LOGIC CIRCUITS

COMBINATIONAL
(non-regenerative)

SEQUENTIAL
(regenerative)

BISTABLE

MONOSTABLE

ASTABLE
COMBINATIONAL LOGIC CIRCUIT

MEMORY
BISTABLE BEHAVIOR

\[ \frac{V_o}{V_{in}} < 1 \]

\[ \frac{V_o}{V_{in}} >> 1 \]

Kenneth R. Laker, University of Pennsylvania
ASSUME: $C_g \gg C_d$

$v_{o1}(0) = v_{o2}(0) = V_{th}$

$i_{g1} = i_{d2} = g_m v_{g2}$  

$i_{g2} = i_{d1} = g_m v_{g1}$

$g_m v_{g2} = C_g \frac{dv_{g1}}{dt}$

$g_m v_{g1} = C_g \frac{dv_{g2}}{dt}$

$=> v_{oi} = \left( \frac{C_g}{g_m} \right)^2 \frac{d^2 v_{oi}}{dt^2}$  for $i = 1, 2$

$= \left( \frac{1}{\tau_0} \right)^2 \frac{d^2 v_{oi}}{dt^2}$

Kenneth R. Laker, University of Pennsylvania
\[ V_{oi} = \left( \frac{C_g}{g_m} \right)^2 \frac{d^2 V_{oi}}{dt^2} = \left( \frac{1}{\tau_0} \right)^2 \frac{d^2 V_{oi}}{dt^2} \quad \text{for } i = 1, 2 \]

\[ V_{o1}(t) = \frac{1}{2} \left( V_{o1}(0) - \tau_0 \frac{d V_{o1}}{dt}(0) \right) e^{-\frac{t}{\tau_0}} + \frac{1}{2} \left( V_{o1}(0) + \tau_0 \frac{d V_{o1}}{dt}(0) \right) e^{\frac{t}{\tau_0}} \approx 0 \]

\[ V_{o2}(t) = \frac{1}{2} \left( V_{o2}(0) - \tau_0 \frac{d V_{o2}}{dt}(0) \right) e^{-\frac{t}{\tau_0}} + \frac{1}{2} \left( V_{o2}(0) + \tau_0 \frac{d V_{o2}}{dt}(0) \right) e^{\frac{t}{\tau_0}} \approx 0 \]

NOTE THAT

\[ \frac{d V_{o1}}{dt}(0) = -\frac{d V_{o2}}{dt}(0) \]

\[ V_{o1}: \quad V_{th} \rightarrow V_{OH} \text{ or } V_{OL} \]

\[ V_{o2}: \quad V_{th} \rightarrow V_{OL} \text{ or } V_{OH} \]
As BISTABLE circuit settles from UNSTABLE Op-Pt to STABLE Op-Pt, signal travels around 2 INV loop \( n \) times

\[
\frac{v_{o1}(t)}{v_{o1}(0)} \approx e^{\frac{t}{\tau_0}}
\]

If during interval \( t = T \), signal travels the loop \( n \) times

\[
A^n \approx e^{\frac{T}{\tau_0}}
\]

Kenneth R. Laker, University of Pennsylvania
STATE OF LATCH can be EXTERNALLY SWITCHED between the 2 STABLE STATES

SET STATE: \( S = 1, R = 0 \) => \( Q = 1, \overline{Q} = 0 \)

RESET STATE: \( S = 0, R = 1 \) => \( Q = 0, \overline{Q} = 1 \)

HOLD: \( S = 0, R = 0 \) (like to cross-coupled Inverters)

NOT ALLOWED: \( S = 1, R = 1 \)
### NOR-based SR Latch

<table>
<thead>
<tr>
<th>S</th>
<th>R</th>
<th>$Q_{n+1}$</th>
<th>$\overline{Q}_{n+1}$</th>
<th>Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>$Q_n$</td>
<td>$\overline{Q}_n$</td>
<td>hold</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>set</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>reset</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>NOT allowed</td>
</tr>
</tbody>
</table>

Kenneth R. Laker, University of Pennsylvania
<table>
<thead>
<tr>
<th>S</th>
<th>R</th>
<th>( Q_{n+1} )</th>
<th>( \overline{Q}_{n+1} )</th>
<th>Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>( V_{OH} )</td>
<td>( V_{OL} )</td>
<td>( V_{OH} )</td>
<td>( V_{OL} )</td>
<td>M1, M2 ON, M3, M4 OFF</td>
</tr>
<tr>
<td>( V_{OL} )</td>
<td>( V_{OH} )</td>
<td>( V_{OL} )</td>
<td>( V_{OH} )</td>
<td>M1, M2 OFF, M3, M4 ON</td>
</tr>
<tr>
<td>( V_{OL} )</td>
<td>( V_{OL} )</td>
<td>( V_{OH} )</td>
<td>( V_{OL} )</td>
<td>M1, M4 OFF, M2 ON</td>
</tr>
<tr>
<td>( V_{OL} )</td>
<td>( V_{OL} )</td>
<td>( V_{OL} )</td>
<td>( V_{OH} )</td>
<td>M1, M4 OFF, M3 ON</td>
</tr>
</tbody>
</table>

Kenneth R. Laker, University of Pennsylvania
Estimate time to simultaneously switch $Q$ & $\overline{Q}$ : solution of two coupled differential equations.

**Pessimistic Estimate:** Assume $Q$ & $\overline{Q}$ switch in sequence

$$
\tau_{\text{rise},Q}(\text{SR latch}) = \tau_{\text{rise},Q}(\text{NOR 2}) + \tau_{\text{rise},\overline{Q}}(\text{NOR 2})
$$

For $S = 1, R = 0$
NAND BASED SR LATCH

V_{DD}

M1

M2

Q

S

R

M3

M4

\[ Q_{n+1} \] \[ \overline{Q}_{n+1} \]

\begin{tabular}{|c|c|c|c|}
\hline
S & R & Q_{n+1} & \overline{Q}_{n+1} \\
\hline
0 & 0 & 0 & 0 \\
0 & 1 & 1 & 0 \\
1 & 0 & 0 & 1 \\
1 & 1 & Q_n & \overline{Q}_n \\
\hline
\end{tabular}

Operation:

- NOT allowed
- set
- reset
- hold

Kenneth R. Laker, University of Pennsylvania
### NAND-based SR Latch

<table>
<thead>
<tr>
<th>S</th>
<th>R</th>
<th>$Q_{n+1}$</th>
<th>$\overline{Q}_{n+1}$</th>
<th>Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>NOT allowed</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>set</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>reset</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>$Q_n$</td>
<td>$\overline{Q}_n$</td>
<td>hold</td>
</tr>
</tbody>
</table>

**NOTE:** $S, R$ (NAND2) = $\overline{S}, \overline{R}$ (NOR2)

active high $S, R$  \hspace{1cm} active low $S, R$
CLOCKED LATCH AND FLIP-FLOP CIRCUITS

CLOCKED SR LATCH:

When \( CK = 0 \), \( S \), \( R \) have no influence of \( Q, \overline{Q} \)  => HOLD

Set State: \( CK = 1, S = 1, R = 0 \)
Reset State: \( CK = 1, S = 0, R = 1 \)
Not Allowed: \( CK = 1, S = 1, R = 1 \)

Active “High”
HOLD STATE: \( CK = 0, S = X, R = X \)
SET STATE: \( CK = 1, S = 1, R = 0 \)
RESET STATE: \( CK = 1, S = 0, R = 1 \)
NOT ALLOWED: \( CK = 1, S = 1, R = 1 \)

WHEN "GLITCH" ON S (OR R) OCCURS DURING \( CK = 1 \), Q IS SET (OR RESET)

LEVEL SENSITIVE: WHEN \( CK = 1 \), ANY CHANGES IN S, R WILL EFFECT Q.
CMOS IMPLEMENTATION OF CLOCKED NOR BASED SR LATCH

Kenneth R. Laker, University of Pennsylvania
WHEN $CK = 1$, $S$, $R$ HAVE NO INFLUENCE OF $Q$, $\overline{Q}$ => HOLD

SET STATE: $CK = 0$, $S = 0$, $R = 1$
RESET STATE: $CK = 0$, $S = 1$, $R = 0$
NOT ALLOWED: $CK = 0$, $S = 0$, $R = 0$

ACTIVE “LOW”
CMOS IMPLEMENTATION OF CLOCKED NAND BASED SR LATCH

Kenneth R. Laker, University of Pennsylvania
CLOCKED JK LATCH:

NAND BASED CLOCKED JK LATCH

NO NOT ALLOWED INPUT COMBINATION

Kenneth R. Laker, University of Pennsylvania
### J KCK SR LATCH

<table>
<thead>
<tr>
<th>J</th>
<th>K</th>
<th>Q_n</th>
<th>Q̅_n</th>
<th>S</th>
<th>R</th>
<th>Q_n+1</th>
<th>Q̅_n+1</th>
<th>Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>hold</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>hold</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>reset</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>reset</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>set</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>set</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>toggle</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>toggle</td>
</tr>
</tbody>
</table>

CK = 0 => hold
CK = 1 => active

CK = 1

OSC

Kenneth R. Laker, University of Pennsylvania
TO PREVENT OSCILLATION WHEN J = K = 1:

\[ \tau_{\text{JKP}} > T_1 \]

\[ \tau_{\text{JKP}} = \text{INPUT-OUTPUT PROP DELAY OF JK LATCH} \]

**NOR BASED CLOCKED JK LATCH**

Kenneth R. Laker, University of Pennsylvania
CMOS AOI IMPLEMENTATION NOR BASED CLOCKED JK LATCH
JK TOGGLE SWITCH

J = K = 1

IFF \( \tau_{JKP} > T_1 \)

OUTPUT Q CHANGES ONLY ONCE PER CLOCK PERIOD
MASTER-SLAVE FLIP-FLOP

USING NAND-BASED JK LATCHES
"GLITCH"

"one's catching"
CMOS D-LATCH AND EDGE-TRIGGERED FLIP-FLOP

D-Latch Version 1

Kenneth R. Laker, University of Pennsylvania
D

CK = 1

D

CK = 0

Kenneth R. Laker, University of Pennsylvania
**t\textsubscript{setup}** - time before the positive-**CLK** edge the **D-input** has to be stable.

**t\textsubscript{hold}** - time after positive-**CLK** edge that the **D-input** has to remain stable.

**t\textsubscript{clock-to-Q}** - Delay from the positive-**CLK** edge to new value of **Q** output.

Kenneth R. Laker, University of Pennsylvania
METASTABILITY AND SYNCRONIZATION FAILURES

If data and clock do not satisfy the setup & hold time constraints of a register, then synchronization failure may occur. This due to inherent analog nature of storage elements.

METASTABLE STATE - indeterminate state between "1" & "0", i.e. latch is perfectly balanced between making decision for "1" or "0". In practice noise will eventually arbitrarily push latch output to "0" or "1".

Example: register entering metastable state

![Diagram showing metastable states with delays and timing relationships]
Negative D - Latch

Positive D - Latch
D-Latch Version 2
POSITIVE EDGE - TRIGGERED MASTER-SLAVE D FLIP-FLOP

1. CLK = 0: master $Q_m$ tracks current $D$; slave $Q_s = \text{previous } D$ sample
2. CLK = 0 -> 1: master stores $Q_m = D$ (new $D$ sample)
3. CLK = 1: master passes $Q_m = D$ to slave output $Q_s$
4. CLK = 1 -> 0: slave locks in new $D$, and master $Q_m$ begins tracking $D$
LEVEL/EDGE-SENSITIVE LATCH/REGISTER TIMING

Negative Latch

Positive Latch

Q stored when CLK high

Q stored when CLK low