Storage Elements and Memory
## R-S Latch

### Circuit Diagram
![R-S Latch Circuit Diagram](image)

### Truth Tables

<table>
<thead>
<tr>
<th>R</th>
<th>S</th>
<th>a</th>
<th>b</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>?</td>
<td>?</td>
</tr>
</tbody>
</table>

### Truth Table for Output a

<table>
<thead>
<tr>
<th>A</th>
<th>R</th>
<th>S</th>
<th>a</th>
<th>b</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

### Truth Table for Output b

<table>
<thead>
<tr>
<th>A</th>
<th>R</th>
<th>S</th>
<th>a</th>
<th>b</th>
</tr>
</thead>
<tbody>
<tr>
<td>x</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>x</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>x</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

### Additional Cases

- For R-S Latch, when R = 1 and S = 1, the output is indeterminate (indicated by ?).

### Notes

- The R-S Latch is a type of flip-flop used in digital electronics to store binary information.

---

University of Texas at Austin    CS310H - Computer Organization    Spring 2010    Don Fussell  2
D Latch

Note: a **latch** is a **level-triggered** storage element. Level-triggered means the stored data can be changed when the clock is at a specified voltage.
D Flip-flop

Note: a flip-flop is an edge-triggered storage element. Edge-triggered means the stored data can be changed when the clock changes voltage, either from low to high (as here) or high to low.
Edge vs. level triggering

D latch

D flip-flop

WE

D

p

q

Time

1

0

1

0

1

0

1

0

1

0

1
Cheaper Memory

- 8+ CMOS transistors per bit is too much for large scale memory (okay for registers, although we can trim the transistor count just a bit)
- 1 transistor per bit DRAM uses a single transistor to trap charge on a capacitor ("dangling wire"), and then read out whether or not there’s charge there. The charge leaks out after a while, so it needs to be refreshed, and it’s pretty slow since it needs lots of amplification.
- Called dynamic memory since it needs to be refreshed (as opposed to self-amplifying, static memory).
Static bit in CMOS

- This is 8 transistors (verify). Can we do better?
- Let’s “cheat” and do our designs with transistors as well as Boolean logic
- Now we only need 6 transistors, and it gives us a D latch (which took 16 or 18 the Boolean way)
  - Actually, the 6 transistor design you’ll see in normal use doesn’t quite work like this, but it’s close.
Memory from bits

A 4-bit register using D latches
Memory organization

- Address space = 4
- Addressability = 3 bits
- The leftmost pink box is a decoder
- What are the other 3 pink boxes?
Memory organization

- Address space = 4
- Addressability = 3 bits
- The leftmost pink box is a decoder
- What are the other 3 pink boxes?
- Reading word 3