About Clocking

- Clock source: 100MHz
- ~ 0.1 seconds: clk/2^23 (0.08 seconds)
- ~ 1 seconds: clk/2^27 (1.34 seconds)
- Common guideline
  You may use the same clock rate (e.g., clk/2^13 – clk/2^17) for
  - Debounce circuits
  - One-pulse circuits
  - 7-segment display

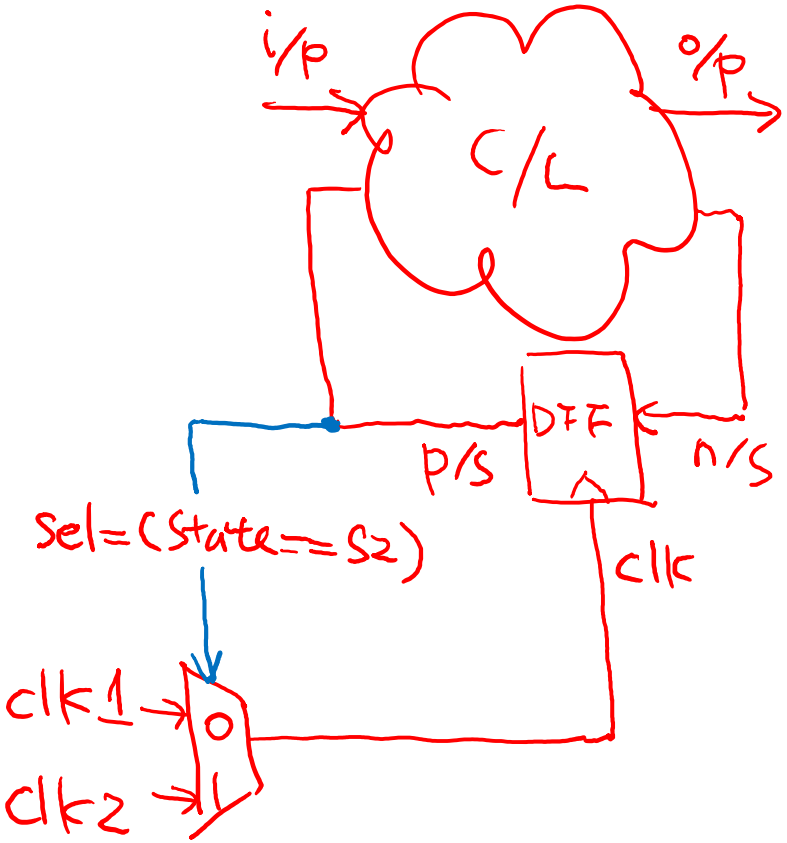
- Using 100MHz for debounce circuits?

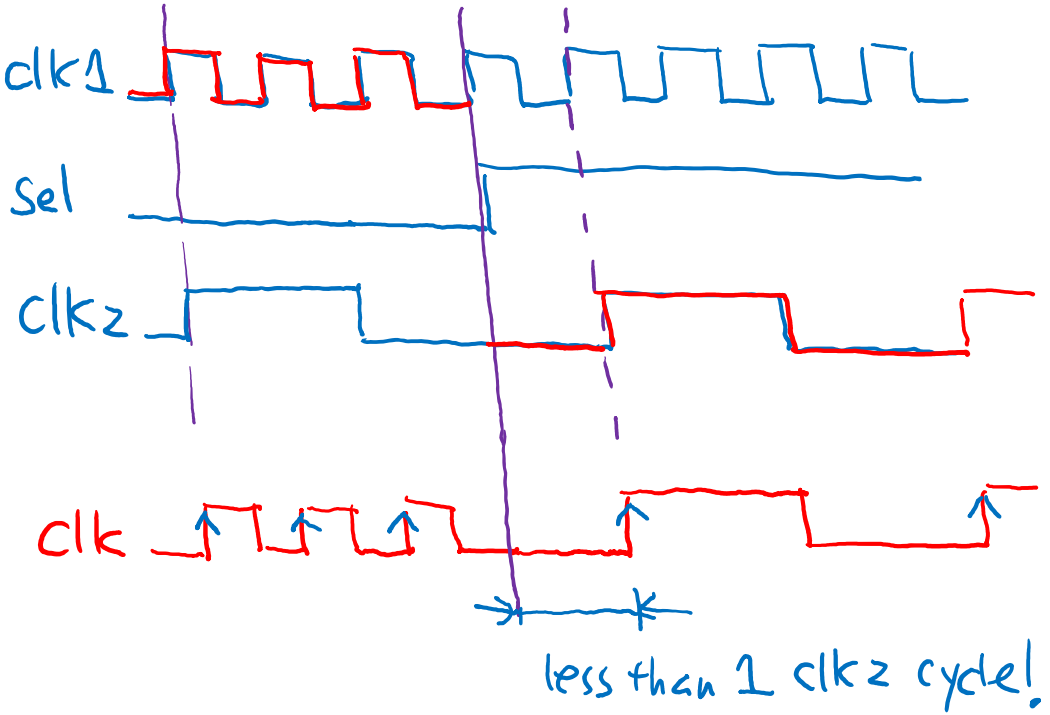
About Manipulating Two or More Clocks

- Method 1
  - You may operate at a higher clock rate and use a counter to slow down the display rate. (e.g., the FSM can operate at the clock rate of clk/2^16.
  - But in some states, it may count for 2^11 cycles for each state transition to behave as the clock rate of clk/2^27).
- Method 2
  - Or you may use a clock selector with multiple clock sources as a quick fix.
    Usually, it also works.

About Manipulating Two or More Clocks Method 2

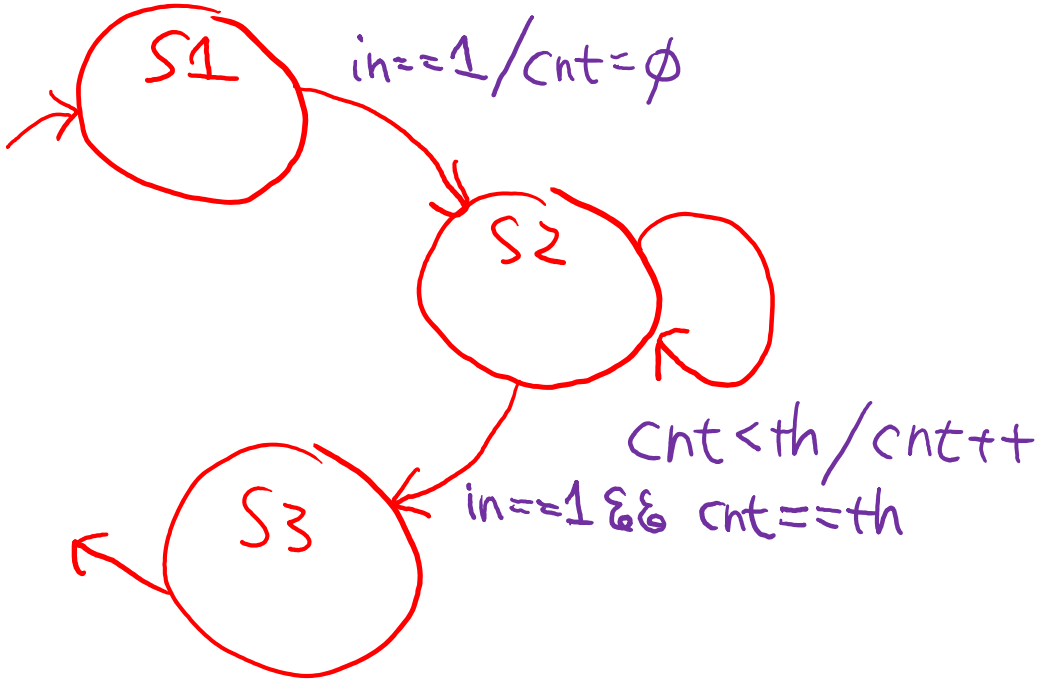
- Using clock selector for multiple clocks





About Manipulating Two or More Clocks Method 1

- Single synchronous clock with counter(s)



Coding Example

```
module example(
  input clk,
  input rst,
  output [15:0] LED
);
  parameter sec = 100000000 - 1;

  reg [31:0] counter;
  reg [31:0] counter_next;

  reg led_state;
  reg led_state_next;

  assign LED = {16{led_state}};
```

```
  always @(posedge clk, posedge rst) begin
    if (rst == 1) begin
      counter <= 0;
      led_state <= 0;
    end else begin
      counter <= counter_next;
      led_state <= led_state_next;
    end
  end

  always @(*) begin
    if (counter == sec) begin
      counter_next = 0;
      led_state_next = ~led_state;
    end else begin
      counter_next = counter + 1;
      led_state_next = led_state;
    end
  end
endmodule
```