Modular Sequential Circuits

- Many useful sequential circuits can be built from flip-flops:
  - Shift registers
    - (left shift, right shift, parallel-in serial out, serial-in parallel out, etc.)
  - Counters
    - (binary counters, up/down counters, etc.)
  - Modulo-N counters
  - Binary rate multipliers
SHIFT REGISTERS

Shift Registers

• Sequential logic module, containing \( n \) flip-flops (\( n \) bits), where every clock cycle shifts the bits stored to the left or right.

• Serial-in, serial-out shift register:
Generic Shift Register

- Four modes of operation:
  - S-in S-out, S-in P-out, P-in S-out, P-in P-out but shifting in between

![Diagram of n-Bit shift register with labels for Parallel in, Parallel out, Serial in, Serial out, Preset control, Shift pulse, Clear control]

Standard Shift Registers

- MSI shift registers, among others:

<table>
<thead>
<tr>
<th>Device</th>
<th>Features</th>
</tr>
</thead>
<tbody>
<tr>
<td>7491A</td>
<td>8-bit, S-in-out</td>
</tr>
<tr>
<td>74164</td>
<td>8-bit, S-in-out, P-out, Clear</td>
</tr>
<tr>
<td>7496</td>
<td>5-bit, S-in-out, async-preset, P-out, Clear</td>
</tr>
<tr>
<td>74165</td>
<td>8-bit, S-in-out, async-load, clock inhibit</td>
</tr>
<tr>
<td>74179</td>
<td>4-bit, S-in-out, Clear, sync-load, P-out</td>
</tr>
<tr>
<td>74194</td>
<td>4-bit, bidirectional, S-in-out, sync-load, P-out, clock inhibit, Clear</td>
</tr>
</tbody>
</table>
7491A 8-bit, S-in-out

74164 S-in-out, P-out, Clear
7496 5-bit, S-in-out, async-preset, P-out, Clear

74165 8-bit, S-in-out, async-load, clock inhibit
74179 4-bit, S-in-out, Clear, sync-load, P-out

74194 4-bit, bidirectional, S-in-out, sync-load, P-out, clock inhibit, Clear
Counters

- Input pulses make a (binary) code increment.
- Can be used for timers (slowing down clocks).
- Example MSI counters (among others):

<table>
<thead>
<tr>
<th>Device</th>
<th>Type</th>
<th>Features</th>
</tr>
</thead>
<tbody>
<tr>
<td>74163</td>
<td>Sync, binary</td>
<td>4-bit, sync-load, sync-clear, enable, carry-out</td>
</tr>
<tr>
<td>74160</td>
<td>Sync, decade</td>
<td>4-bit, sync-load, async-clear, enable, carry-out</td>
</tr>
<tr>
<td>74191</td>
<td>Sync, up/down</td>
<td>4-bit, async-load, enable, max out, carry out</td>
</tr>
<tr>
<td>74293</td>
<td>Async, binary</td>
<td>/2, /8, /16, common clear</td>
</tr>
<tr>
<td>74177</td>
<td>Async, binary</td>
<td>/2, /8, /16, common clear, async-load</td>
</tr>
</tbody>
</table>
Synchronous Binary Counter

- J-K realization:

74163 Binary, Sync-load
Asynchronous Binary Counter

- Asynchronous does not refer to not needing a counting clock, it refers to the internal architecture.

![Asynchronous Binary Counter Diagram]

74293 Async-Counter

![74293 Async-Counter Diagram]
Down Counter

- A.k.a. backward counter:

- Recall, the forward counter:

Up/Down Counter
Modulo-N Counters

- Counts from state 0 to state N and then resets to zero and counts on.
- N does not have to be a two-power!
- What is a BCD counter then?
- Can we make a counter that counts to 5, how about 6 or 12?
- Counters perform integer division! (except very slowly, as the dividend has to feed a clock, the divisor is the “N”, and the quotient is the count of the overflows (needs a second counter) and the modulo is the (first) counter’s output.

Powerline Digital Timer

![Diagram of Powerline Digital Timer]
SHIFT REGISTERS AS COUNTERS

Ring Counters

- Counting to 5 (5-bit ring counter):
Twisted Ring Counter

Selecting Clock Impulses (Timing)

• Using a (twisted) ring counter
FRACTIONAL RATE MULTIPLIERS

Binary Rate Multiplier

- We have seen how to divide down the number of clock pulses by an integer. How do we divide (multiply) it by a fraction?
- The number of output pulses should be a (binary) fraction of the number of input pulses: \( N_o = N_i B / 2^n \)
74167 Decade Fractional Rate Multiplier

• $N_o = N_i * B / 10$

Fractional Rate Multiplier Cascading

• How do we get other fractions? E.g., $N_o = N_i * 63 / 320$
• $N_o = N_i * (7/10) * (18/64)$
Summary

- We looked at some standard sequential modules that are easy to build using flip-flops:
  - Shift registers
  - Counters
  - Shift registers as ring counters
  - Counters as fractional clock rate multipliers