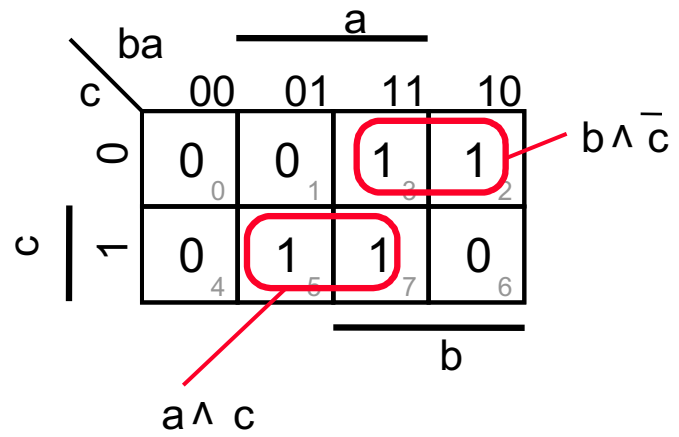
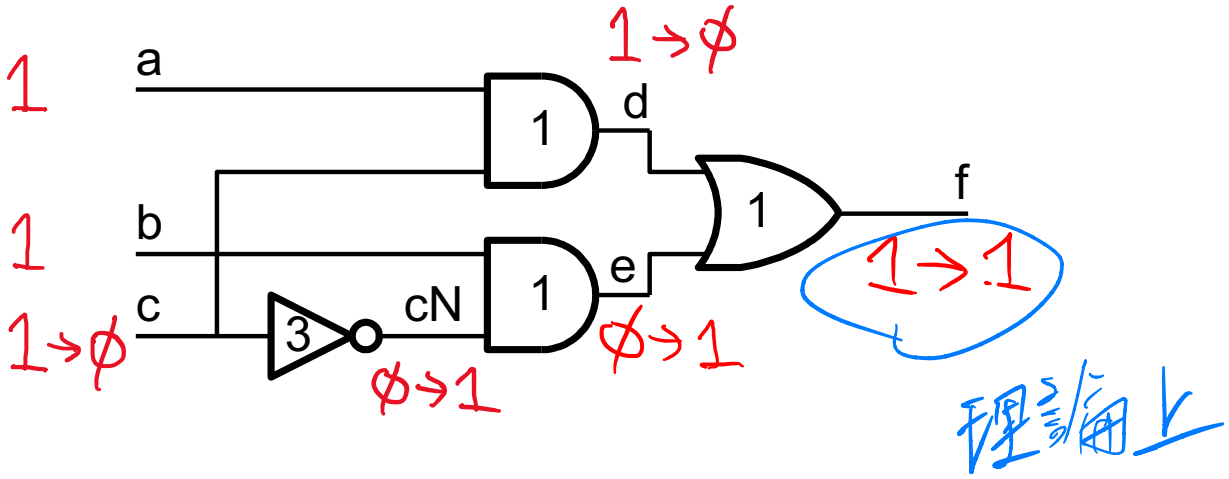
Hazards (or Glitches)

- Hazards: Unwanted switching transients appearing in the output of a combinational circuit while its inputs change
- Types of hazards (assuming only a single input can change at a time and no other input will change until the circuit has stabilized)

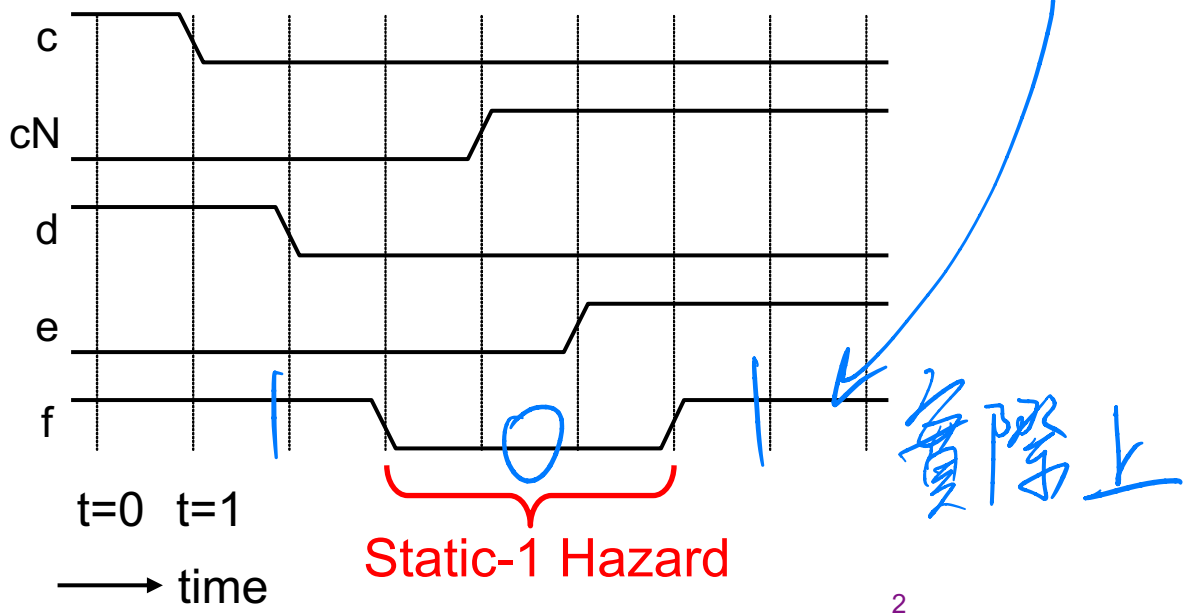
(a) Static 1-hazard (b) Static 0-hazard (c) Dynamic hazards

Hazards (cont)

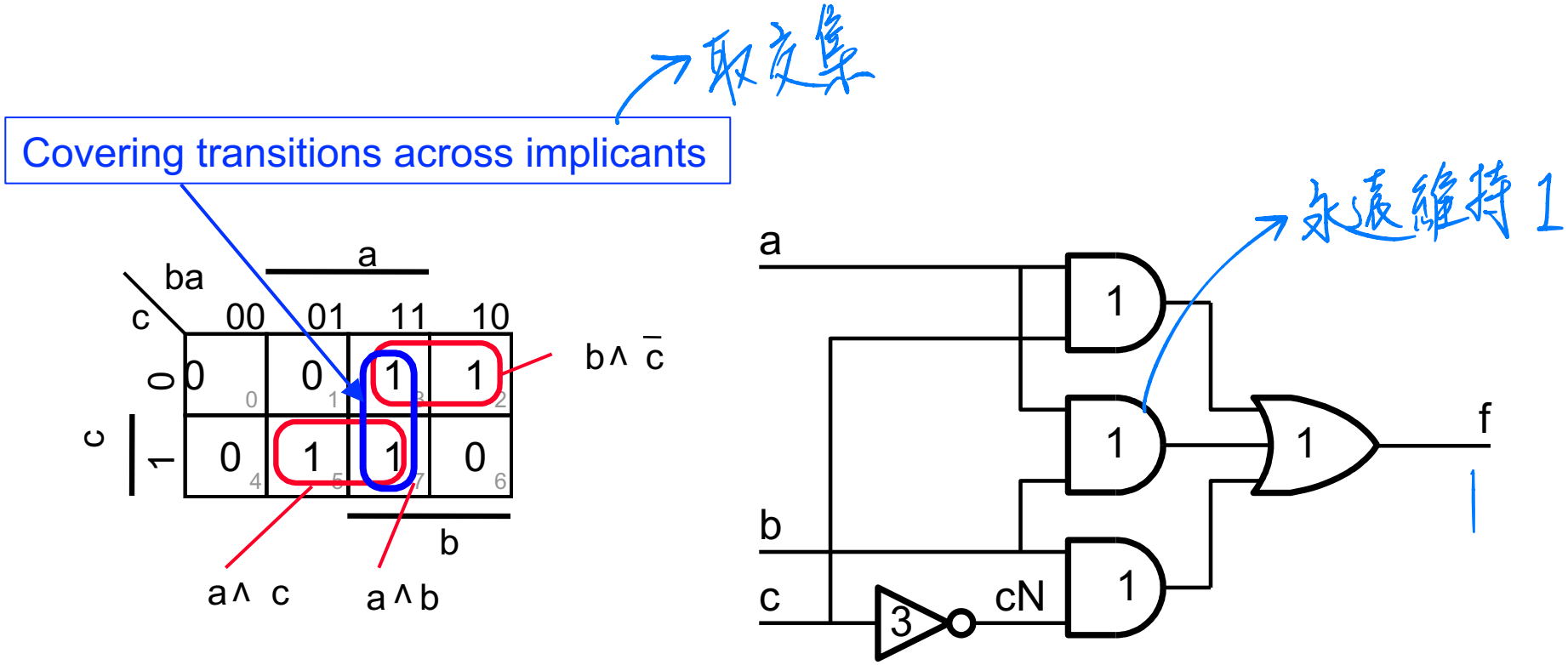




- Function output transits across two implicants
  → Possible hazards

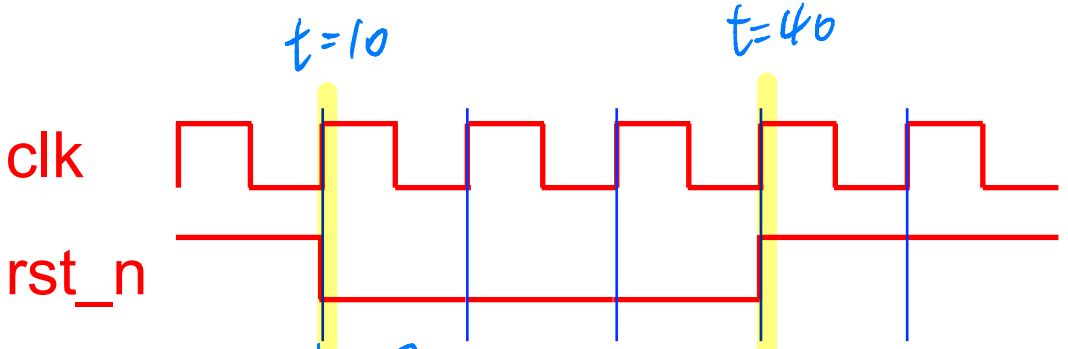


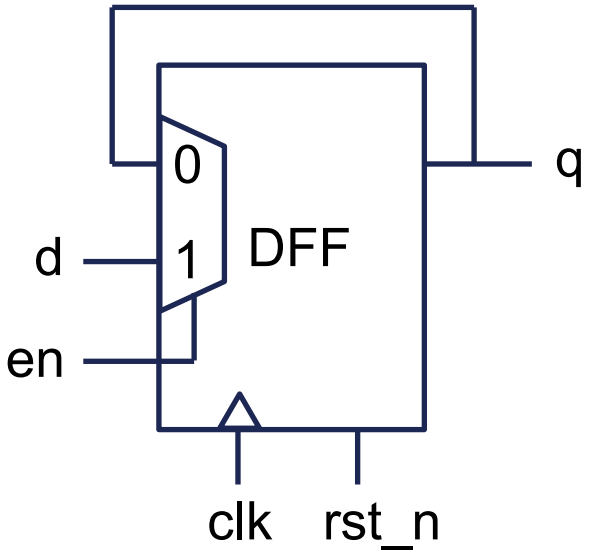
Hazard Eliminating



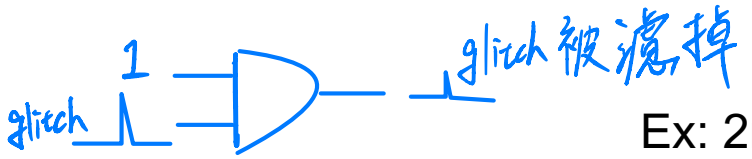
Verilog Discussion: Reset and Clock

```
always #5 clk = ~clk;
initial begin
  clk = 1;
  rst_n = 1;
  #10 rst_n = 0;
  #30 rst_n = 1;
  …
end
always @(posedge clk, negedge rst_n) begin
  if (rst_n == 1'b0) begin
    q <= 0;
  else if (en == 1'b1) begin
    d <= q;
  end
end
```



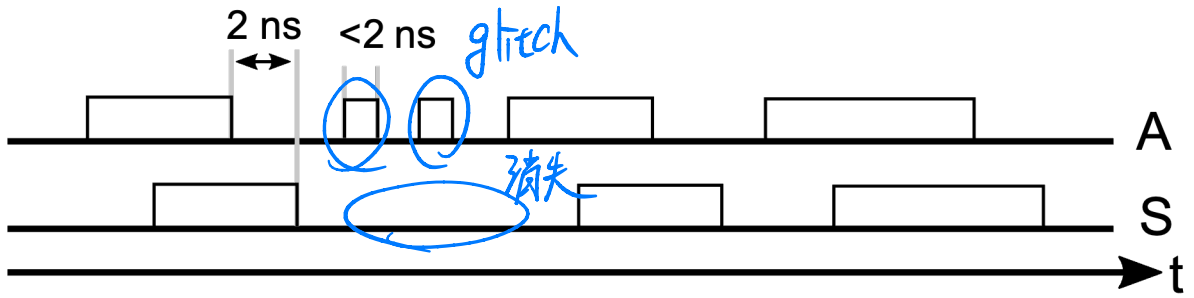


Inertial Delay and Transport Delay



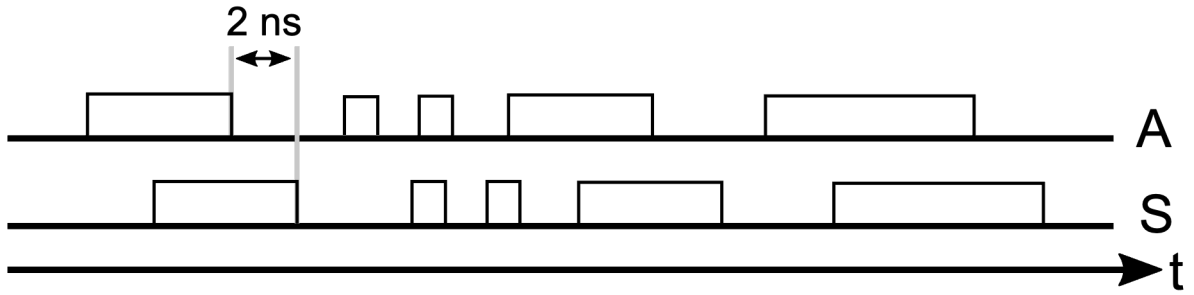
- Inertial delay
  - models the inherent latency introduced by logic gates
  - Signal transitions are only transferred when the new value remains constant for a minimum amount of time, i.e., spikes are suppressed

Ex: 2ns inertial delay



- Transport delay
  - models the inherent latency introduced by wires with propagation delay

Ex: 2ns transport delay



Verilog Discussion: Inertial Delay

- What happens with the following statement?

Continuous Assignment

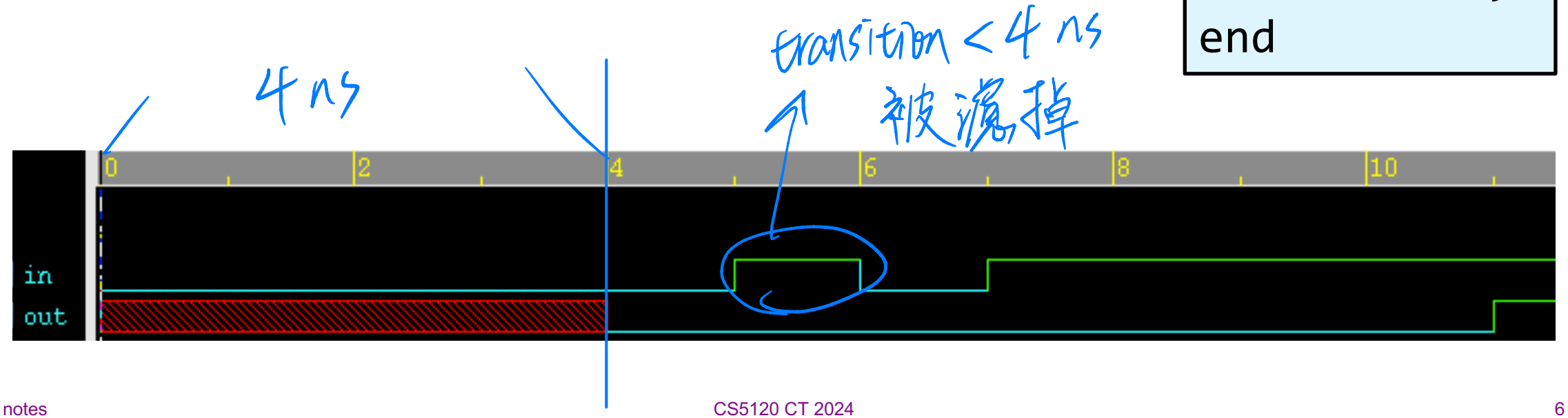
```
assign #4 out = in;
```

inertial delay model

✗

```
assign out = #4 in;
```

```
initial begin
      in = 0;
  #5 in = 1;
  #1 in = 0;
  #1 in = 1;
end
```



Verilog Discussion: Inertial Delay & Transport Delay

Procedural Assignment

```
reg b, a0, a1, a2, a3, a4;
always @(b) begin
  a0 = b;
end
```

```
#10 a1 = b;
a2 = #10 b;
#10 a3 <= b;
a4 <= #10 b;
```

Transport

```
assign a5 = b;
assign #10 a6 = b;
```

Inertial