

# Digital Logic Systems

## Chapter 17: Flip-Flops

Guy Even Moti Medina

School of Electrical Engineering Tel-Aviv Univ.

January 9, 2012

# Preliminary questions

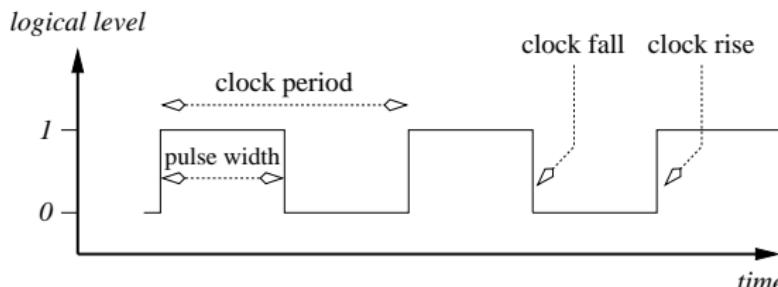
- 1 How is time measured in a synchronous circuit?
- 2 What is a “clock” in a microprocessor?
- 3 What is the frequency of a clock?
- 4 How are bits stored?
- 5 What is the functionality of a flip-flop?
- 6 What is a stable state? How many stable states does a flip-flop have?
- 7 How does a flip-flop move from one stable state to another?
- 8 How fast is this transition?

# The clock

the clock is generated by rectifying and amplifying a signal generated by special non-digital devices (e.g., crystal oscillators).

## Definition

A **clock** is a periodic logical signal that oscillates instantaneously between logical one and logical zero. There are two instantaneous transitions in every clock period: (i) in the beginning of the clock period, the clock transitions instantaneously from zero to one; and (ii) at some time in the interior of the clock period, the clock transitions instantaneously from one to zero.



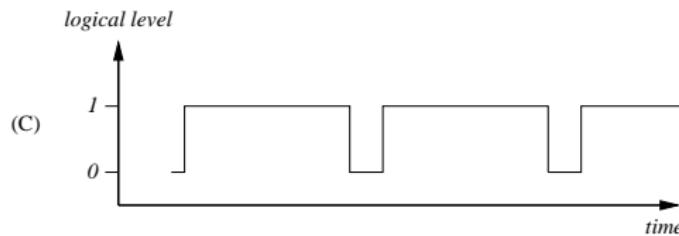
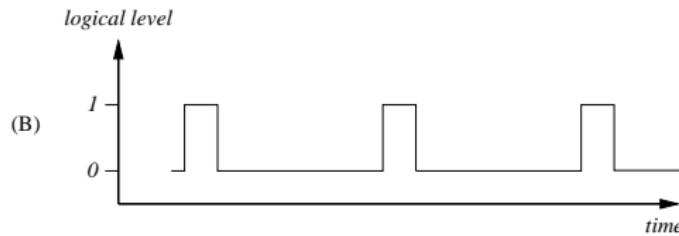
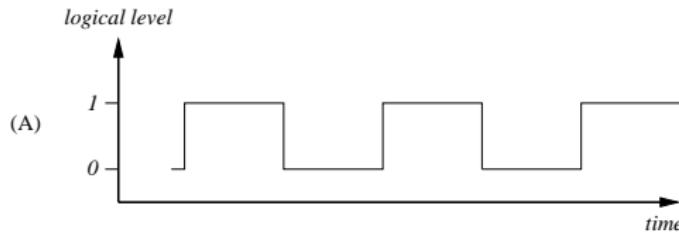


Figure: (A) A symmetric clock (B) narrow pulses (C) wide pulses.

# Clock cycles

- A clock partitions time into discrete intervals.
- $t_i$  - the starting time of the  $i$ th clock period.
- $[t_i, t_{i+1})$  -clock cycle  $i$ .

## Definition (edge-triggered flip-flop)

Inputs:  $D(t)$  and a clock CLK.

Output:  $Q(t)$ .

Parameters: Four parameters are used to specify the functionality of a flip-flop:

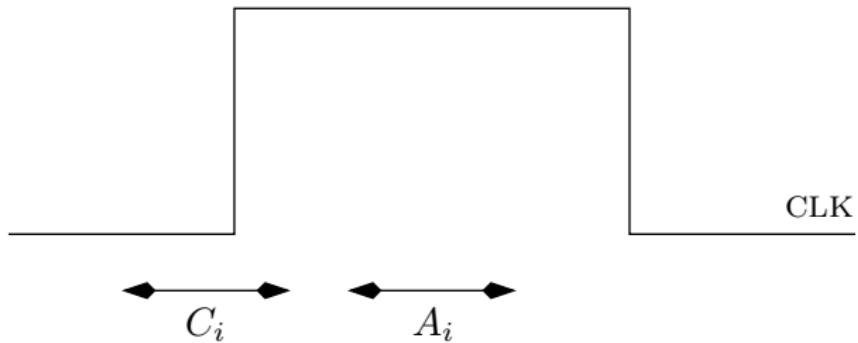
- **Setup-time** denoted by  $t_{su}$ ,
- **Hold-time** denoted by  $t_{hold}$ ,
- **Contamination-delay** denoted by  $t_{cont}$ , and
- **Propagation-delay** denoted by  $t_{pd}$ .

Terminology Require  $-t_{su} < t_{hold} < t_{cont} < t_{pd}$ .

- $C_i \stackrel{\triangle}{=} [t_i - t_{su}, t_i + t_{hold}]$  - the **critical segment**  $C_i$
- $A_i \stackrel{\triangle}{=} [t_i + t_{cont}, t_i + t_{pd}]$  - the **instability segment**

Functionality: If  $D(t)$  is stable during the critical segment  $C_i$ , then  $Q(t) = D(t_i)$  during the interval  $(t_i + t_{pd}, t_{i+1} + t_{cont})$ .

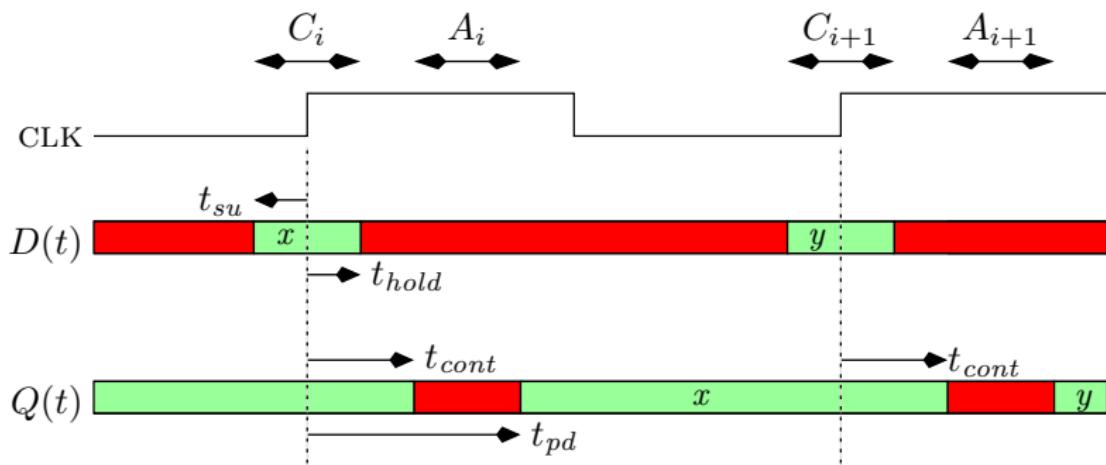
# Critical and instability segments in a flip-flop



**Figure:** The critical segment  $C_i = [t_i - t_{su}, t_i + t_{hold}]$  and instability segment  $A_i = [t_i + t_{cont}, t_i + t_{pd}]$  corresponding the clock period starting at  $t_i$ .

# Timing diagram of a Flip Flop

- The  $x$ -axis corresponds to time.
- A green interval means that the signal is stable during this interval.
- A red interval means that the signal may be instable.

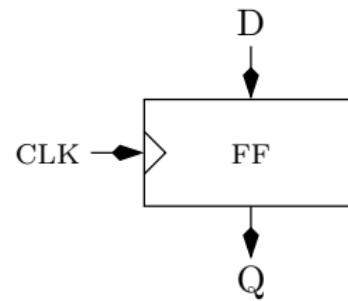


## Remarks about flip-flops

- 1 The assumption  $-t_{su} < t_{hold} < t_{cont} < t_{pd}$  implies that the critical segment  $C_i$  and the instability segment  $A_i$  are disjoint.
- 2 If  $D(t)$  is stable during the critical segment  $C_i$ , then the value of  $D(t)$  during the critical segment  $C_i$  is well defined and equals  $D(t_i)$ .
- 3 The flip-flop **samples** the input signal  $D(t)$  during the critical segment  $C_i$ . Sampling is successful only if  $D(t)$  is stable while it is sampled.
- 4 If the input  $D(t)$  is stable during the critical segments  $\{C_i\}_i$ , then the output  $Q(t)$  is stable in between the instability segments  $\{A_i\}_i$ .
- 5 The stability of the input  $D(t)$  during the critical segments depends on the clock period. We will later see that slowing down the clock (i.e., increasing the clock period) helps in achieving a stable signal  $D(t)$  during the critical segments.

# Flip-flop schematic

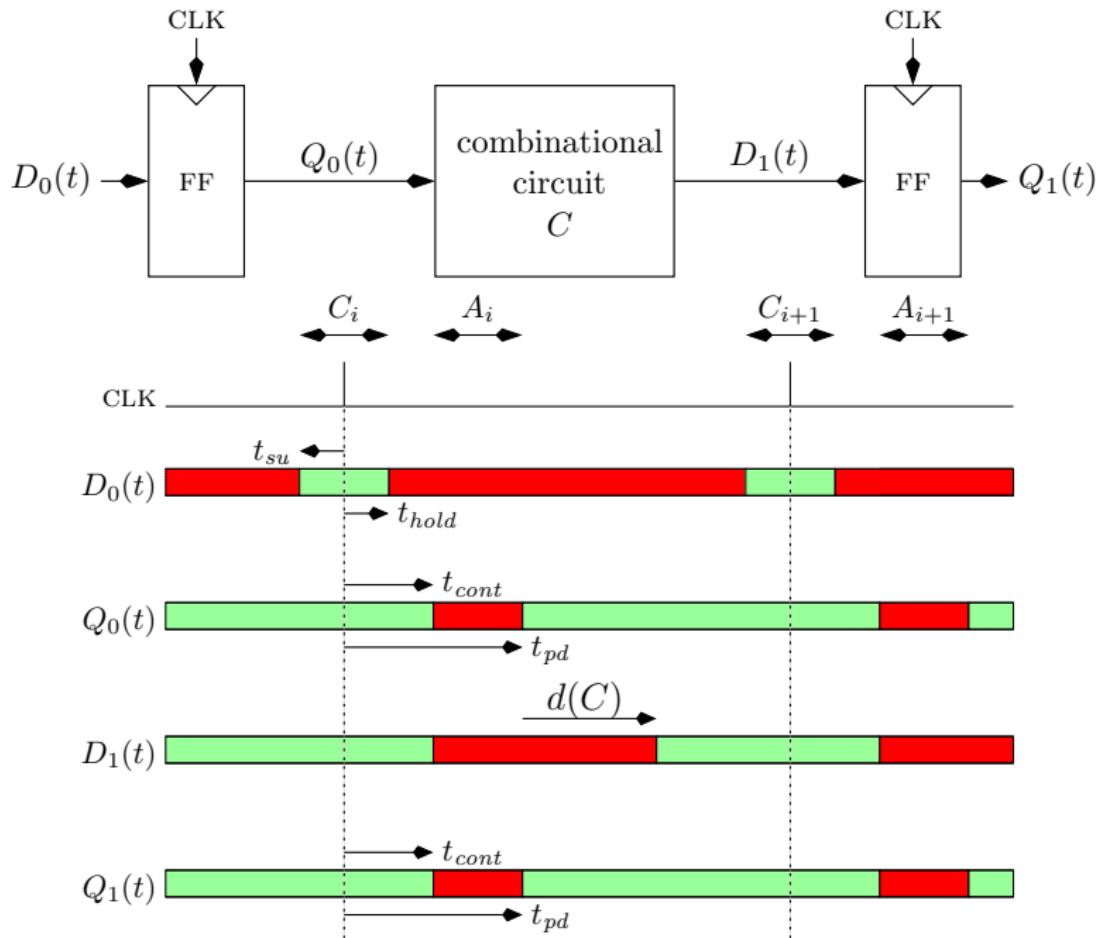
The special “arrow” that marks the clock-port.

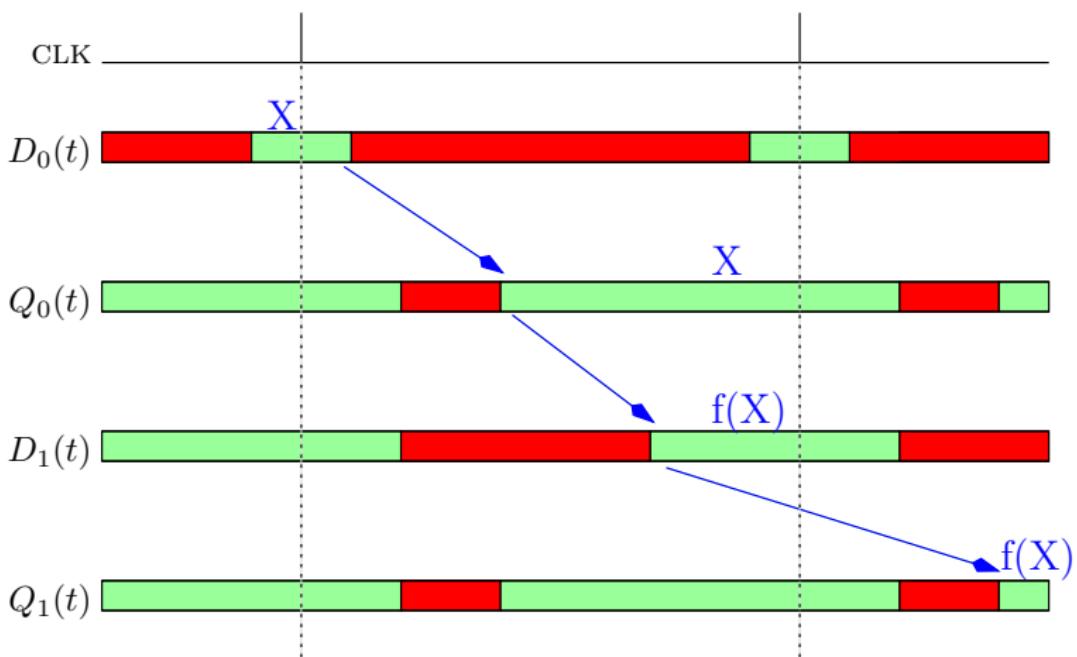
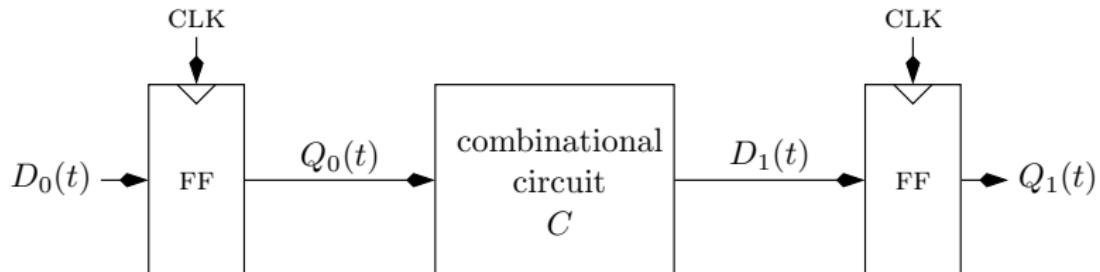


In the book you can find a proof of the following theorem:

### Theorem

*There does not exist an edge-triggered flip-flop without a critical segment.*





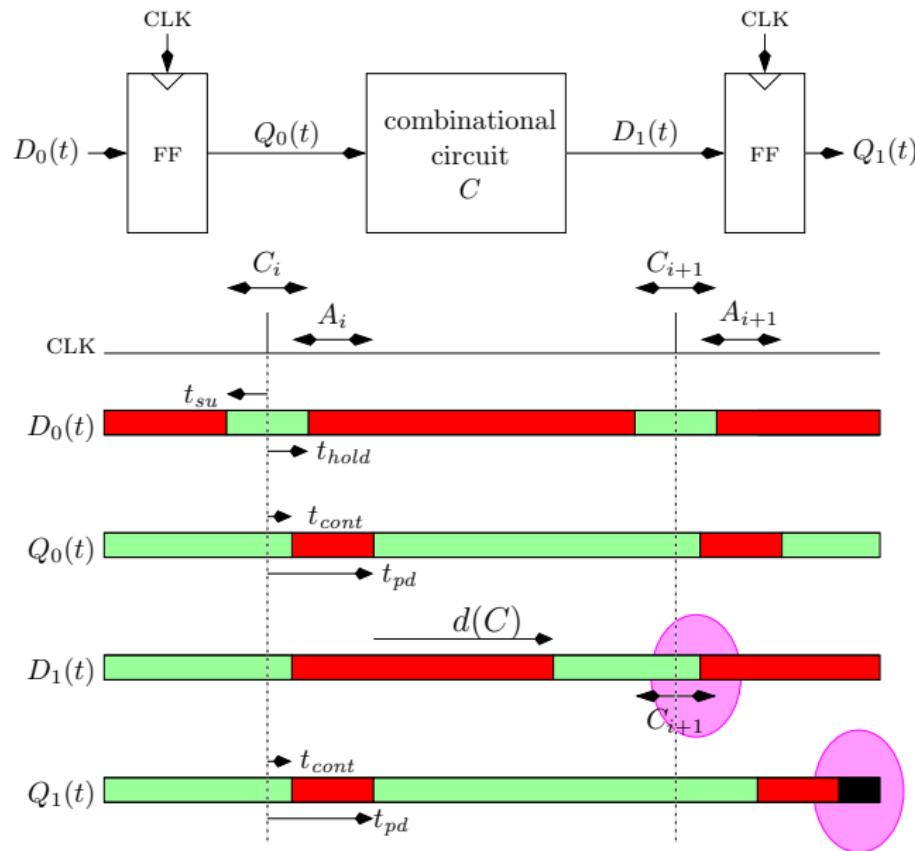
# Non-empty intersection of $C_i$ and $A_i$

The timing analysis fails if

$$C_i \cap A_i \neq \emptyset.$$

This could happen, if  $t_{hold} > t_{cont}$  (in contradiction to the definition of a flip-flop).

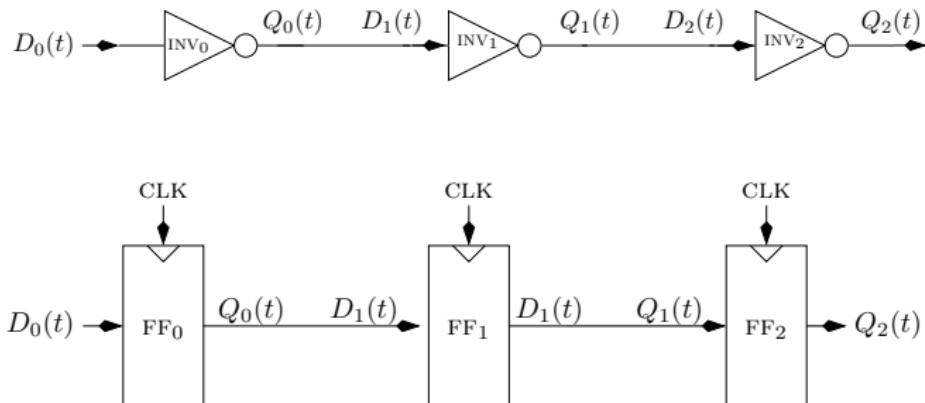
$$t_{hold} > t_{cont}$$



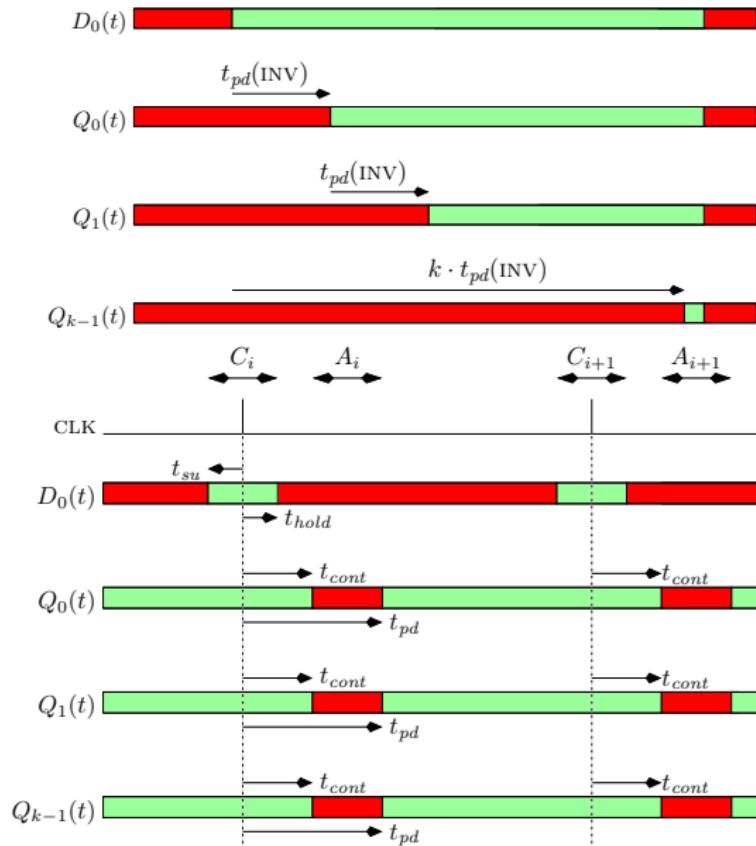
# Bounding Instability

Flip-flops play a crucial role in bounding the segments of time during which signals may be unstable. Informally, uncertainty increases as the segments of stability become shorter. Flip-flops help bounding instability.

# A chain of $k$ inverters and a chain of $k$ flip-flops



## timing: chain of inverters vs. chain of FFs



# Clock enabled flip-flops

## Definition

A clock enabled flip-flop is defined as follows.

Inputs: Digital signals  $D(t)$ ,  $CE(t)$  and a clock  $CLK$ .

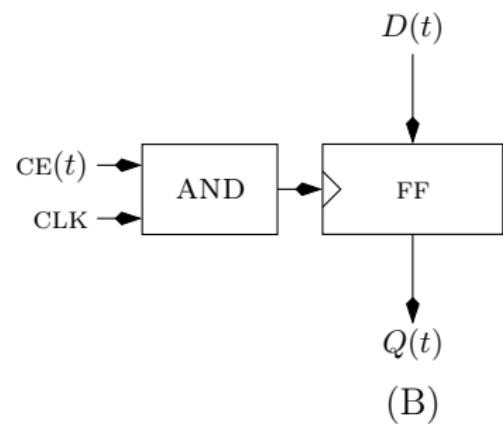
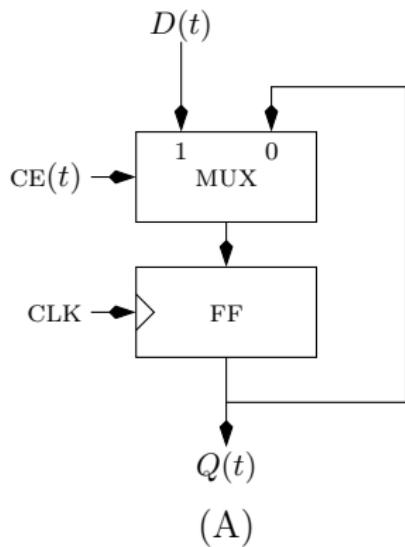
Output: A digital signal  $Q(t)$ .

Functionality: If  $D(t)$  and  $CE(t)$  are stable during the critical segment  $C_i$ , then for every  $t \in (t_i + t_{pd}, t_{i+1} + t_{cont})$

$$Q(t) = \begin{cases} D(t_i) & \text{if } CE(t_i) = 1 \\ Q(t_i) & \text{if } CE(t_i) = 0. \end{cases}$$

We refer to the input signal  $CE(t)$  as the clock-enable signal. Note that the input  $CE(t)$  indicates whether the flip-flop samples the input  $D(t)$  or maintains its previous value.

## Which design is a correct clock enabled FF?



# Summary

- Presented memory devices called **flip-flops** and the **clock signal**.
- The clock signal causes the flip-flop to sample the value of the input towards the end of a clock cycle and output the sampled value during the next clock cycle.
- Flip-flops play a crucial role in bounding the segments of time during which signals may be instable.
- Flip-flops and combinational circuits have opposite roles.
  - Combinational circuits compute interesting Boolean functions but increase uncertainty.
  - Flip-flops, on the other hand, output the same value that is fed as input but they limit uncertainty.
- We considered a task called **arbitration**. We proved that no circuit can implement an arbiter.
- We proved that a flip-flop with an empty critical segment can be used to build an arbiter. This proves that a flip-flop must have a non-empty critical segment.