td>20</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
homework

- For each of the following sequencing styles, determine the maximum logic propagation delay available within a 500ps clock cycle. Assume there is zero clock skew
  - FF
  - Two-phase transparent latches
  - Pulsed latches with 80ps pulse width

<table>
<thead>
<tr>
<th>Sequencing element parameters</th>
<th>Setup Time</th>
<th>clk-to-Q Delay</th>
<th>D-to-Q Delay</th>
<th>Contamination Delay</th>
<th>Hold Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Flip-flops</td>
<td>65 ps</td>
<td>50 ps</td>
<td>n/a</td>
<td>35 ps</td>
<td>30 ps</td>
</tr>
<tr>
<td>Latches</td>
<td>25 ps</td>
<td>50 ps</td>
<td>40 ps</td>
<td>35 ps</td>
<td>30 ps</td>
</tr>
</tbody>
</table>
For each of the following sequencing styles, determine the minimum logic contamination delay in each clock cycle (or half-cycle, for two-phase latches). Assume there is zero clock skew:

- **FF**
- Two-phase transparent latches with 60ps of non-overlap between phases
- Pulsed latches with 80ps pulse width

### Sequencing element parameters

<table>
<thead>
<tr>
<th></th>
<th>Setup Time</th>
<th>clk-to-Q Delay</th>
<th>D-to-Q Delay</th>
<th>Contamination Delay</th>
<th>Hold Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Flip-flops</td>
<td>65 ps</td>
<td>50 ps</td>
<td>n/a</td>
<td>35 ps</td>
<td>30 ps</td>
</tr>
<tr>
<td>Latches</td>
<td>25 ps</td>
<td>50 ps</td>
<td>40 ps</td>
<td>35 ps</td>
<td>30 ps</td>
</tr>
</tbody>
</table>