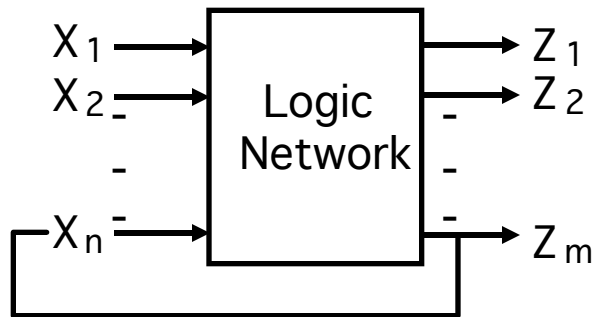


Combinational vs. Sequential Logic

❖ Readings: 5-5.4.4

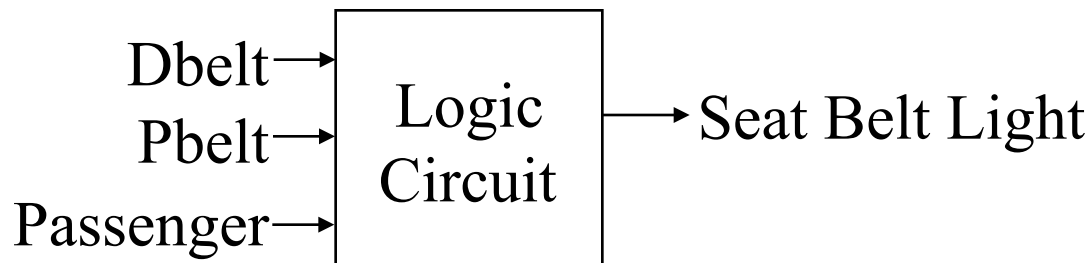


Network implemented from logic gates. The presence of feedback distinguishes between **sequential** and **combinational** networks.

Combinational logic

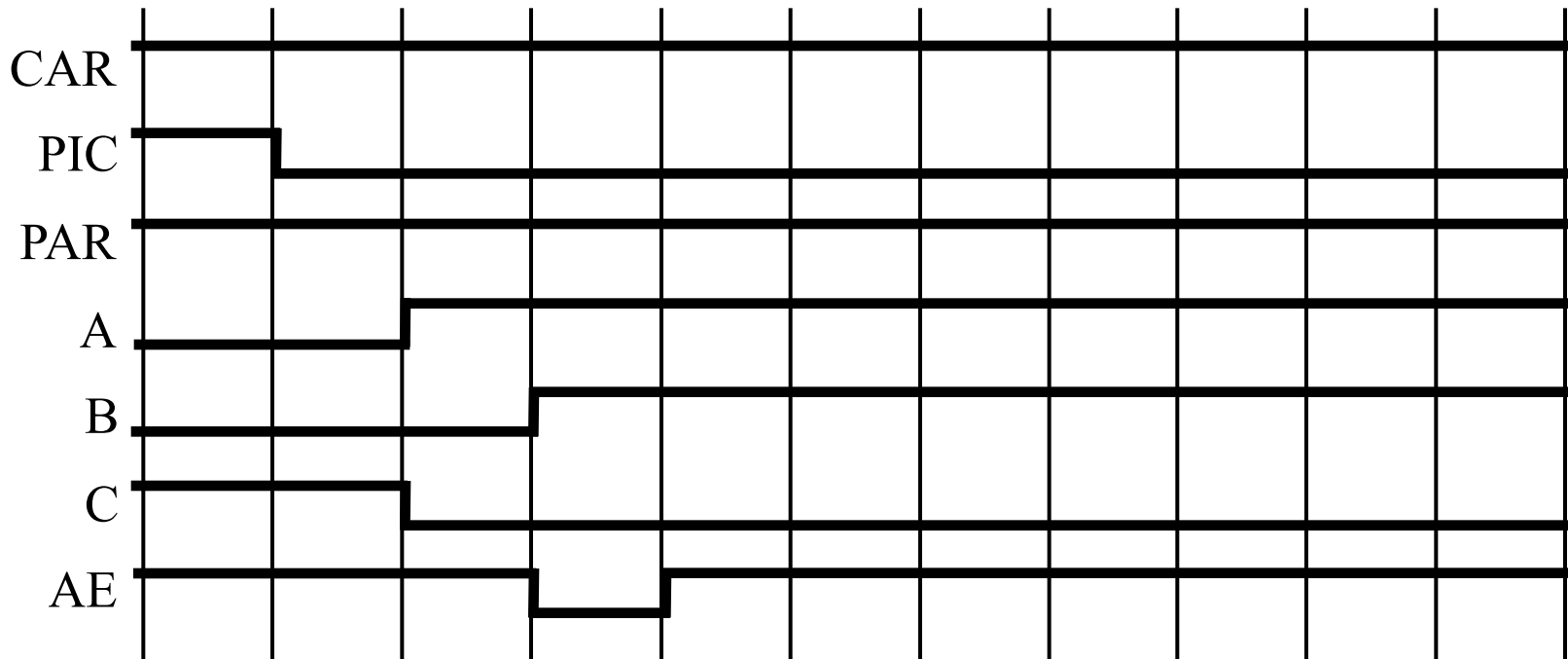
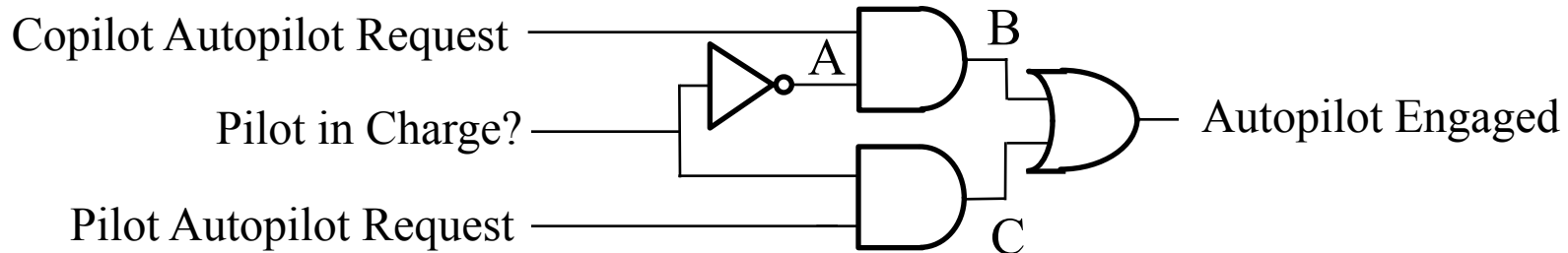
no feedback among inputs and outputs
outputs are a pure function of the inputs
e.g., seat belt light:

(Dbelt, Pbelt, Passenger) mapped into (Light)



Hazards/Glitches

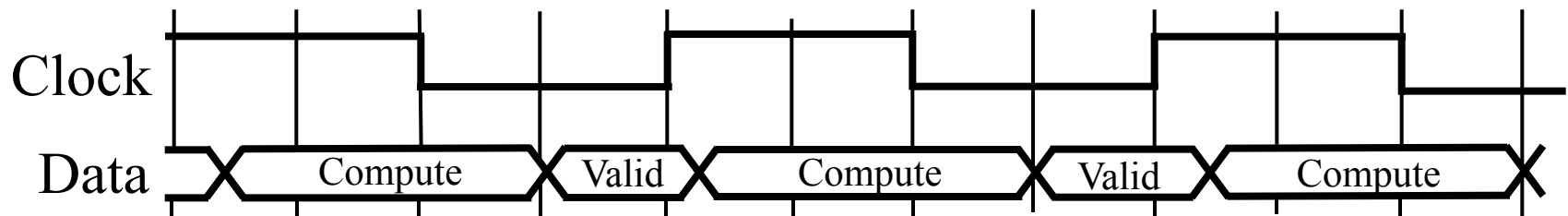
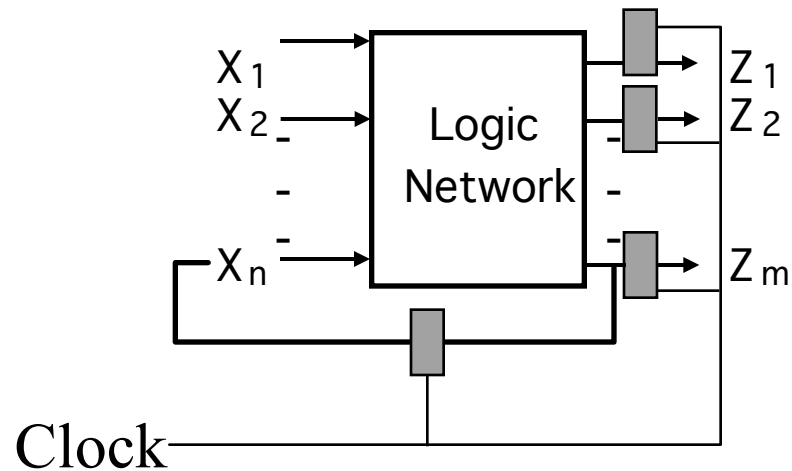
- Circuit can temporarily go to incorrect states



❖ Must filter out temporary states

Safe Sequential Circuits

- Clocked elements on feedback, perhaps outputs
 - Clock signal synchronizes operation
 - Clocked elements hide glitches/hazards

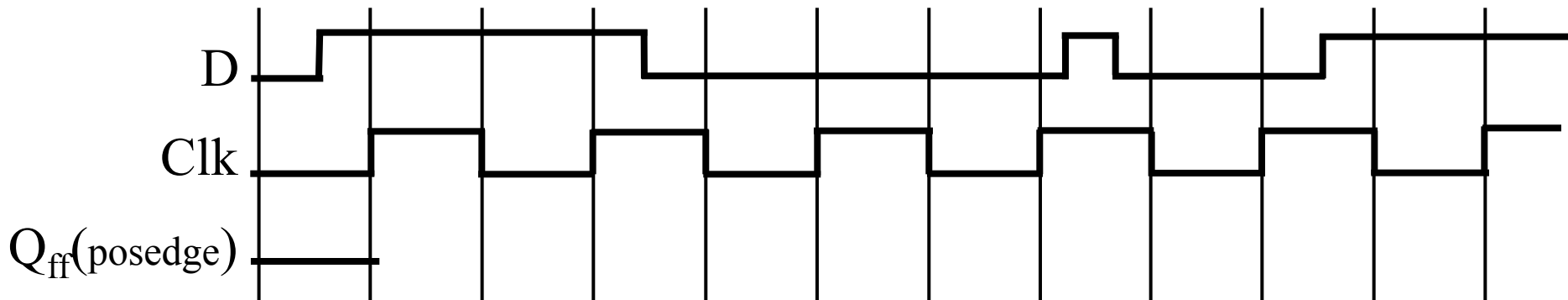
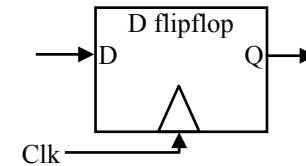


Basic D Flip Flop

```
// Basic D flip-flop
```

```
module basic_D_FF (q, d, clk);  
    output q;  
    input d, clk;  
    reg q; // Indicate that q is stateholding
```

```
    always @(posedge clk)  
        q <= d; // ALWAYS use <= to assign to clocked elements  
endmodule
```



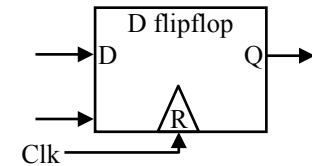
D Flip Flop w/Synchronous Reset

```
// D flip-flop w/synchronous reset

module D_FF (q, d, reset, clk);
    output q;
    input d, reset, clk;
    reg q; // Indicate that q is stateholding

    always @(posedge clk)
        if (reset)
            q <= 0; // On reset, set to 0
        else
            q <= d; // Otherwise out = d

endmodule
```



Verilog Testbench

```
module stimulus;
  reg clk, reset, d;
  wire q;

  parameter ClockDelay = 100;

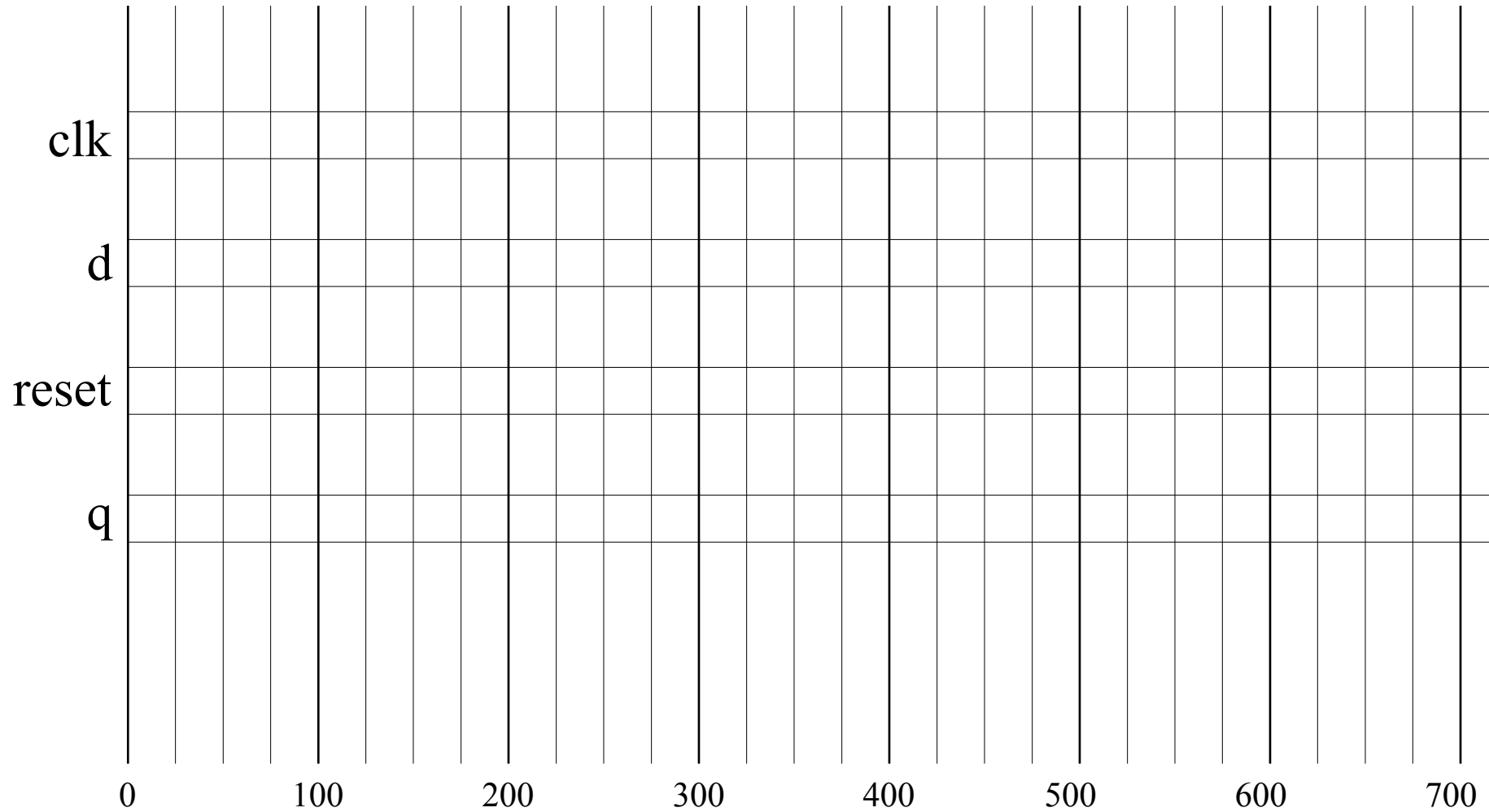
  D_FF dut (.q, .d, .reset, .clk); // Instantiate the D FF

  initial clk <= 0; // Set up the clock
  always #(ClockDelay/2) clk <= ~clk; // Toggle clock every 50 time units

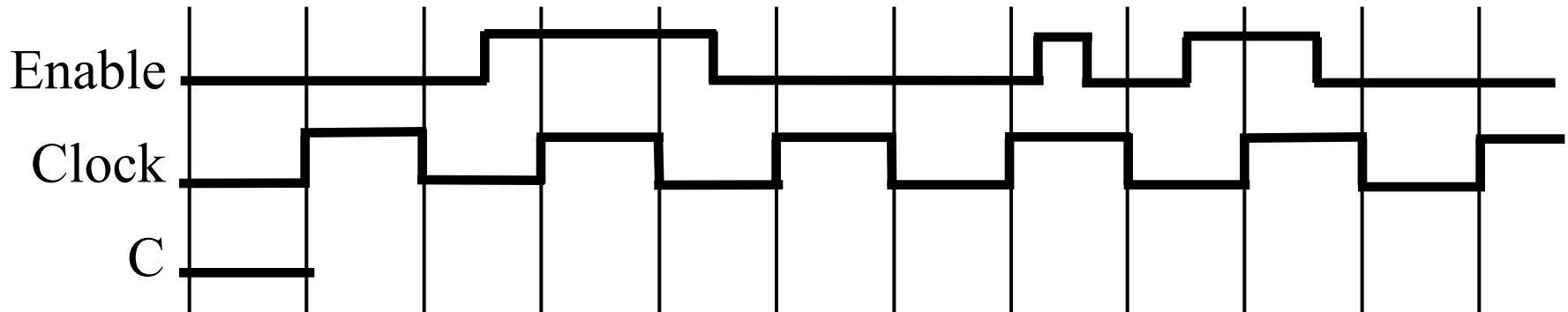
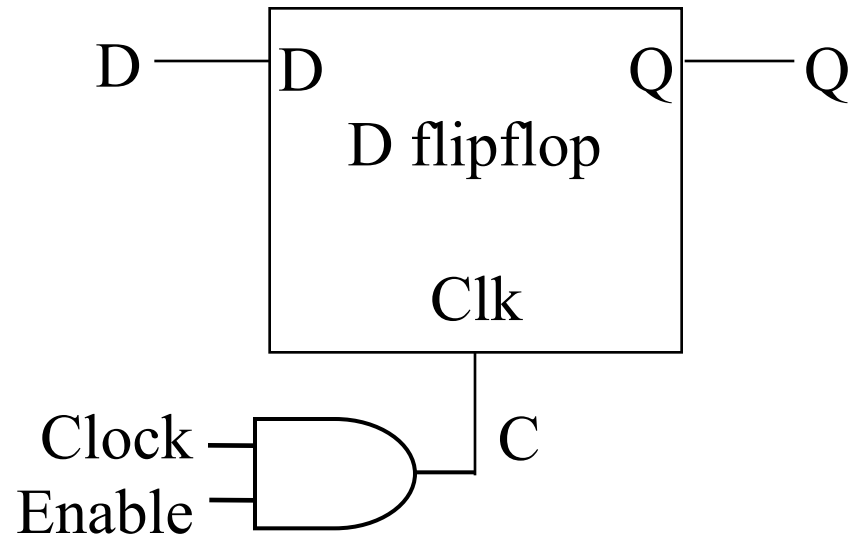
  initial // Set up the reset signal
  begin
    d <= 0; reset <= 1; @(posedge clk);
    reset <= 0; @(posedge clk);
    d <= 1;          @(posedge clk);
    d <= 0;          @(posedge clk);
    d <= 1;          @(posedge clk);
    $stop(); // end the simulation
  end

endmodule
```

Testbench Waveforms

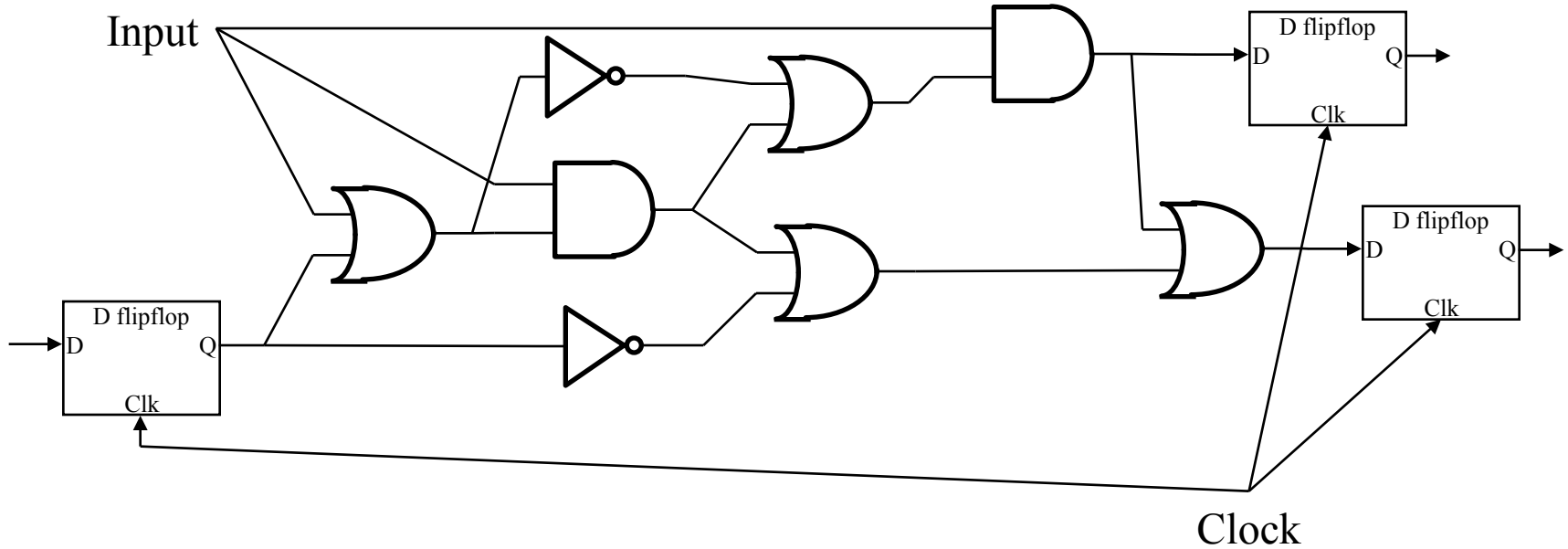


Flipflop Realities 1: Gating the Clock

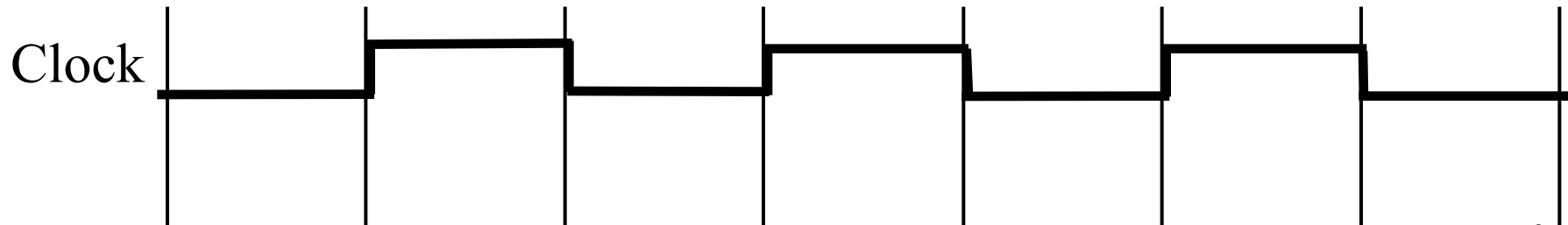


- **NEVER put a logic gate between the clock and DFF's CLK input.**

Flipflop Realities 2: Clock Period, Applying Stimulus

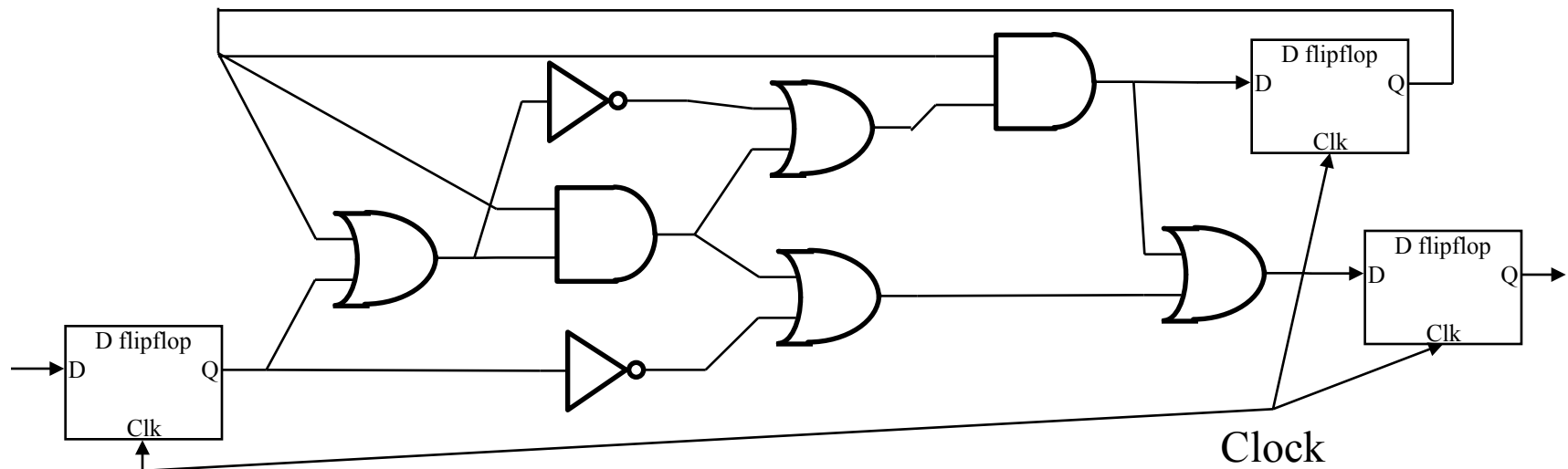
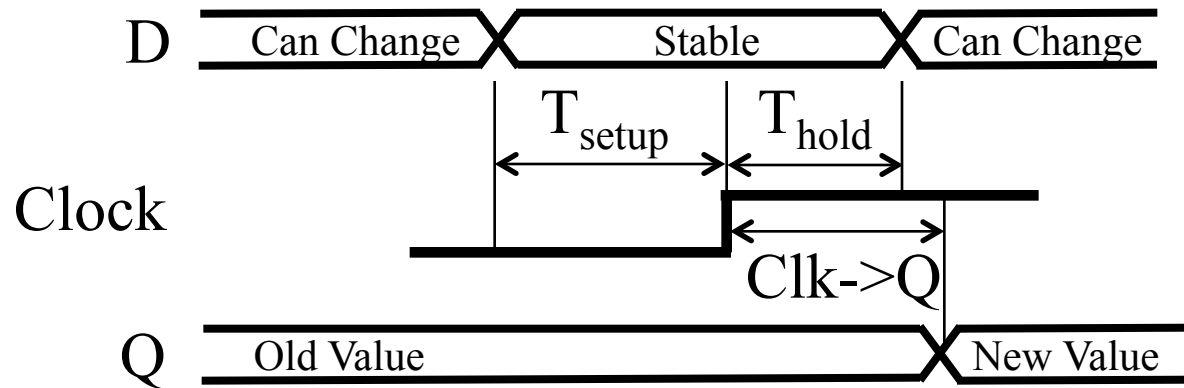


- Clock Period?
- Apply Inputs when?



T_{setup} , T_{hold} , Clk- \rightarrow Q

- Flipflops require their inputs be stable for time period around clock edge

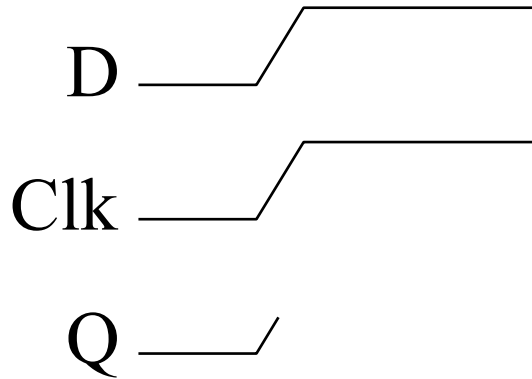


Timing Definitions

- T_{setup} : Time D must be stable BEFORE clock edge
 - Adds to critical path delay
- Clk->Q: Time from clock edge to Q changing
 - Adds to critical path delay
- T_{hold} : Time D must be stable AFTER clock edge
 - Sets minimum path from Q of one DFF to D of another

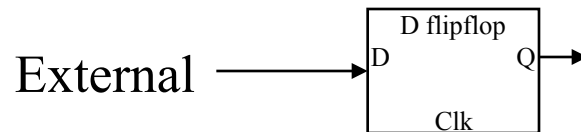
Flipflop Realities 3: External Inputs

- External inputs aren't synchronized to the clock

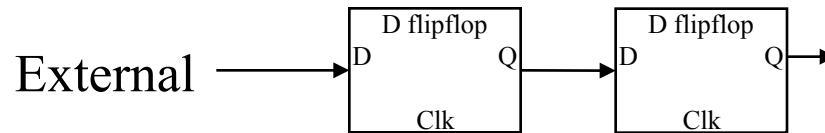


Dealing with Metastability

■ Single DFF



■ 2 DFFs in series



■ 2 DFFs in parallel

