**Set-Reset (SR) Latch**

Asynchronous

Level sensitive

cross-coupled Nor gates

active high inputs (only one can be active)

cross-coupled Nand gates

active low inputs (only one can be active)

<table>
<thead>
<tr>
<th>S</th>
<th>R</th>
<th>Q</th>
<th>Q</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>Q+</td>
<td>Q+</td>
<td>Storage State</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>Reset</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>Set</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0-?</td>
<td>0-?</td>
<td>Indeterminate State</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>S</th>
<th>R</th>
<th>Q</th>
<th>Q</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>1-?</td>
<td>1-?</td>
<td>Indeterminate State</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+</td>
<td>Q+</td>
<td>Storage State</td>
</tr>
</tbody>
</table>
Enabled Set-Reset (SR) Latch

Asynchronous

Level sensitive
cross-coupled AOI21 gates
active high inputs (S & R cannot be active)

+ Storage State
+ Storage State
Reset
Set
Indeterminate State

 cross-coupled OAI21 gates
active low inputs (S & R cannot be active)

Indeterminate State
Set
Reset
Storage State
Storage State
**Transparent D Latch**

Asynchronous
Level sensitive
cross-coupled AOI21 gates and inverter
active high enable (E)

cross-coupled OAI21 gates
active low enable (E)

---

### E D Q Function

<table>
<thead>
<tr>
<th>E</th>
<th>D</th>
<th>Q</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>x</td>
<td>$Q^+$</td>
<td>Storage State</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>Transparent Mode</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>Transparent Mode</td>
</tr>
</tbody>
</table>

---

### E D Q Function

<table>
<thead>
<tr>
<th>E</th>
<th>D</th>
<th>Q</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>x</td>
<td>$Q^+$</td>
<td>Storage State</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>Transparent Mode</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>Transparent Mode</td>
</tr>
</tbody>
</table>
**D Flip-Flop**

Synchronous (also know as Master-Slave FF)

Edge Triggered (data moves on clock transition)
- one latch transparent - the other in storage
- active low latch followed by active high latch
- positive edge triggered (rising edge of CK)

active high latch followed by active low latch
- negative edge triggered (falling edge of CK)
Timing Considerations

Set-up time ($t_s$) = minimum time data must be valid before active edge of clock

Hold time ($t_h$) = minimum time data must be held valid after active edge of clock

Clock-to-output delay ($t_{do}$) = maximum time before output data is valid with respect to active edge of clock

Set-up or Hold Time violation => metastability (Q & Q̅ go to intermediate voltage values which are eventually resolved to an unknown state)

Set-up & Hold Time violations in a vector set referred to as clock-data races
Good Design Practices

Use single clock, single edge synchronous design techniques as much as possible

Asynchronous interfaces lead to metastability (minimize the async interface & double clock data to reduce probability of metastability)

Avoid asynchronous presets & clears on FFs (use sync presets & clears whenever possible)

DO NOT construct a FF from two level sensitive latches of the same type with an inverter on the clock input to one latch

DO NOT gate clocks!!!
Create clock enabled FFs via a MUX to feed back current data

Active high clock enable (CEN)
Transmission Gate Latches/FFs

A single NFET (PFET) looks like a active high (low) level sensitive dynamic latch (storage mechanism is capacitive load)

But NFET (PFET) passes a poor logic 1 (0) with both, enable inverter can cause timing problems inverter on Q gives consistent capacitive load $C_L$

Dynamic FF made with 2 latches

falling edge triggered dynamic FF

Dynamic latches/FFs require data refresh
Transmission Gate Latches/FFs

Static FFs & Latches require feedback

Cross-coupled gate FFs are static (like OAI21s & AOI21s - data lasts indefinitely)

Static T-gate Latch => extra inverter & T-gate

Static T-gate FF made with 2 latches

T-gate based latches/FFs require careful design, layout, and simulation to ensure proper operation (no timing problems)