
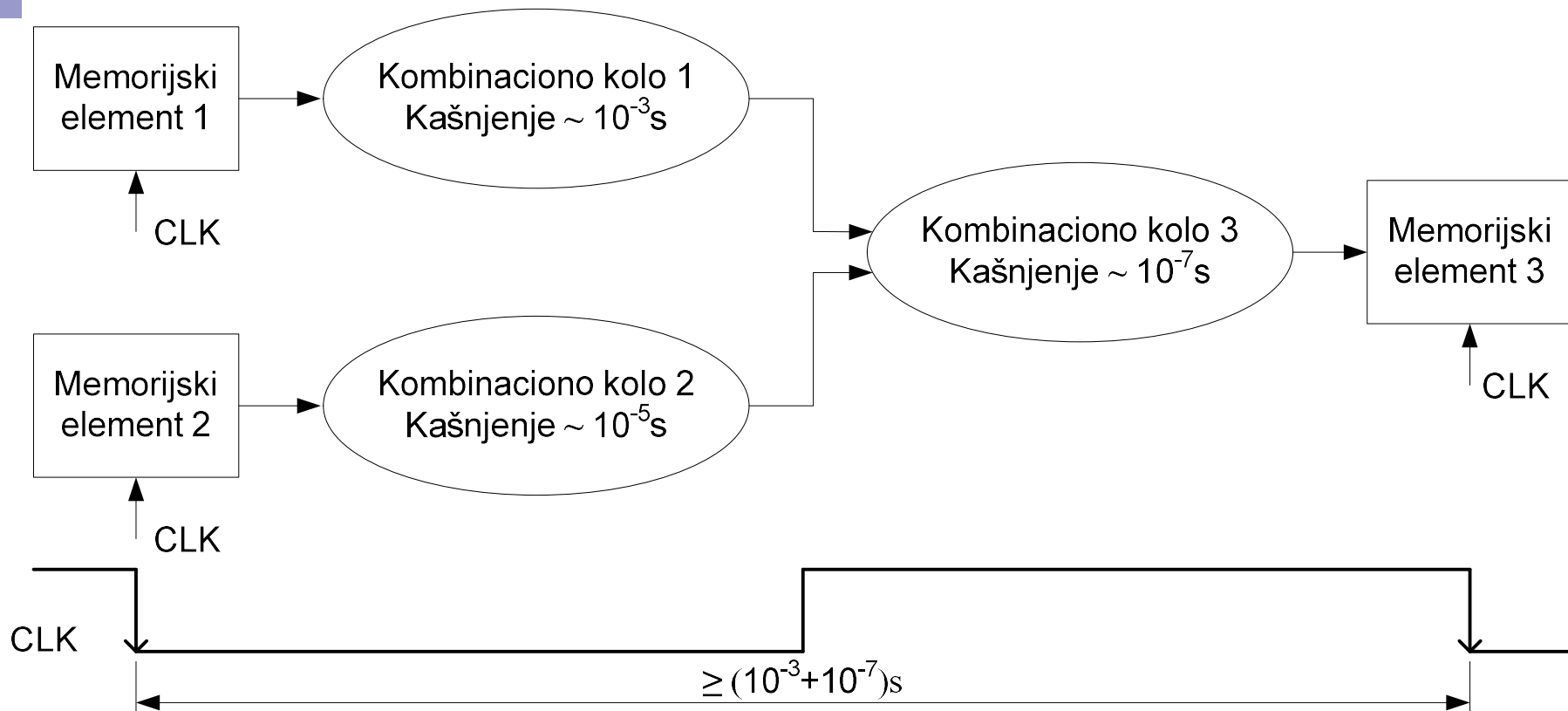




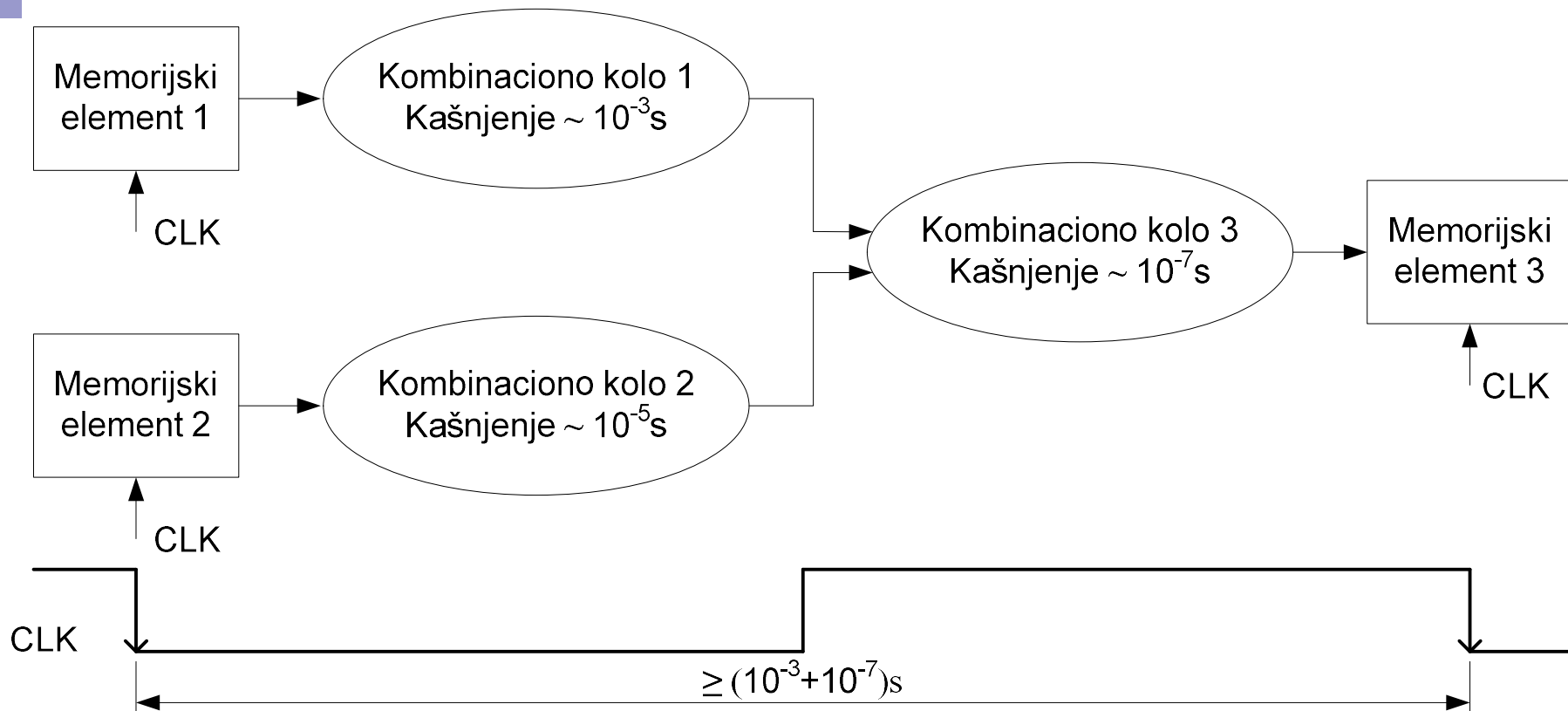
# Osnovi računarstva I

Osnovni digitalni sistemi  
(leč i flip-flop)

- 
- Do sada smo posmatrali tzv. *idealni slučaj*: kolo reaguje u istom trenutku kada se promijeni signal na ulazu
  - Prolaskom ulaznog signala kroz jedno ili veći broj logičkih kola unosi se određeno vremensko **kašnjenje**
  - Kašnjenja po jednom elektronskom elementu su obično veoma mala, reda veličine nekoliko nanosekundi ( $10^{-9}$  s)
  - Prolaskom kroz veliki broj elektronskih elemenata ukupna kašnjenja mogu postati značajna
  - Da bi sistem ispravno funkcionisao nekada je potrebno uvesti *kontrolni* signal
  - Često se u mrežama koriste memorijski elementi – kod njih je naročito važna kontrola upisa
  - Osnovni memorijski elementi su *leč (latch)* i *flip-flop*
  - Pomoću osnovnih memorijskih elemenata grade se složeniji memorijski elementi, npr. *registri*



- Pomoću kontrolnih signala vrši se **sinhronizacija** između mreža različitog stepena složenosti, odnosno mreža koje unose različita kašnjenja
- Sindhronizacija se najčešće vrši pomoću kontrolnog signala u obliku periodične povorke pravougaonih impulsa, koju generiše vremenski nezavisni uređaj, a koja se naziva **takt** (eng. clock)
- **Uzlazna** i **silazna** ivica taktnog impulsa

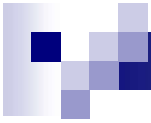


- Takt se kreira tako da omogući izvršavanje najduže akcije u sistemu (akcija koja unosi najveće kašnjenje) za vrijeme trajanja jednog taktnog impulsa
- Zbirno kašnjenje kombinacionih kola 1 i 3 (kao veće od zbirnog kašnjenja kombinacionih kola 2 i 3) određuje vremensko ponašanje sistema, odnosno trajanje takta

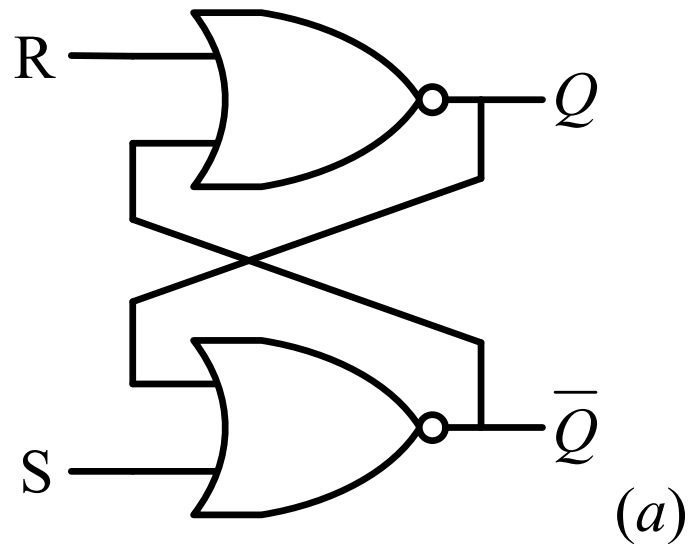


# OSNOVNI MEMORIJSKI ELEMENTI – LEČ I FLIP-FLOP –

- Osnovni memorijski element predstavlja jedinicu memorije koja može čuvati jedan bit informacije
- Memorijski element kod kojeg se sadržaj mijenja sa promjenom nivoa signala na njegovim ulazima, bez uticaja bilo kakvog kontrolnog signala, u literaturi se obično naziva *leč* (eng. *latch*)
- Ako postoji kontrolni signal koji će upravljati procesom upisivanja u leč, onda se takav leč naziva *upravljivi leč*.
- Obično se u leč može upisivati kada je kontrolni signal na nivou logičke jedinice
- U praksi je često potrebno da se upis u memorijski element izvršava u tačno određenim, diskretnim vremenskim trenucima koji se određeni promjenom stanja kontrolnog signala (uzlazna ili silazna ivica takta)
- Memorijski elementi koji reaguju na ivicu kontrolnog signala nazivaju se *edge-triggerred flip-flopovi*



## RS LEČ

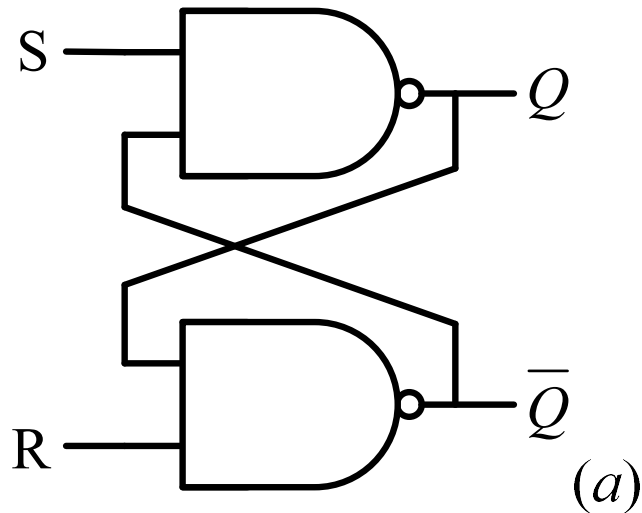


R	S	$Q_{t+1}$	$\bar{Q}_{t+1}$	
0	0	$Q_t$	$\bar{Q}_t$	← Zatečeno stanje
0	1	1	0	← Set
1	0	0	1	← Reset
1	1	0	0	← Nedef. stanje

(b)

- Varijanta 1: upotrebom **NILI** kola

## RS LEČ

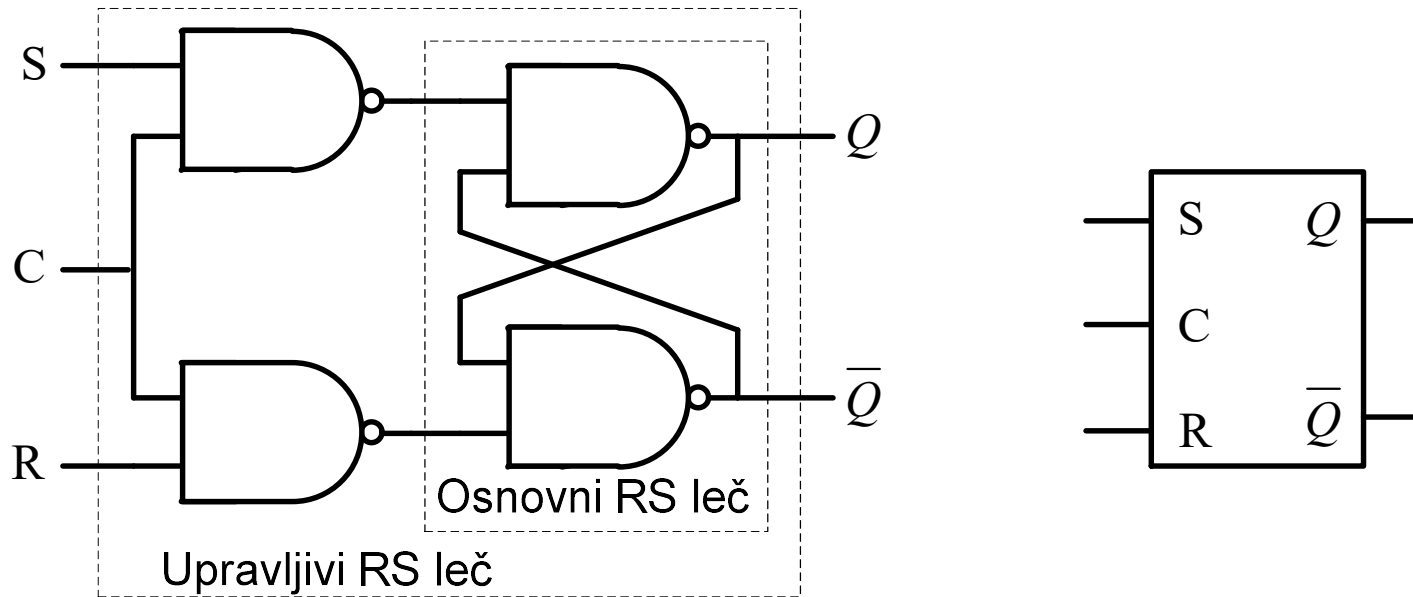


R	S	$Q_{t+1}$	$\bar{Q}_{t+1}$	
0	0	1	1	← Nedef.stanje
0	1	0	1	← Reset
1	0	1	0	← Set
1	1	$Q_t$	$\bar{Q}_t$	← Zatečeno stanje

(b)

- Varijanta 2: upotrebom **NI** kola
- Inverzna (negativna) logika na ulazima: signali R i S su aktivni kada su na nivou logičke nule
- Zbog toga se RS leč sa međusobno povezanim NI kolima u literaturi često naziva  $\text{inv}(R)\text{inv}(S)$  leč

# UPRAVLJIVI RS LEČ



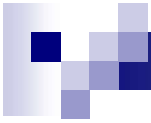
Osnovni:

R	S	$Q_{t+1}$	$\bar{Q}_{t+1}$	
0	0	1	1	← Nedef. stanje
0	1	0	1	← Reset
1	0	1	0	← Set
1	1	$Q_t$	$\bar{Q}_t$	← Zatečeno stanje

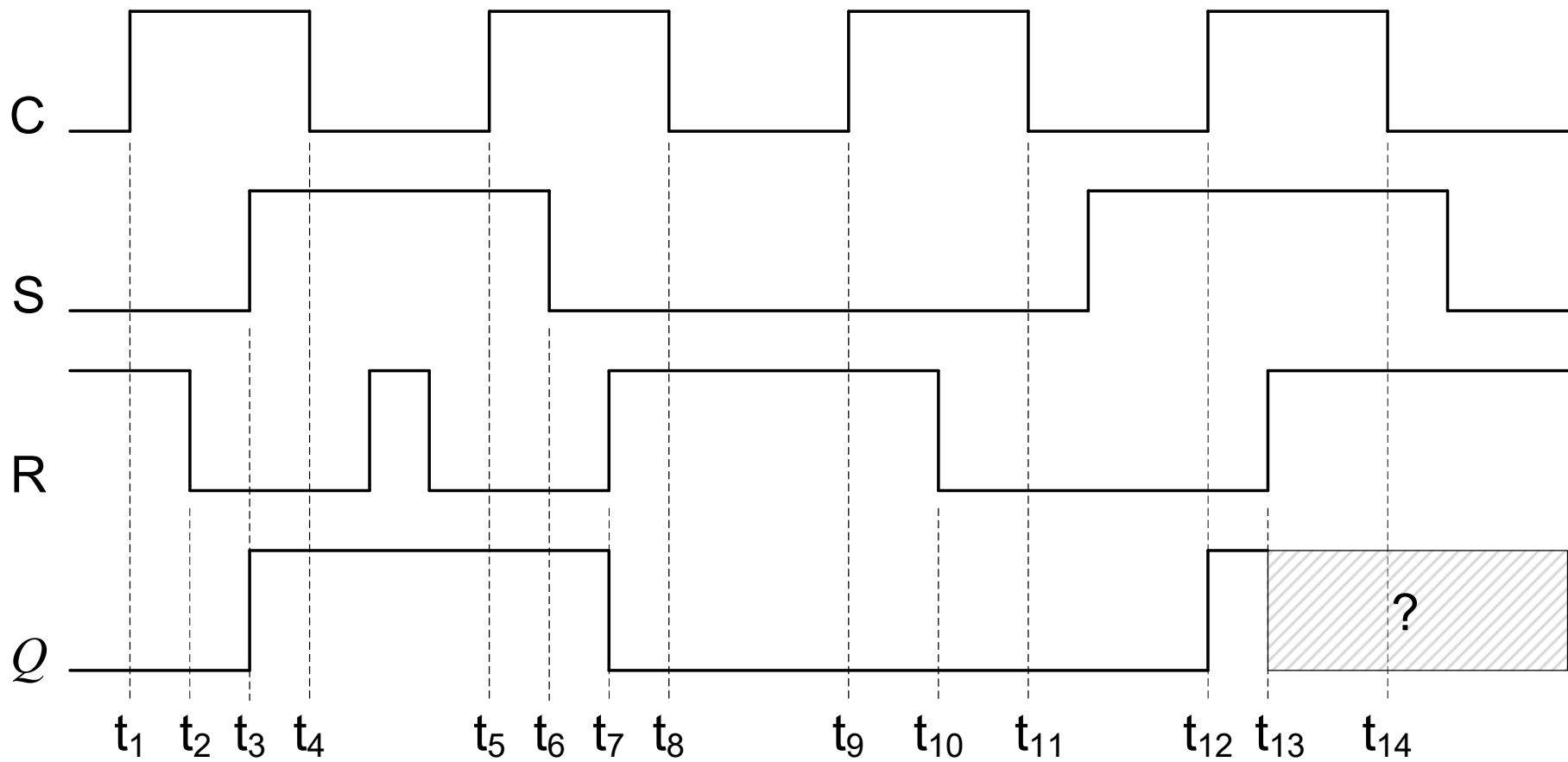
Upravljivi:

C	R	S	$Q_{t+1}$	
0	X	X	$Q_t$	← Zatečeno stanje
1	0	0	$Q_t$	← Zatečeno stanje
1	0	1	1	← Set
1	1	0	0	← Reset
1	1	1	Nedef.	← Nedef. stanje

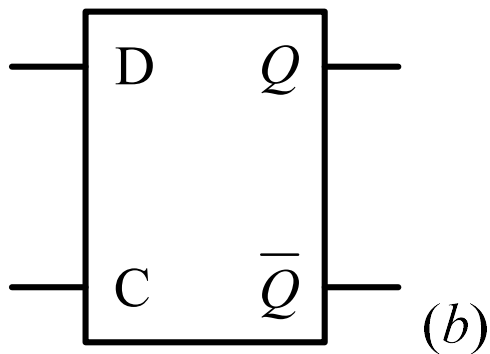
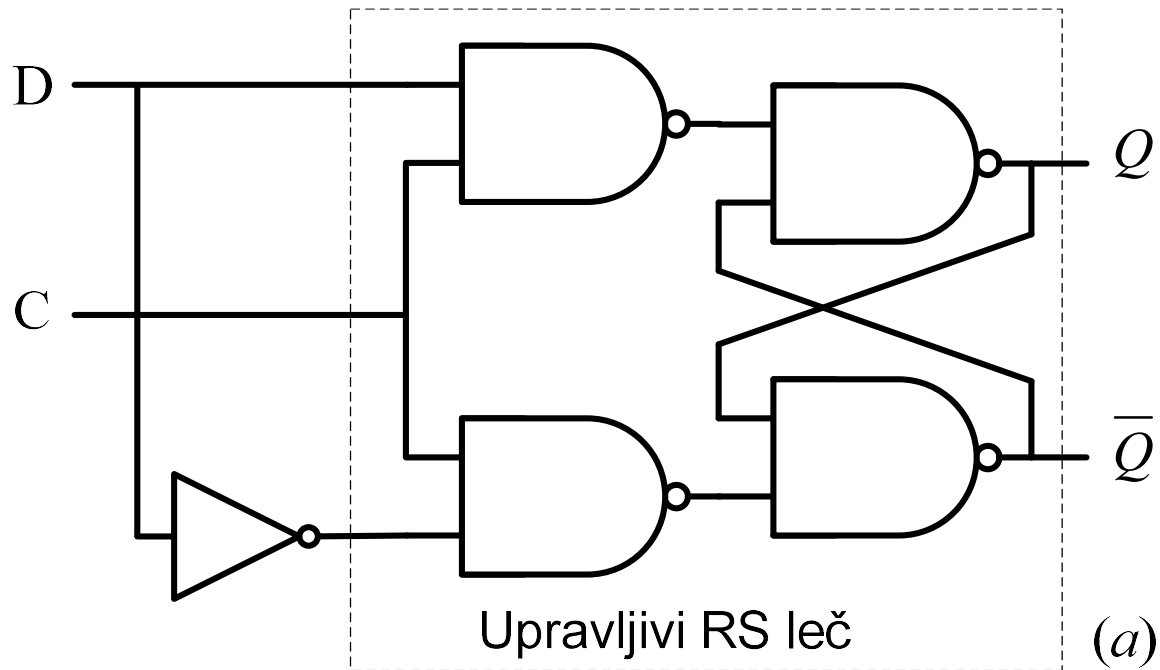




■ Primjer:

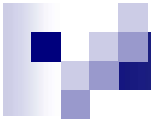


# UPRAVLJIVI D LEČ

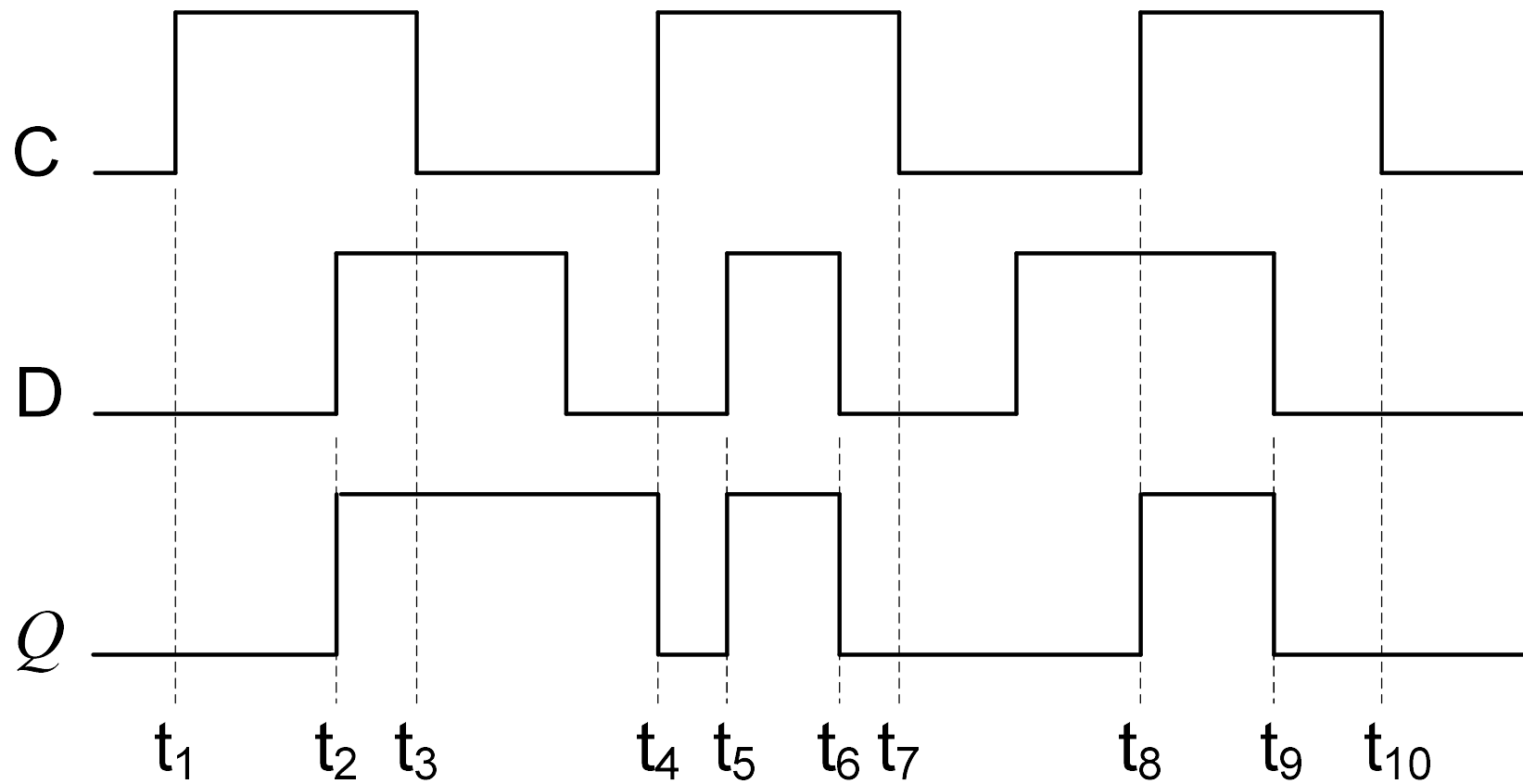


C	D	$Q_{t+1}$	
0	X	$Q_t$	← zatečeno stanje
1	0	0	← Reset
1	1	1	← Set

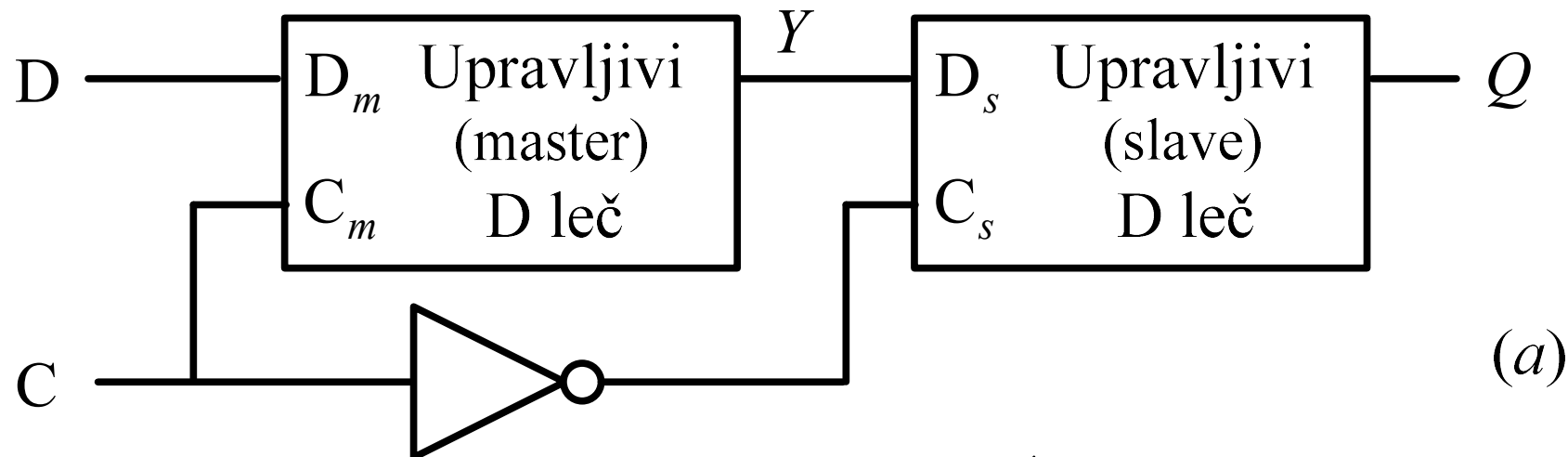
(c)



■ Primjer:



# D FLIP-FLOP



C	D	$Q_{t+1}$	
↓	0	0	← Reset
↓	1	1	← Set

(b)

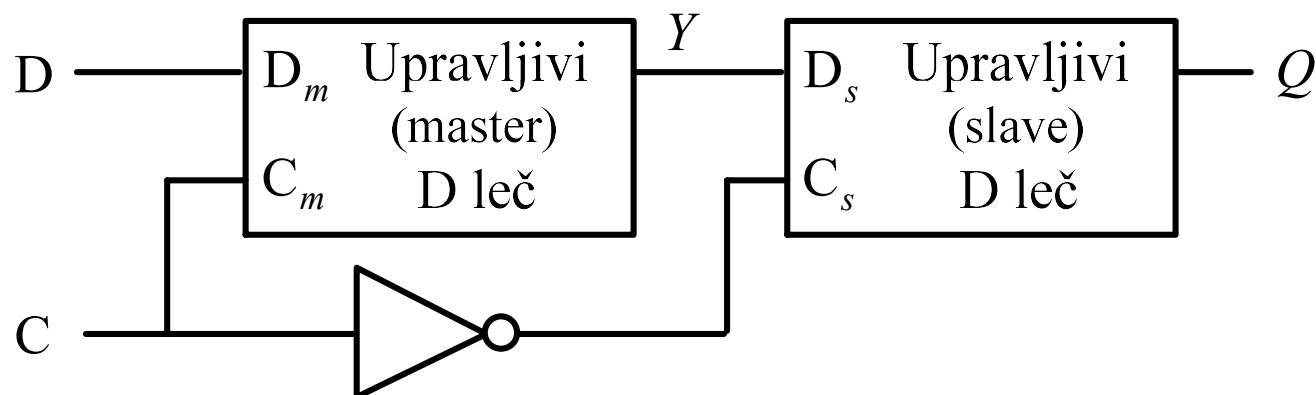
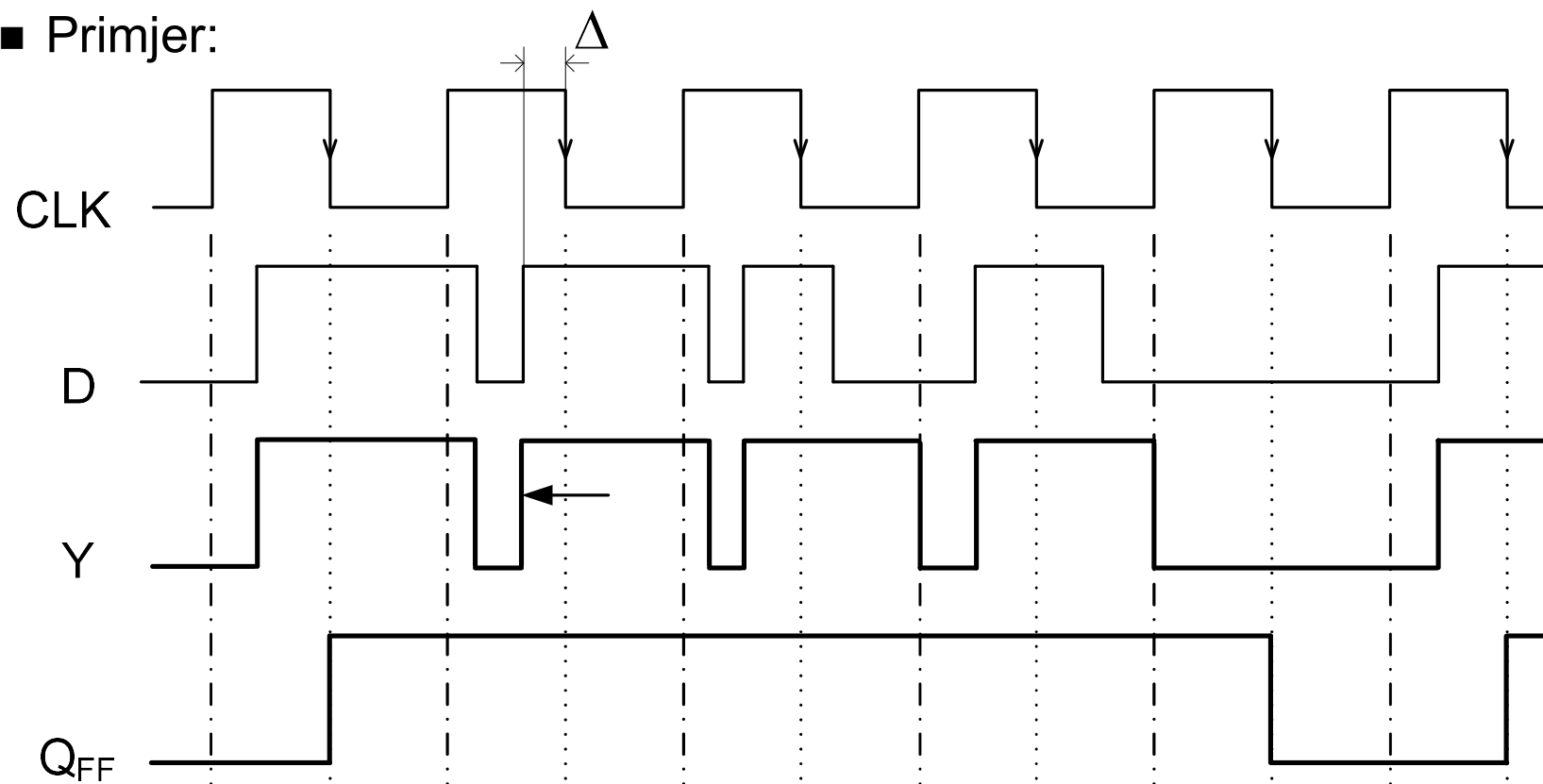


Silazna ivica

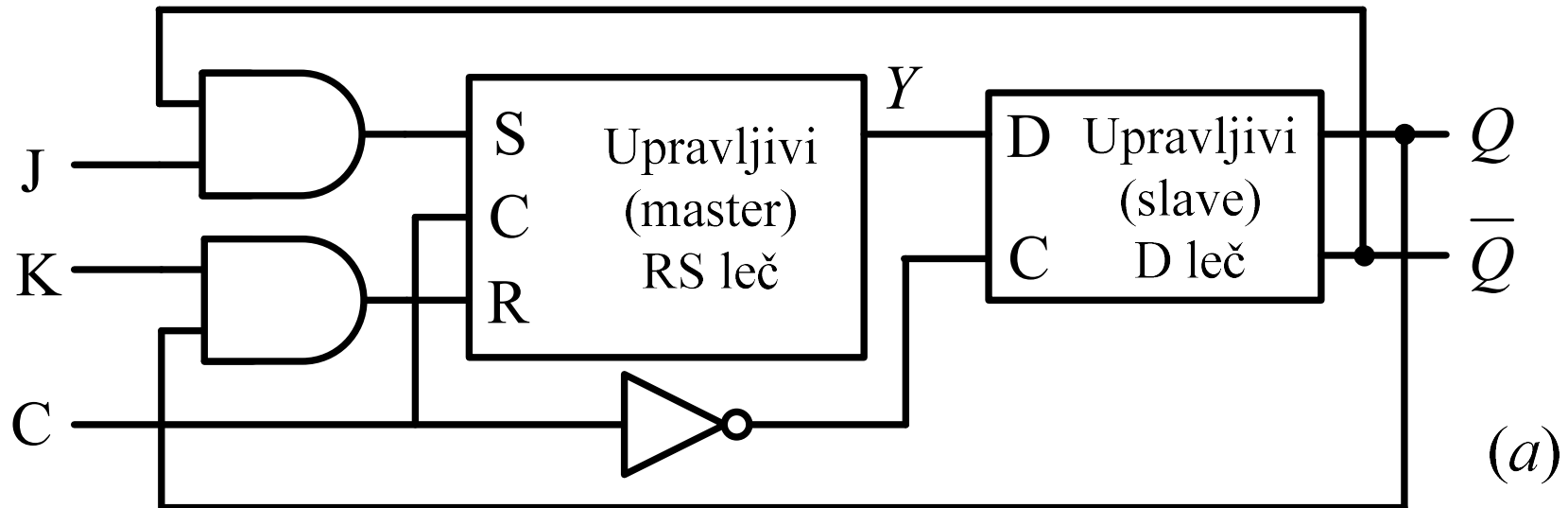


Uzlazna ivica

■ Primjer:

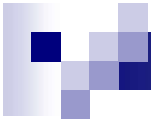


# JK FLIP-FLOP



C	J	K	$Q_{t+1}$	
↓	0	0	$Q_t$	← Zatečeno stanje
↓	0	1	0	← Reset
↓	1	0	1	← Set
↓	1	1	$\bar{Q}_t$	← Komplement

(b)



■ Primjer:

