Design Example: Counters

- **Counter**: a sequential circuit that repeats a specified sequence of output upon clock pulses.
  - A, B, C, ..., Z.
  - 0, 1, 2, 3, 4, 5, 6, 7.
  - 7, 6, 5, 4, 3, 2, 1, 0.
  - 0, 1, 0, 1, 2, 3, 2, 3, 2, 1.

- **Binary counter**: follows the binary sequence.
  - 2-bit (up) binary counter: 0, 1, 2, 3.
  - 3-bit down binary counter: 7, 6, 5, 4, 3, 2, 1, 0

- **Other useful counters**:
  - Decimal counter (e.g. BCD counter)
    - 0000, 0001, 0010, ..., 1001, 1000, 0001, ...
  - Modulo-k counter
    - Modulo-5 counter: 0, 1, 2, 3, 4, 0, 1, 2, ...
  - M-to-N counter
    - 3-to-8 counter: 3, 4, 5, 6, 7, 8, 3, 4, ...
  - Ripple counter
  - Ring counter and Johnson counter

### 3-Bit Binary Counter

#### current state | next state | flip-flop inputs
<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>TA</th>
<th>TB</th>
<th>TC</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<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>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

- **Figure 6.32**: gives a 4-bit binary counter. When $Q_3 = C$, $Q_4 = B$, $Q_2 = A$, delete $Q_3$, set Count enable bit to be constant 1, it becomes this 3-bit counter.
Binary Counter with JK Flip-Flops

- when $E = 1$ and CP goes from $0$ to $1$:
  - $A_1$
  - $A_2$
  - $A_3$
  - $A_4$

Count-Down Binary Counter

- Exercise:
  Verify that the circuit is a binary counter that counts down from 15 to 0, and then back to 15 again.

Summary

- Sequential circuit design example
  - Shift registers
  - Basic counters
- Next time
  - Binary counter with parallel load
  - Ripple counter
  - Johnson counter
- Next Monday: Exam IV
  - PLD, Chapter 6, 7.1, 7.2
  - Discussion on Wednesday’s class

Binary Counter with Parallel Load

- 2 control signals, 3 modes:
  - Load ($Q_i = D_i$) Load = 1
  - Count (up) Load = 0, Count = 1
  - No change Load = 0, Count = 0
    (change, load or count, happens only at positive edge of the clock pulse.)
- Carry Out: 1 if and only if the counter is in “count” mode with content 1111.
- Read Figure 6.34 for the detailed implementation.
Design Example

8-bit counter with two 4-bit counters

Exam IV

- PLD
- Timing diagram for basic latch/flip-flop
- Sequential circuit analysis
- Sequential circuit design
- Registers and counters
Ripple Counter

- Synchronous counter: the CP signal of all flip-flops are from the common clock.
- Ripple counter: the CP of some flip-flops are from other flip-flops (and through logic gates).
  - Ripple counter is asynchronous
  - Binary ripple (up) counter (read Figure 6.31)
  - Binary ripple down counter
  - Where the CP signal comes from? (By default, flip-flop is positive edge triggered.)

BCD Ripple Counter

- Verify the following circuit is a BCD Ripple counter triggered by negative edge.

Ring Counter

- Ring counter: a circular shift register (with \( k \) flip-flops) that at any time, only one flip-flop is set (having value 1) and all others are cleared (with value 0). It is used to generate \( k \) (periodic) timing signals.
- Example: see Figure 6.37 for circuit.
Ring Counter as “Counter + Decoder”

- To generate 4 (periodic) timing signals, we need
  - a 4-bit ring counter,
  - or a 2-bit counter
    and a 2x4 decoder.

![Ring Counter Diagram]

Johnson Counter

- **Johnson counter**: a k-bit circular shift register with
  the complement of the last flip-flop connected to the input
  of the first flip-flop, and 2k decoding gates. It is used to
  generate 2k (periodic) timing signals.

![Johnson Counter Diagram]

<table>
<thead>
<tr>
<th>States</th>
<th>AND gate for output</th>
</tr>
</thead>
<tbody>
<tr>
<td>S1, S2, S3, T1, T2, T3, T4, T5, T6</td>
<td></td>
</tr>
<tr>
<td>1, 0, 0, 0, 1</td>
<td></td>
</tr>
<tr>
<td>2, 1, 0, 0, 1</td>
<td></td>
</tr>
<tr>
<td>3, 1, 1, 0, 1</td>
<td></td>
</tr>
<tr>
<td>4, 1, 1, 1, 1</td>
<td></td>
</tr>
<tr>
<td>5, 0, 1, 1, 1</td>
<td></td>
</tr>
<tr>
<td>6, 0, 0, 1, 1</td>
<td></td>
</tr>
</tbody>
</table>

6 AND gates for decoding

S1, S2, S3