Ensembles of flip-flops

- Registers
- Shift registers
- Counters

Registers

- Collections of flip-flops with similar controls and logic
  - stored values somehow related (for example, form binary value)
  - share clock, reset, and set lines
  - similar logic at each stage
- Examples
  - shift registers
  - counters
Shift register

- Holds samples of input
  - store last 4 input values in sequence
  - 4-bit shift register:

```
IN  D  Q  D  Q  D  Q  D  Q  OUT1  OUT2  OUT3  OUT4
CLK
```

Universal shift register

- Holds 4 values
  - serial or parallel inputs
  - serial or parallel outputs
  - permits shift left or right
  - shift in new values from left or right

```
input
left_in
left_out
clear
s1

output
right_out
right_in
clock

s0 s1 function
0 0  hold state
0 1  shift right
1 0  shift left
1 1  load new input

clear sets the register contents and output to 0
s1 and s0 determine the shift function
```
Design of universal shift register

- Consider one of the 4 flip-flops
  - new value at next clock cycle:

<table>
<thead>
<tr>
<th>clear</th>
<th>s0</th>
<th>s1</th>
<th>new value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>-</td>
<td>-</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>output</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>output value of FF to left (shift right)</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>output value of FF to right (shift left)</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>input</td>
</tr>
</tbody>
</table>

- Autumn 2010
- CSE370 - XV - Registers and Counters

Shift register application

- Parallel-to-serial conversion for serial transmission

- Autumn 2010
- CSE370 - XV - Registers and Counters
Pattern recognizer

- Combinational function of input samples
  - in this case, recognizing the pattern 1001 on the single input signal

![Pattern recognizer diagram]

Counters

- Sequences through a fixed set of patterns
  - in this case, 1000, 0100, 0010, 0001
  - if one of the patterns is its initial state (by loading or set/reset)

![Counters diagram]
Activity

- How does this counter work (assuming it starts in state 0000)?

 Binary counter

- Logic between registers (not just multiplexer)
  - XOR decides when bit should be toggled
  - always for low-order bit, only when first bit is true for second bit, and so on
How fast is our counter?

- Period > \( T_{\text{propFF}} + T_{\text{propCL}} + T_{\text{setupFF}} \)
- Period > \( 3.6\text{ns} + \max(T_{\text{propCL}}) \) (over all paths from Qs to Ds) + 1.8ns
- Period > \( 5.4\text{ns} + T_{\text{propXOR}} + T_{\text{propAND3}} \)
- Frequency < \( 1/\text{Period} \)

Four-bit binary synchronous up-counter

- Standard component with many applications
  - positive edge-triggered FFs w/ synchronous load and clear inputs
  - parallel load data from D, C, B, A
  - enable inputs: must be asserted to enable counting
  - RCO: ripple-carry out used for cascading counters
    - high when counter is in its highest state 1111
    - implemented using an AND gate

(1) Low order 4-bits = 1111
(2) RCO goes high
Eight-bit counter

- Use of enable to allow high-order four bits to count
  - Always enable 4 low-order bits
  - Enable 4 high-order bits only when RCO is high for low-order bits (reached 15)

Offset counters

- Starting offset counters
  - use of synchronous load
  - e.g., 0110, 0111, 1000, 1001, 1010, 1011, 1100, 1101, 1111, 0110, ...

- Ending offset counter
  - comparator for ending value
  - e.g., 0000, 0001, 0010, ..., 1100, 1101, 0000

- Combinations of the above (start and stop value)

- Speed?
Sequential logic and programmable logic

- FFs added to PLAs, PALs, and ROMs
- FFs added to FPGAs
- Other features
  - Feedback of Qs back in as inputs
  - Multiplexor for selecting combinational or sequential function
  - Arithmetic mode in FPGAs
  - Many other options .....

The 22V10 PAL

- Combinational logic elements (SoP)
- Sequential logic elements (D-FFs)
- Up to 10 outputs
- Up to 10 FFs
- Up to 22 inputs

- A typical PAL
Implementation using PALs

- Programmable logic building block for sequential logic
  - macro-cell: FF + logic
    - D-FF
    - two-level logic capability like PAL (e.g., 8 product terms)

22V10 PAL Macro Cell

- Sequential logic element + output/input selection
Cyclone II FPGA Logic Array Blocks (LABs)

Cyclone II FPGA Logic Element (LE)
(16 per LAB)
Logic Element (LE) in Normal Mode

Logic Element (LE) in Arithmetic Mode