Chapter 9: Flip-Flops

Computer Structure
&
Intro. to Digital Computers

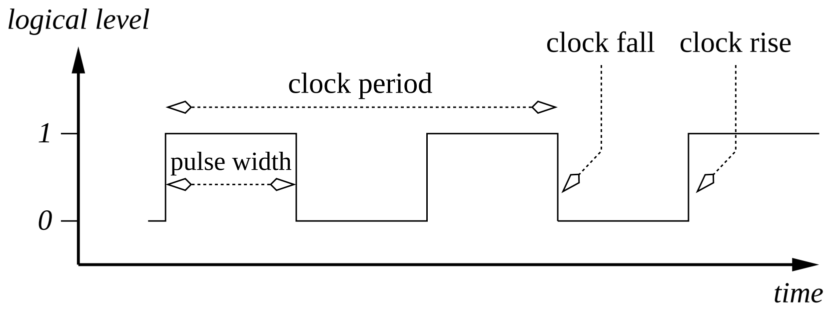
Dr. Guy Even

Tel-Aviv Univ.

Goals

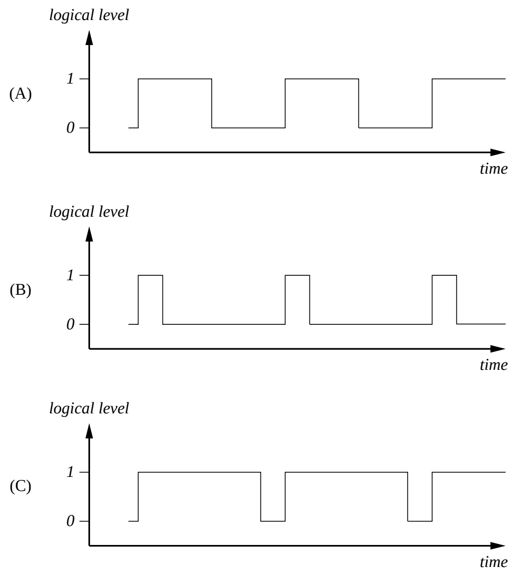
- introduce clock signal.
- define edge-triggered flip-flops.
- discuss parameters of flip-flops: setup time, hold time, contamination delay, propagation delay.
- explain importance of critical segment.
- understand timing of a flip-flop.
- other memory devices.

The Clock



- digital signal with periodic oscillations between 0 and 1.
- oscillations are instantaneous.
- each clock period starts with a $0 \to 1$ transition.
- $1 \to 0$ transition in the interior of the clock period.
- we denote the clock signal by $\textsc{clk}$.

Clock terminology



- clock period - denoted by $\varphi(\textsc{clk})$.
- clock pulse - interval during which $\textsc{clk}(t) = 1$.
- $\textsc{clk}_{pw}$ - duration of clock pulse.
- symmetric clock - if $\textsc{clk}_{pw} = \varphi(\textsc{clk})/2$.
- narrow pulses - if $\textsc{clk}_{pw} < \varphi(\textsc{clk})/2$.
- wide pulses - if $\textsc{clk}_{pw} > \varphi(\textsc{clk})/2$.

Clock cycles

A clock partitions time into discrete intervals as follows:

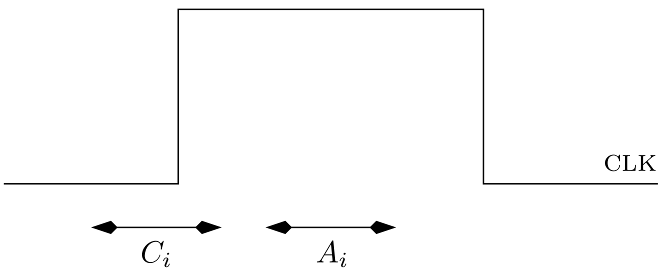
- Let t_i denote the starting time of the ith clock period.
- We refer to the half-closed interval $[t_i, t_{i+1})$ as *clock cycle* i.

Parameters of an Edge-triggered Flip-Flop

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

These parameters satisfy $-t_{su} < t_{hold} < t_{cont} < t_{pd}$.
Notation:

- critical segment: $C_i = [t_i - t_{su}, t_i + t_{hold}]$.
- instability segment: $A_i = [t_i + t_{cont}, t_i + t_{pd}]$.

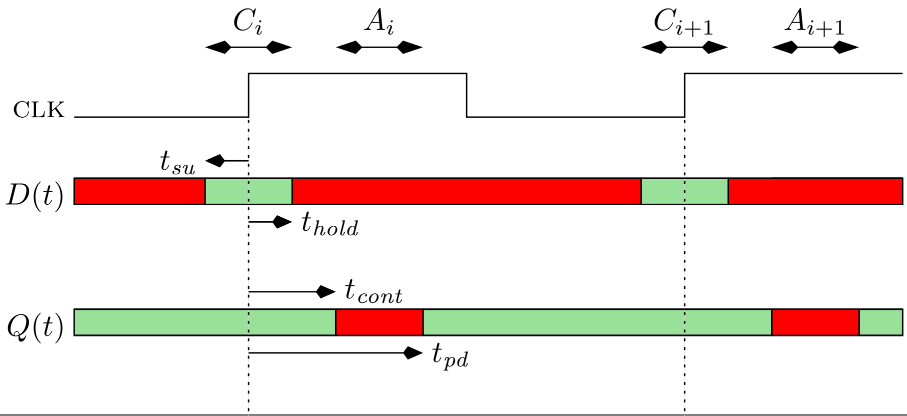


Definition: Edge-triggered Flip-Flop

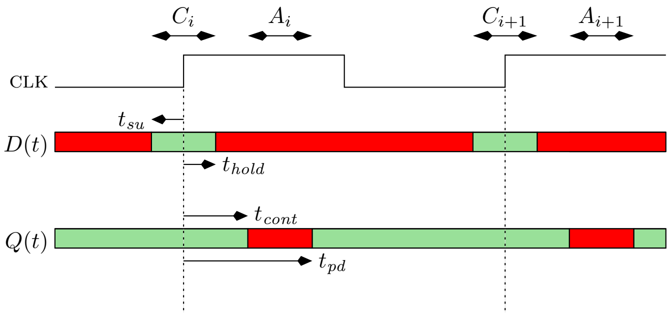
Inputs: A digital signal $D(t)$ and a clock CLK.

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

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

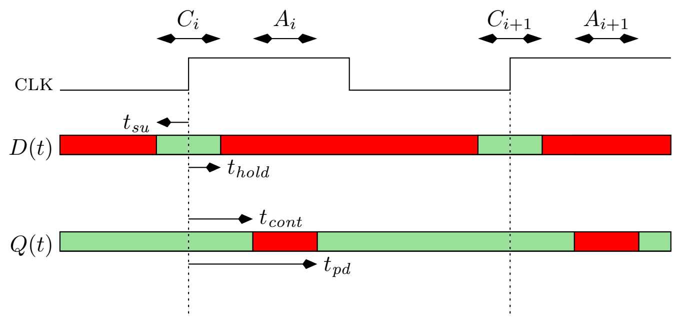


Remarks on definition of flip-flop



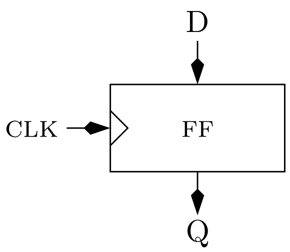
- $-t_{su} < t_{hold} < t_{cont} < t_{pd} \implies C_i \cap A_i = \emptyset$.
- Stability of $D(t)$ during $C_i \Rightarrow$ digital value of $D(t)$ during the critical segment C_i is logical and equals $D(t_i)$.
- Flip-flop samples $D(t)$ during C_i. The sampled value $D(t_i)$ is output during the interval $[t_i + t_{pd}, t_{i+1} + t_{cont}]$.
- Sampling is successful only if $D(t)$ is stable while it is sampled. This is why we refer to C as a critical segment.

Remarks on definition of flip-flop - cont.



- 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 $\{A_i\}_i$.
- 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 $D(t)$ during the critical segments.

schematic of an edge triggered flip-flop



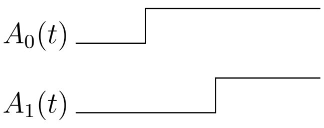
- clock port is marked by an “arrow”.
- we abbreviate and refer to an edge-triggered flip-flop simply as a flip-flop.

Question: Prove that an edge-triggered flip-flop is not a combinational circuit.

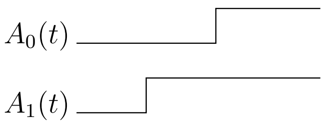
Arbitration

Arbitration is the problem of deciding which event occurs first.

Focus on the task of determining which of two signals reaches 1 first.



$A_0(t)$ reaches 1 first



$A_1(t)$ reaches 1 first

Definition: arbiter

Inputs: Non-decreasing analog signals $A_0(t), A_1(t)$ defined for every $t \geq 0$.

Output: An analog signal $Z(t)$.

Functionality: Assume that $A_0(0) = A_1(0) = 0$. Define T_i, for $i = 0, 1$, as follows:

$$T_i \triangleq \inf\{t \mid \textit{dig}(A_i(t)) = 1\}.$$

Let $t' \triangleq 10 + \max\{T_0, T_1\}$. The output $Z(t)$ must satisfy, for every $t \geq t'$,

$$\textit{dig}(Z(t)) = \begin{cases} 0 & \text{if } T_0 < T_1 - 1 \\ 1 & \text{if } T_1 < T_0 - 1 \\ 0 \text{ or } 1 & \text{otherwise.} \end{cases}$$

Arbiter - remarks

- $$T_i \triangleq \inf\{t \mid \textit{dig}(A_i(t)) = 1\}.$$

  If T_0 or T_1 equals infinity, then $t' = \infty$, and there is no requirement on the output $Z(t)$.
- Arbiter circuit is given 10 time units starting from $\max\{T_0, T_1\}$ to determine if $T_0 < T_1$ or $T_1 < T_0$.
- tie: the case that $|T_0 - T_1| \leq 1$.
- In the case of a tie, the arbiter is free to decide, but must decide. $Z(t)$ is stable in the interval $[t, \infty)$.

Arbiters - an impossibility result

Claim: There does not exist a circuit C that implements an arbiter.

- Inherent limitation - not just a weakness of the digital abstraction.
- Use the claim to show that flip-flops must have critical segments.

Proof: every circuit C is not an arbiter

- Define $A_0(t)$ so that $T_0 = 100$ as follows:

$$A_0(t) \triangleq \begin{cases} \frac{t}{100} \cdot V_{\textit{high,in}} & \text{if } t \in [0,100] \\ V_{\textit{high,in}} & \text{if } t > 100. \end{cases}$$

- Fix a parameter $x \in [-2, 2]$ and define $A_1(t)$ so that $T_1 = 100 + x$ as follows:

$$A_1(t) \triangleq \begin{cases} \frac{t}{100+x} \cdot V_{\textit{high,in}} & \text{if } t \in [0,100+x] \\ V_{\textit{high,in}} & \text{if } t > 100+x. \end{cases}$$

- Define the function $f(x)$ by $f(x) \triangleq Z(200)$.
- We study the function $f(x)$ in the interval $x \in [-2, 2]$.

Proof: every circuit C is not an arbiter - cont.

- $x = -2 \Rightarrow T_1 = 100 + x = 98$. It follows that $A_1(t)$ “wins”, and $Z(200) = 1$. Hence $f(-2) = 1$.
- $x = 2 \Rightarrow T_1 = 100 + x = 102$. It follows that $A_0(t)$ “wins”, and $Z(200) = 0$. Hence $f(2) = 0$.
- claim: $f(x)$ is continuous (will prove this later).
- Mean Value theorem $\Rightarrow$

$$\forall y \in [V_{\textit{low,out}}, V_{\textit{high,out}}] \ \exists x \in [-2, 2] \ : \ f(x) = y.$$

- Pick y such that *dig*$(y) =$ non-logical.
- $\Rightarrow$ There exist valid inputs $A_0(t), A_1(t)$ with $t' \leq 112$, such that *dig*$(Z(200)) =$non-logical.
- $\Rightarrow$ C is not an arbiter. QED.

Proof: $f(x)$ is continuous

Rely on the assumption that an infinitesimal change in the energy of input signals causes an infinitesimal change in the energy of the output. Otherwise, noise would cause uncontrollable changes in $Z(t)$ and the circuit C would not be useful anyhow.

The output $Z(200)$ depends on the following:

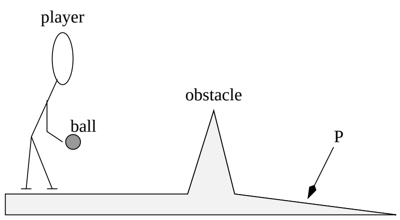
1. The initial state of the device C at time $t = 0$. We assume that the device C is in a stable state and that the charge is known everywhere.
2. The signal $A_i(t)$ in the interval $[0, 200]$, for $i = 0, 1$.

Proof: $f(x)$ is continuous - cont.

- Consider an infinitesimal change in x. This change affects $A_1(t)$ but does not affect $A_0(t)$ and the initial state.
- infinitesimal change of $x \Rightarrow$ infinitesimal difference in energy of $A_1(t)$.
- infinitesimal difference in energy of $A_1(t) \Rightarrow$ infinitesimal difference in $Z(200)$.
- $\Rightarrow f(x)$ is continuous.

Discussion: Arbiters - an impossibility result

- Claim is counter-intuitive.
- For every judge in a 100-meter dash, there exist two runners whose running times are such that the judge still hangs after an hour.
- Implies that there does not exist a perfect judge who can determine the winner in a 100-meters dash even if:
  1. high speed cameras located at the finish line and runners run very slowly.
  2. we allow the judge several hours to decide.
  3. we allow the judge to decide arbitrarily if the running times of the winner and runner-up are within a second.

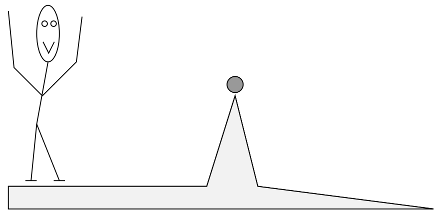


- Player - rolls a ball. Judge - announces decision if ball passes point P one day after.
- If speed of ball is above v', then ball passes the obstacle and then rolls past point P.
- If speed of ball is below v', then ball does not pass the obstacle.

Judge is in trouble:

- If speed$= v'$, then the ball reaches the tip of the obstacle and may remain there indefinitely long!
- If the ball remains on the obstacle's tip 24 hours past the throw, then the judge cannot announce her decision.

Meta-stability



- Meta-stability - a state of equilibrium (i.e. zero force) which is not a local minimum of energy (i.e. a slight force causes a movement away from the state).
- Inclined to say that the "probability of meta-stability occurring is very small". This requires a probability distribution over the rolling speed v where

$$\lim_{\varepsilon \to 0} Pr(|v - v'| < \varepsilon) = 0.$$

Lessons learned

- Certain tasks are not achievable with probability 1.
  - coin toss might end up with the coin standing on its perimeter.
  - noise could be big enough to cause the digital value of a signal to flip from zero to one. (increase noise margin to reduce the probability of such an event.)

Reducing the probability of meta-stability

- Increase length of segment of instability.
  Increasing the delay of the arbiter (significantly) decreases the chances of meta-stability. E.g., ball resting on the tip of the obstacle is likely to fall to one of the sides.
- Increase the slope of the transfer function in the range of non-logical values. Similar to sharpening the tip of the obstacle.
- However, increasing the clock rate means that "decisions" must be made faster (i.e. within a clock period) and the chance of meta-stability increases.

Question

Does the proof of the Claim hold only if the signals $A_i(t)$ rise gradually?

Prove the claim with respect to "fast" non-decreasing signals $A_i(t)$. Namely, the length of the interval during which *dig*$(A_i(t))$ is non-logical equals ε.

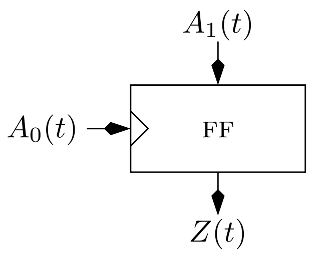
Flip-flops: necessity of critical segments

DEF: A flip-flop without a critical segment is a flip-flop in which the setup-time and hold-time satisfy $t_{su} = t_{hold} = 0$. The functionality is defined as follows:

- For every i, $Q(t)$ is logical (either zero or one) during the interval $t \in (t_i + t_{pd}, t_{i-1} + t_{cont})$ regardless of whether $D(t_i)$ is logical.
- If $D(t_i)$ is logical, then $Q(t) = D(t_i)$ during the interval $t \in (t_i + t_{pd}, t_{i-1} + t_{cont})$.

Just as the arbiter's decision is free if a tie occurs, the flip-flop is allowed to output zero or one if $D(t_i)$ is not logical. However, the output of the flip-flip must be logical once the instability segment ends.

An arbiter based on a flip-flop without a critical segment



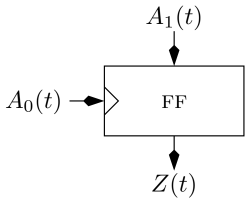
Assumptions:

- flip-flop is without a critical segment.
- $t_{cont}, t_{pd} \approx 10^{-9}$ time unit.
- intervals during which the inputs $A_0(t)$ and $A_1(t)$ are non-logical are also very short (e.g. 10^{-9} time unit).

Claim: The circuit above is an arbiter.

CORO: There does not exist a flip-flop without a critical section.

Remarks



- the signal $A_0(t)$ is input as a clock to the flip-flop, but $A_0(t)$ is not a clock.
- requirements from $A_0(t)$ are weaker than the requirements from a clock. Instead of periodic instantaneous transitions from zero to one and back, $A_0(t)$ is non-decreasing.
- the claim assumes only one "tick of the clock", so we may regard $A_0(t)$ as a clock with a very long period.
- proof of claim does not rely on $A_0(t)$ rising slowly; the claim holds regardless of the rate of change of $A_0(t)$.

Proof that circuit is an arbiter

We consider three cases:

- $|T_1 - T_0| \leq 1$: flip-flop's output $Z(t)$ is always logical at time $T_0 + t_{pd}$, so circuit functions properly.
- $T_1 < T_0 - 1$: if $T_1 < T_0 - 1$, then *dig*$(A_1(T_0)) = 1$. Hence sampled value equals 1, and hence, *dig*$(Z(t)) = 1$, for every $t \geq T_0 + t_{pd}$.
- $T_0 < T_1 - 1$: we claim that *dig*$(A_1(T_0)) = 0$, and hence, *dig*$(Z(t)) = 0$, for every $t \geq T_0 + t_{pd}$.

Proof that circuit is an arbiter - cont.

We need to show that $T_0 < T_1 - 1 \Rightarrow$ *dig*$(A_1(T_0)) = 0$.

- $T_0 < T_1 \Rightarrow$ *dig*$(A_1(T_0)) \in \{0,$ non-logical$\}$.
- assumption on the fast transition of *dig*$(A_1(t))$ implies:

dig$(A_1(T_0)) =$ non-logical $\Rightarrow$ *dig*$(A_1(T_0 + 10^{-9})) = 1.$

Hence, $T_1 \leq T_0 + 10^{-9}$ contradicting $T_1 > T_0 + 1$.

It follows that if $T_0 < T_1 - 1$, then *dig*$(A_1(T_0)) = 0$. QED

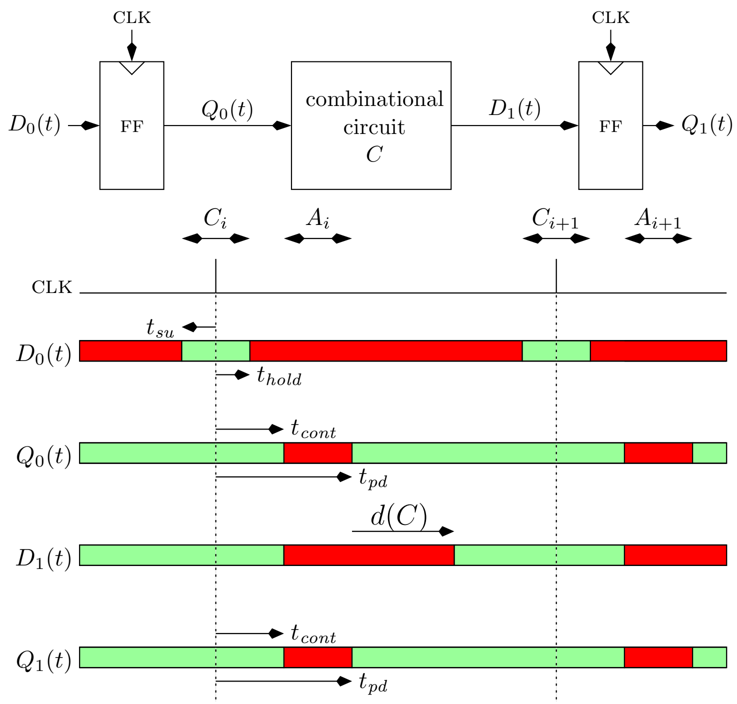
Corollary: conclusion

Critical segment is required to avoid meta-stability of the flip-flop.

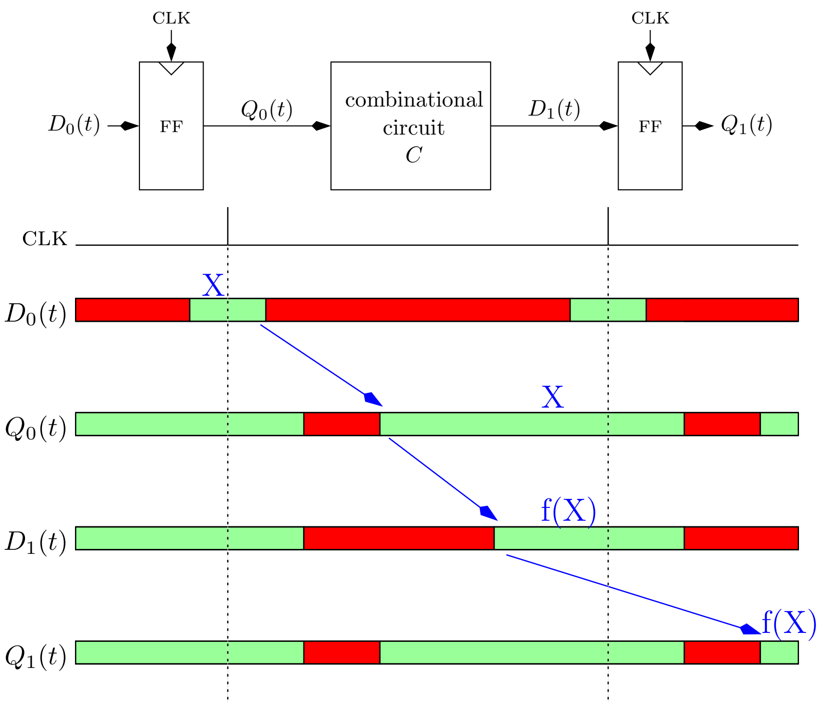
Without critical segment, flip-flop's output can be non-logical

even after $t_i + t_{pd}$.

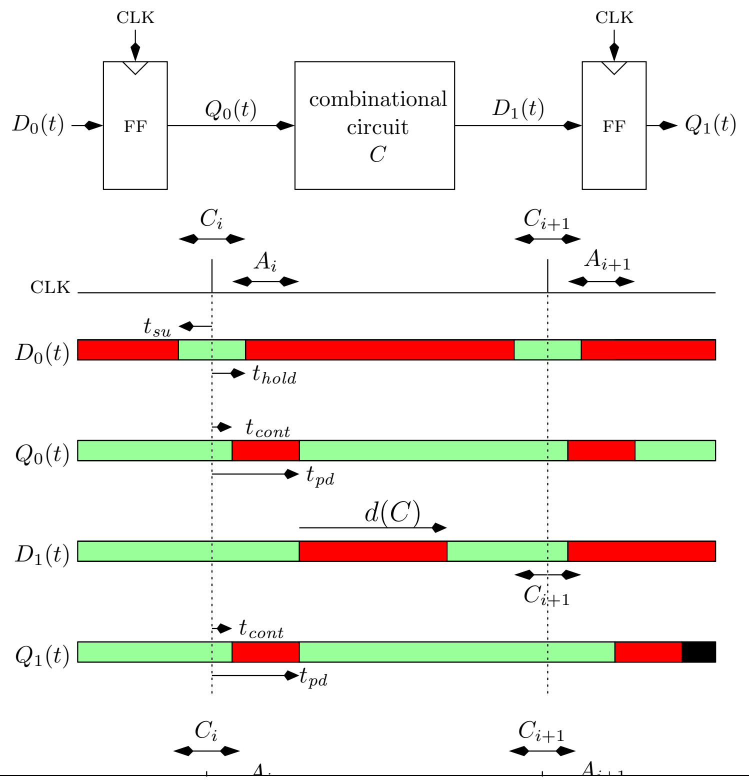
An example: timing



An example: functionality



Non-disjoint segments: $A_i \cap C_i \neq \emptyset$



What if $A_i \cap C_i \neq \emptyset$?

Stability interval of $D_1(t)$ is:

$$[t_i + t_{pd} + d(C), t_{i+1} + t_{cont}].$$

If $t_{cont} < t_{hold}$, then $D_1(t)$ is not stable during

$$C_{i+1} = [t_{i+1} - t_{su}, t_{i+1} + t_{hold}].$$

In this case, we need to rely on the contamination delay $\mathit{cont}(C)$ of the combinational circuit C.

Now $D_1(t)$ is stable during the interval

$$[t_i + t_{pd} + d(C), t_{i+1} + t_{cont} + \mathit{cont}(C)].$$

If $t_{cont} + \mathit{cont}(C) > t_{hold}$, then the signal $D_1(t)$ is stable during the critical segment C_{i+1}, and correct functionality is obtained.

Contamination delay of combinational circuits

- Can help in obtaining stability during the critical segment.
- Many combinational gates have a positive contamination delay. But some don't.
- Relying on the contamination delay of combinational circuits complicates timing analysis.
- We use a strict assumption that $\textit{cont}(C) = 0$, for every combinational circuit C. This does not cause incorrect circuits even if $\textit{cont}(C) > 0$.

Fixing $A_i \cap C_i \neq \emptyset$

Question: Assume that we have an edge-triggered flip-flop FF in which $t_{hold} > t_{cont}$. Suppose that we have an inverter with a contamination delay $\textit{cont}(\text{INV}) > 0$.

- Suggest how to design an edge-triggered flip-flop FF$'$ that satisfies $t_{hold}(\text{FF}') < t_{cont}(\text{FF}')$.
- What are the parameters of FF$'$?

D-Latch: parameters

- characterized by two parameters t_{su}, t_{hold}
- the critical segment is defined with respect to the falling edge of the clock.
- t'_i - time of the falling edge of the clock during the ith clock cycle.
- critical segment of a D-latch is

$$[t'_i - t_{su}, t'_i + t_{hold}].$$

- d - combinational delay of the D-latch.

D-Latch: definition

- During the interval $[t_i + d, t'_i)$, the output $Q(t)$ satisfies: $Q(t) = D(t)$, provided that $D(t)$ is stable during the interval $[t - d, t]$. We say that the D-latch is transparent during the interval $[t_i + d, t'_i)$.
- During the interval $(t'_i + t_{hold}, t_{i+1})$, if $D(t)$ is stable during the critical segment $[t'_i - t_{su}, t'_i + t_{hold}]$, then $Q(t) = D(t'_i)$. We say that the D-latch is opaque during the interval $(t'_i + t_{hold}, t_{i+1})$.

D-Latch : story

- D-latches are very important devices.
- D-latches are cheaper than flip-flops, and in fact, D-latches are the building blocks of flip-flops (e.g. master/slave designs).
- using D-latches wisely leads to faster designs.
- designs based on D-latches require multiple clock phases (or at least a clock CLK and its negation $\overline{\text{CLK}}$).
- Although timing with multiple clock phases is an important and interesting topic, we do not deal with it in this course.

Definition : clock enabled flip-flips

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

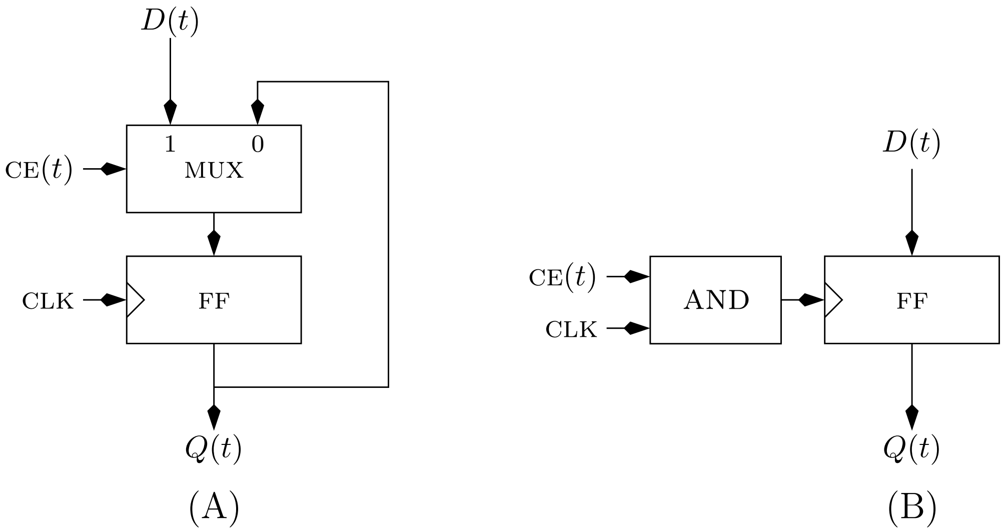
Output: A digital signal $Q(t)$.

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

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

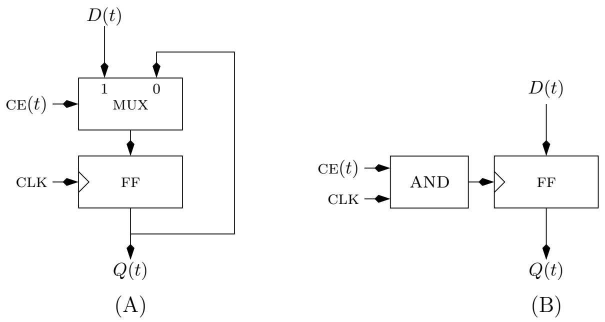
- CE(t) - clock-enable signal.
- CE(t) indicates whether the flip-flop samples the input $D(t)$ or maintains its previous value.

Clock enabled flip-flips : implementation



Question: Which design is correct?

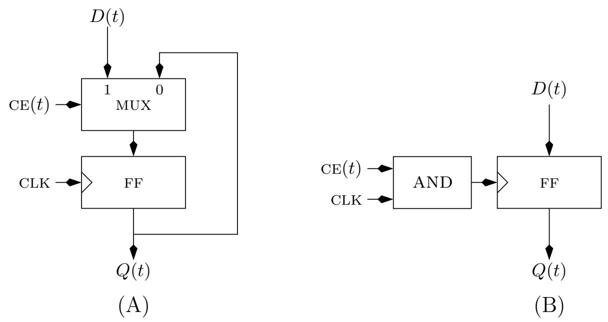
Clock enabled flip-flips : implementation - cont



Design (B) is wrong because:

- output of the AND-gate is not a clock signal (glitches).
- slow transitions of the output of the AND-gate (increase hold time)
- in some technologies, the flip-flop does not retain the stored bit forever. $\Rightarrow$ if $\text{CE}(t) = 0$ for a long period, then the flip-flop's output may become non-logical.

Clock enabled flip-flips : implementation - cont



Question: Compute the parameters of the clock-enabled flip-flop depicted in part (A) in terms of the parameters of the edge-triggered flip-flop and the MUX.

Summary

- clock signal - definition, terminology
- define edge-triggered flip-flops
- prove that critical segments are crucial:
  - arbitration - the problem of deciding “whose first”
  - prove that arbiters do not exist
  - use this proof to show that critical segments are crucial
- a timing example
- other memory devices: D-latch & clock-enabled flip-flop