

Digitális technika 2.

BMEVIIIAA02

előadás

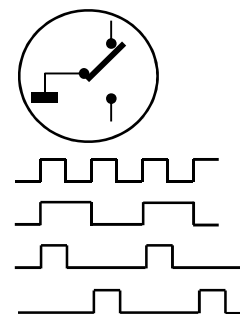
2020/21 tavaszi félév

Számlálók

Sorrendi hálózatok alaphelyzetbe állítása

Flip-flopok alkalmazása

- Nyomógomb pergésmentesítés
- Órajel felezés
- Kétfázisú órajel

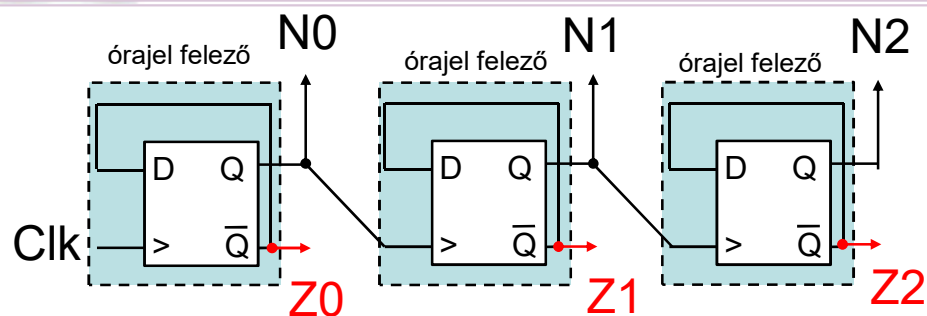


Regiszter, LATCH

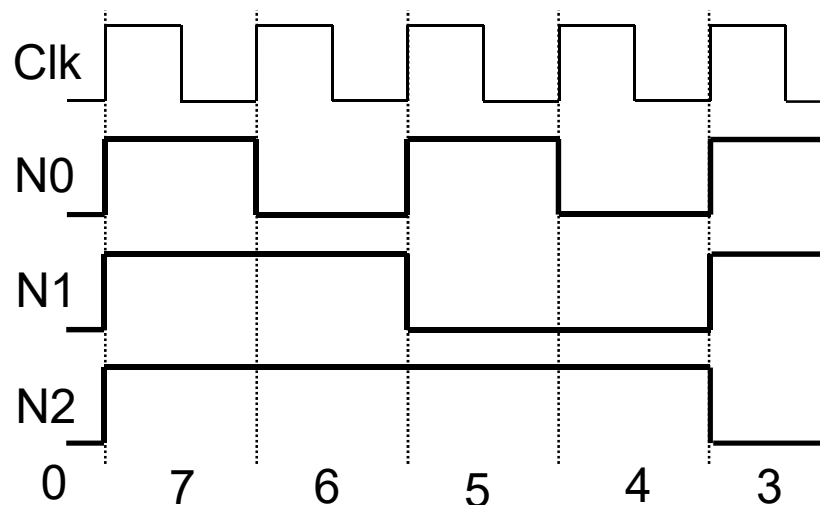
Léptető regiszter

- Soros → párhuzamos átalakítás
- Párhuzamos → soros átalakítás

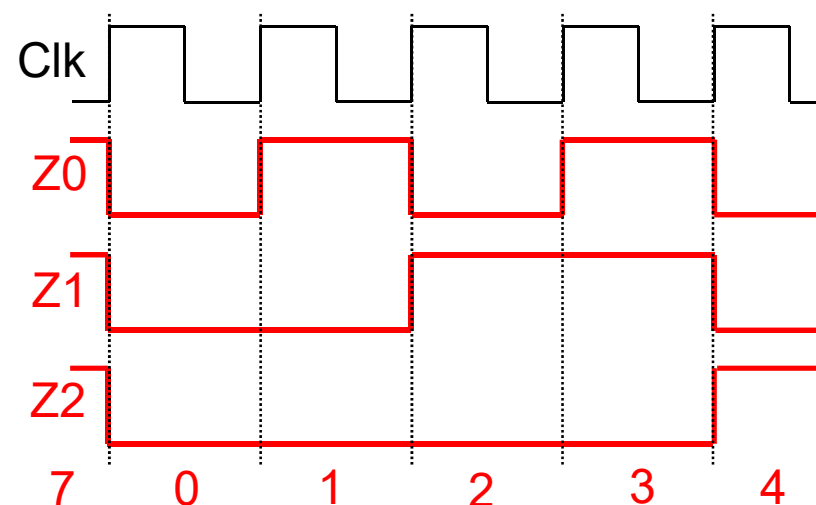
Általános regiszter



Aszinkron számláló

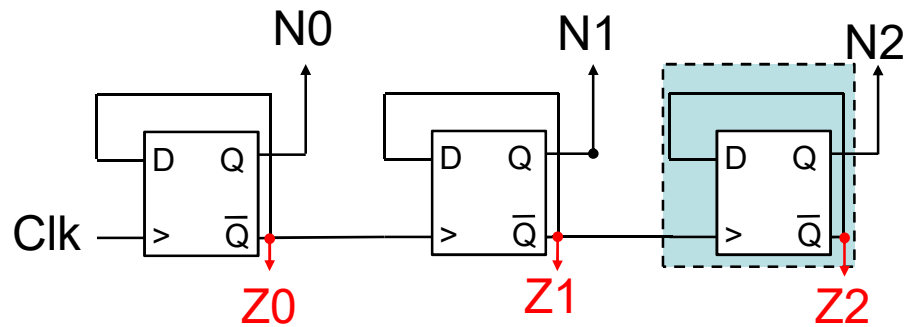


Lefelé számláló

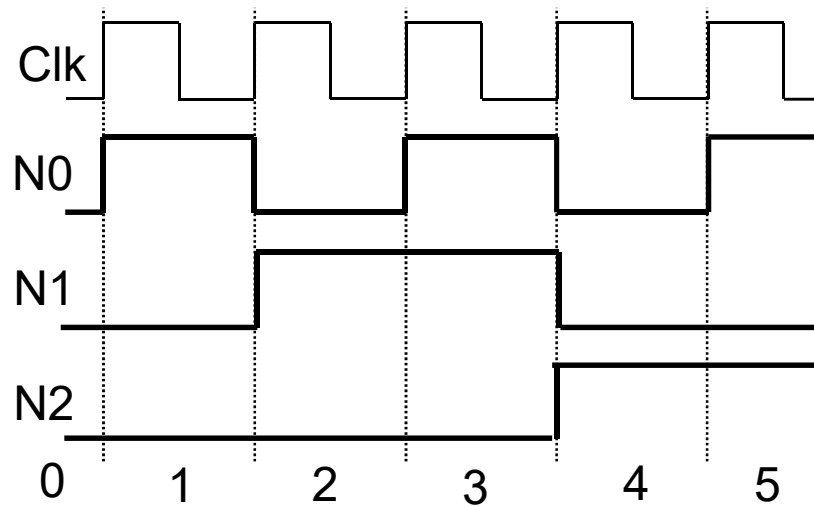
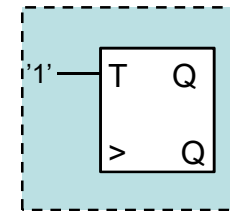


Felfelé számláló

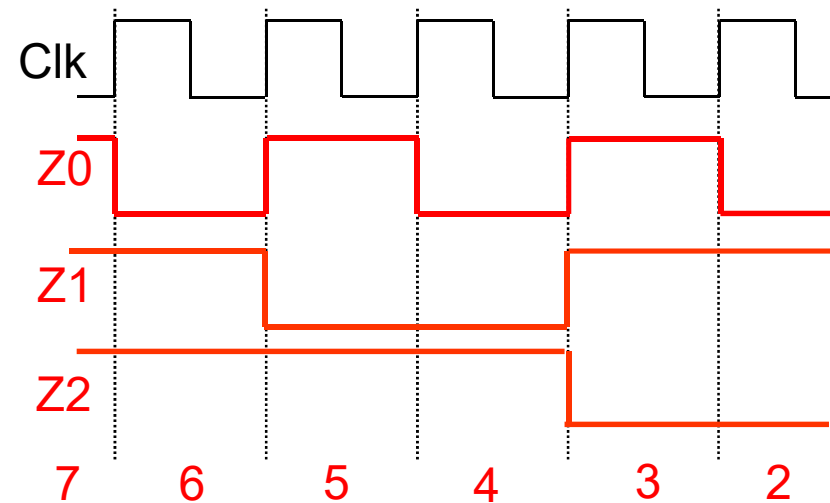
7	111	↔	000	0
6	110	↔	001	1
5	101	↔	010	2
4	100	↔	011	3
3	011	↔	100	4
2	010	↔	101	5
1	001	↔	110	6
0	000	↔	111	7



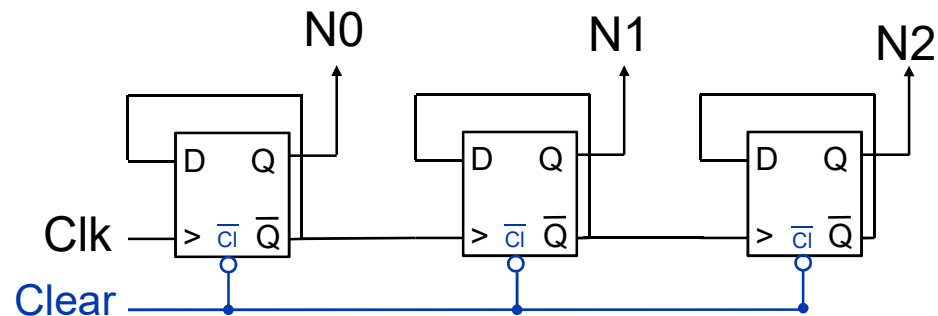
Aszinkron számláló



Felfelé számláló



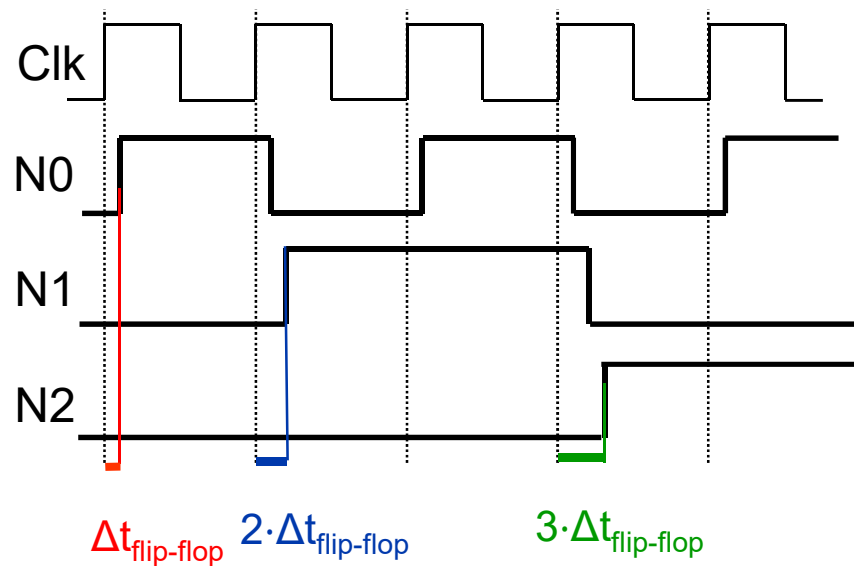
Lefelé számláló



Aszinkron számláló

Aszinkron alaphelyzetbe állítás
(Miért nem lehet szinkron?)

Kimenet érvényessége



n bites számláló: $n \cdot \Delta t_{\text{flip-flop}}$

$\Delta t_{\text{flip-flop}} = 30 \text{ ns}$

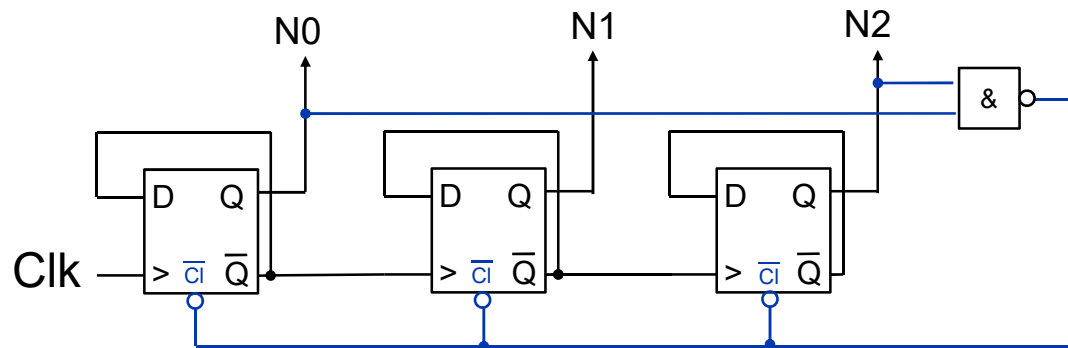
16 bites számláló: 480 ns

Felhasználás: frekvencia osztás

Számlálók

Számlálási ciklus csökkentése 0...4-ig számlálás (5 ciklus)

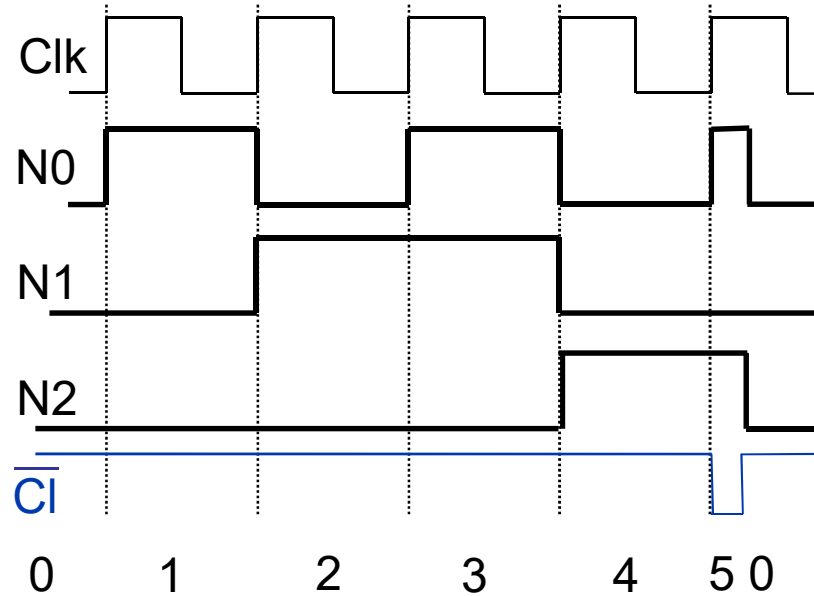
Aszinkron törlés – komparálás az 5-ös értékre



5 értéke binárisan: 101

$$Cl = N0 \cdot N2$$

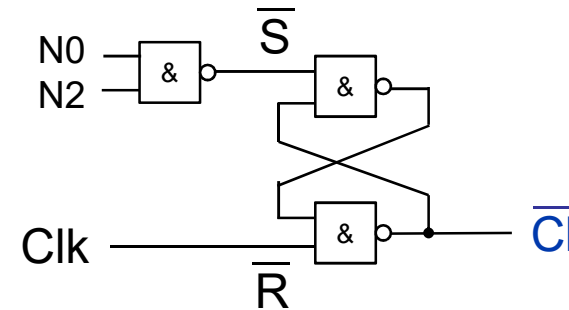
$$\overline{Cl} = \overline{N0 \cdot N2}$$



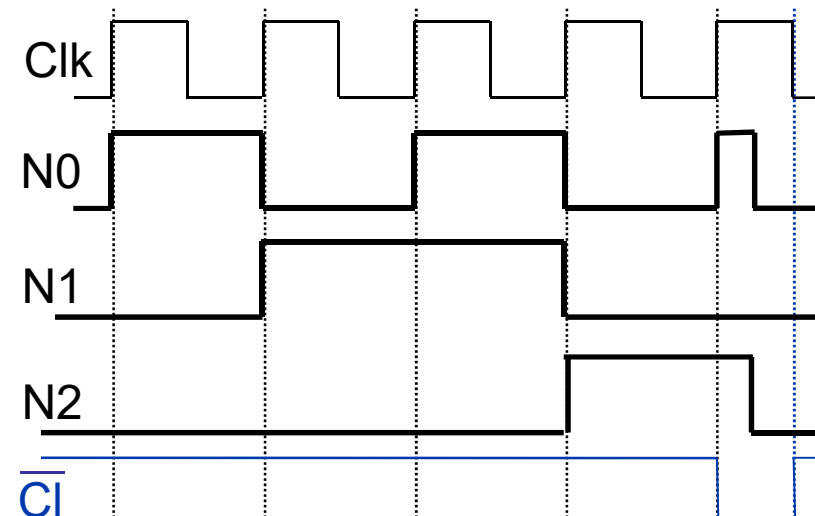
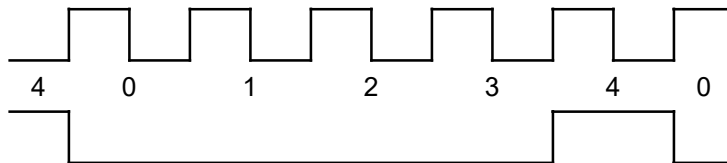
Gond: milyen széles lesz a pulzus?

Számlálási ciklus csökkentése 0...4-ig számlálás (5 ciklus)
Impulzus formálás

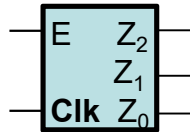
Használjunk SR flip-flopot
Set: Elértük az 5-ös értéket
Reset: Órajel = 0



N2 – kitöltési tényező
5 periódusból 1 periódusig magas
20%



Számlálók



Szinkron számláló, engedélyező bemenettel

E = 1: a számláló számlál

E = 0: tartja az értékét

Állapot tábla (Moore modell)

y \ E	0	1	Z ₂ Z ₁ Z ₀
a	a	b	0 0 0
b	b	c	0 0 1
c	c	d	0 1 0
d	d	e	0 1 1
e	e	f	1 0 0
f	f	g	1 0 1
g	g	h	1 1 0
h	h	a	1 1 1

Állapot kód: Z = y

Kódolt állapot tábla

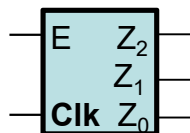
y \ E	0	1
000	000	001
001	001	010
010	010	011
011	011	100
100	100	101
101	101	110
110	110	111
111	111	000

Megvalósítás T flip-floppal

Vezérlési tábla

y \ E	0	1
000	0 0 0	0 0 1
001	0 0 0	0 1 1
010	0 0 0	0 0 1
011	0 0 0	1 1 1
100	0 0 0	0 0 1
101	0 0 0	0 1 1
110	0 0 0	0 0 1
111	0 0 0	1 1 1

$$\begin{aligned} T_0 &= E \\ T_1 &= E \cdot y_0 \\ T_2 &= E \cdot y_0 \cdot y_1 \end{aligned}$$

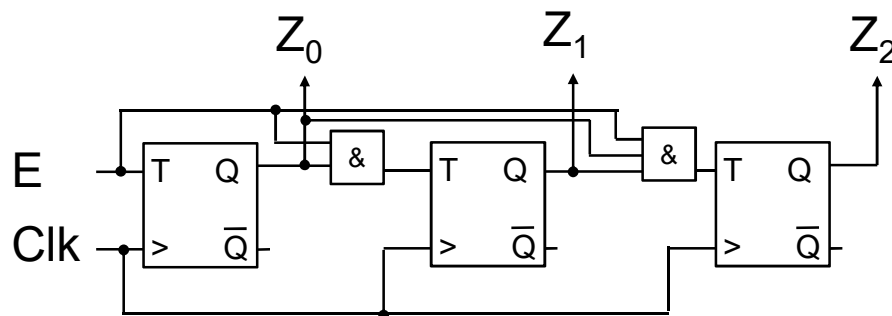


Szinkron számláló

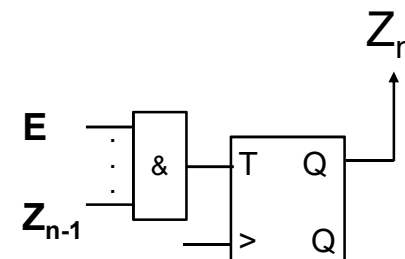
$$T_0 = E$$

$$T_1 = E \cdot y_0$$

$$T_2 = E \cdot y_0 \cdot y_1$$



$$T_n = E \cdot y_0 \cdot y_1 \cdot \dots \cdot y_{n-1}$$



n . flip-flop bemenetén n bemenetű ÉS kapu

Kimenet érvényes: $\Delta t_{\text{flip-flop}}$

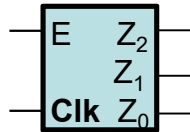
Következő órajel: $\Delta t_{\text{flip-flop}} + \Delta t_{\text{kapu}}$

$$\Delta t_{\text{flip-flop}} = 30 \text{ ns}$$

$$\Delta t_{\text{kapu}} = 10 \text{ ns}$$

$$40 \text{ ns} \rightarrow 25 \text{ MHz}$$

a számláló bitszámától független



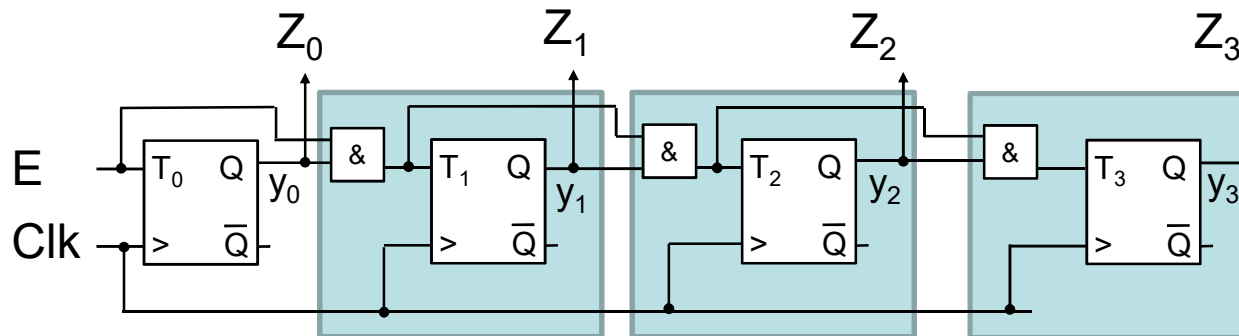
Szinkron számláló

$$T_0 = E$$

$$T_1 = E \cdot y_0$$

$$T_2 = E \cdot y_0 \cdot y_1 = T_1 \cdot y_1$$

$$T_3 = E \cdot y_0 \cdot y_1 \cdot y_2 = T_2 \cdot y_2$$



minden flip-flop bemenetén 2 bemenetű ÉS kapu

Kimenet érvényes: $\Delta t_{\text{flip-flop}}$

Következő órajel: $\Delta t_{\text{flip-flop}} + (n-1) \cdot \Delta t_{\text{kapu}}$

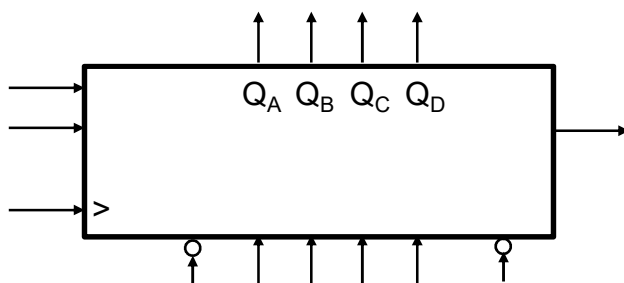
$$\Delta t_{\text{flip-flop}} = 30 \text{ ns}$$

$$\Delta t_{\text{kapu}} = 10 \text{ ns}$$

16 bites számláló

$$180 \text{ ns} \rightarrow 5.5 \text{ MHz}$$

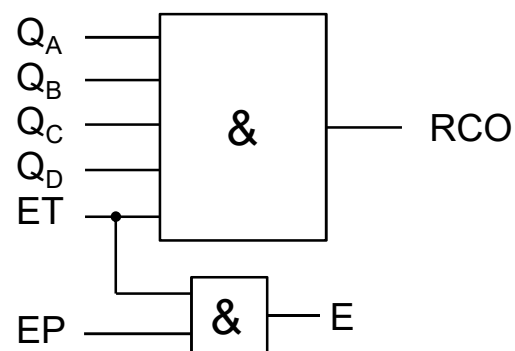
4 bites szinkron bináris számláló



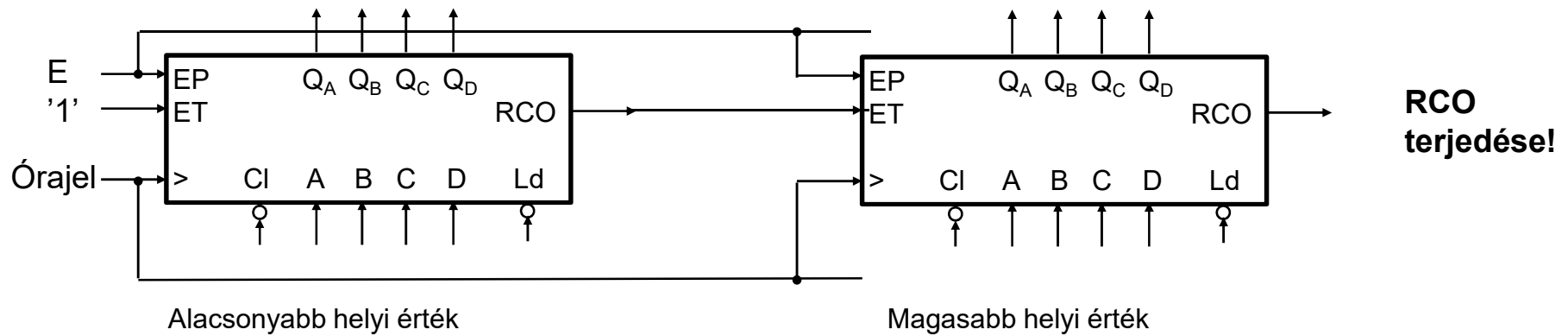
EP,ET: engedélyező bemenetek
 Ld: szinkron töltés
 Cl: aszinkron vagy szinkron törlés
 RCO: telítődés jelzése

Funkcionális igények:

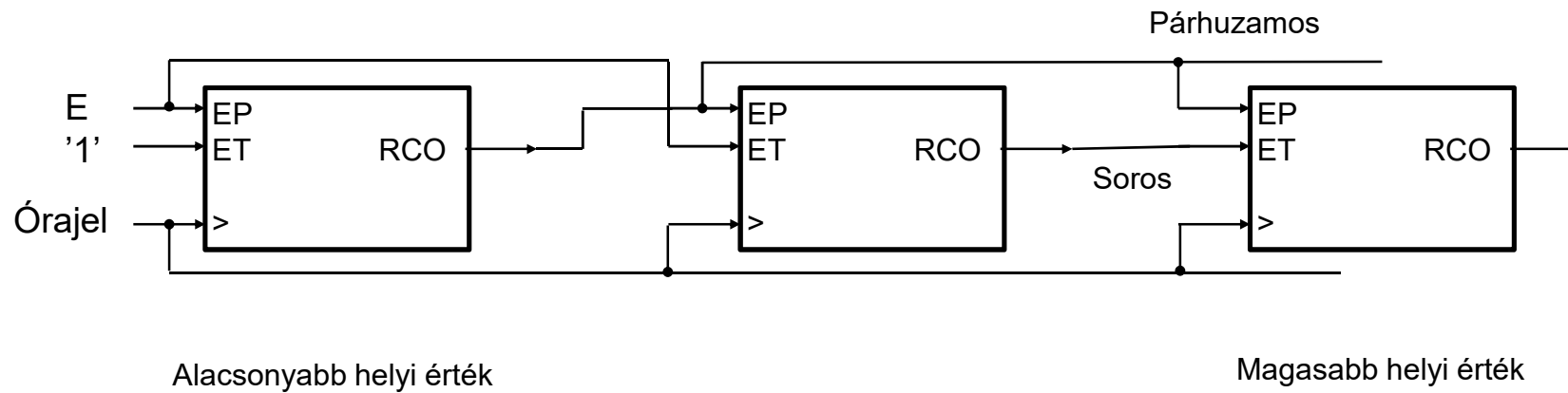
- számlálás engedélyezése
- kezdeti érték adás
 - tetszőleges érték (load)
 - nullázás (clear)
 - aszinkron
 - szinkron
- kaszkádosítás támogatása



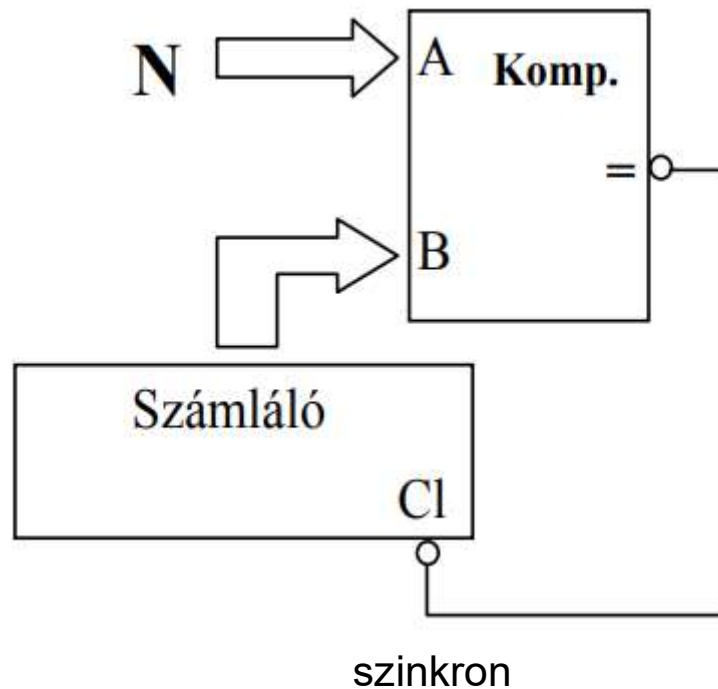
Soros kaszkádosítás



Soros/párhuzamos kaszkádosítás

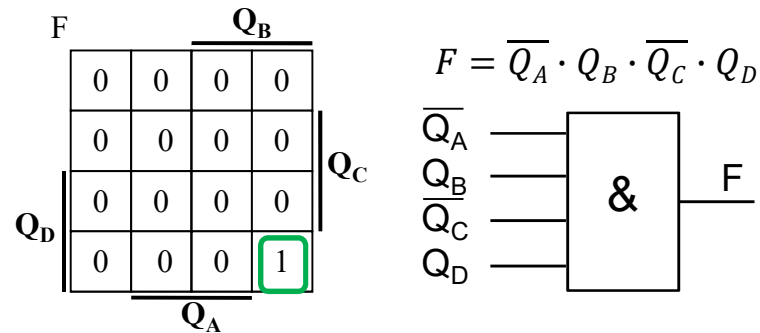


Számlálási ciklus módosítása 0 ... N Szinkron CI

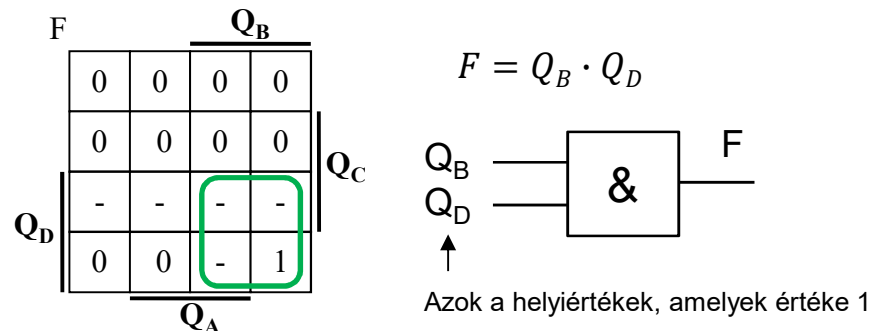


Ha N konstans, a komparátor egyetlen ÉS kapu

Például: $N = 10 \rightarrow$ binárisan: 1010



Kihasználhatjuk, hogy tudjuk 10-nél nagyobb érték nem fordulhat elő



Számlálási ciklus módosítása $0 \dots N$ Aszinkron Cl

Aszinkron törlés nem jó a **hazárd** miatt

Számláló kimenete $7 \rightarrow 8$ binárisan: $0111 \rightarrow 1000$

A flip-flopok ugyan azt az órajelet kapják

Mindegyik flip-flopnak változtatnia kell a kimenetét

A kimenet megváltozásának időpontja az adott flip-flop késleltetésétől függ

→ Tranziensben (rövid ideig) az összes érték előfordulhat

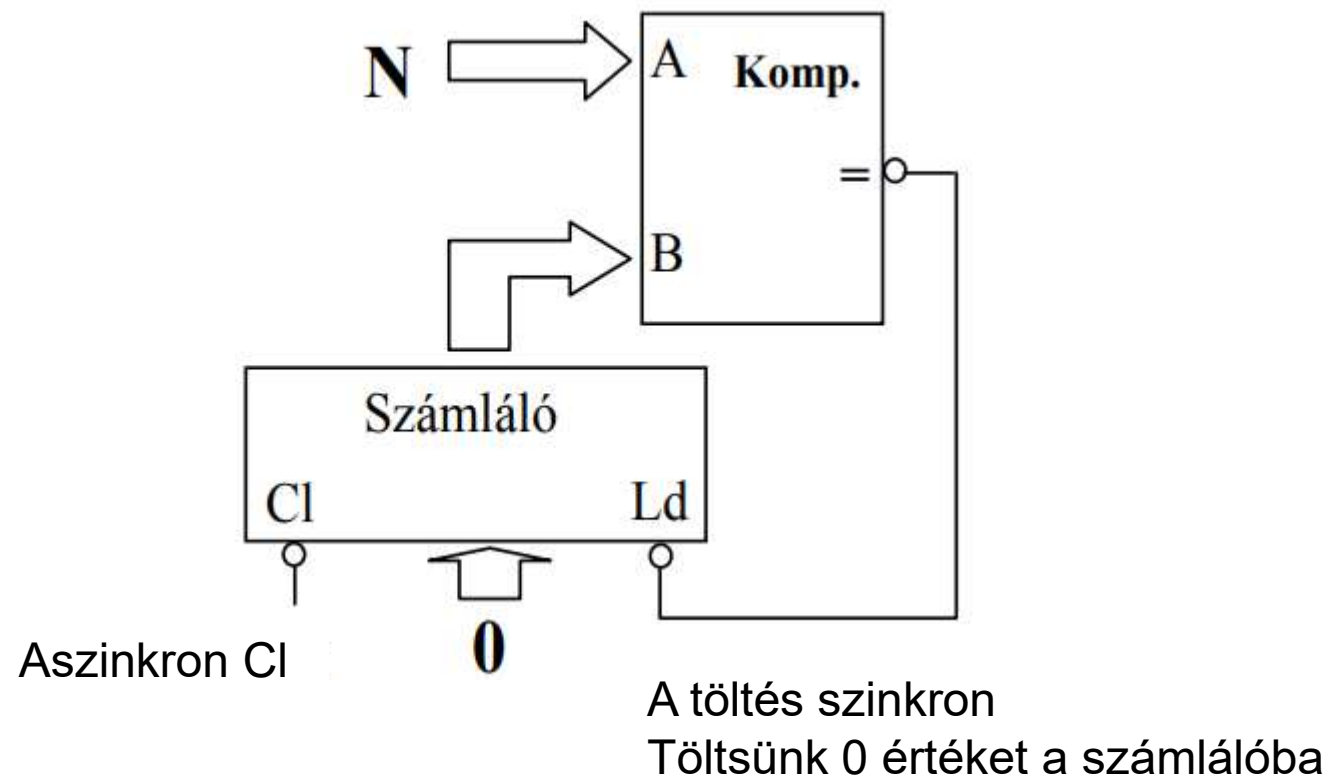
Aszinkron számlálónál miért működik?

A változás sorrendje a struktúra miatt kötött

$0111 \rightarrow 0110 \rightarrow 0100 \rightarrow 0000 \rightarrow 1000$

Számlálási ciklus módosítása $0 \dots N$ Aszinkron CI

Blokk vázlat



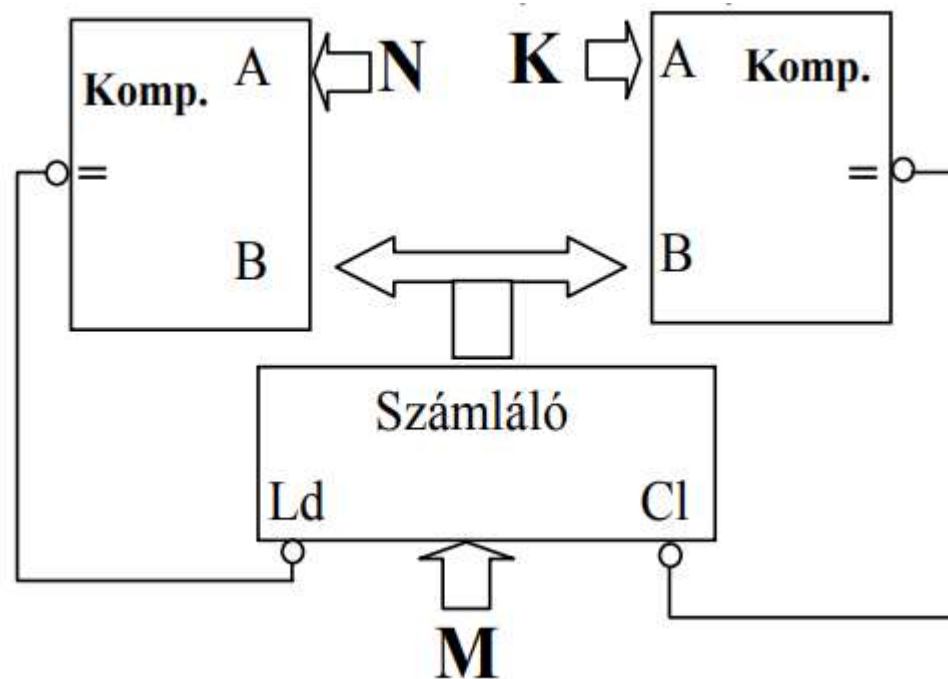
Számlálási ciklus módosítása $0 \dots N, M \dots K$ ($N < M < K$) Szinkron CI

Két komparátor

N-re tölteni kell M-et

K-ra törölni kell a számlálót

Blokk vázlat



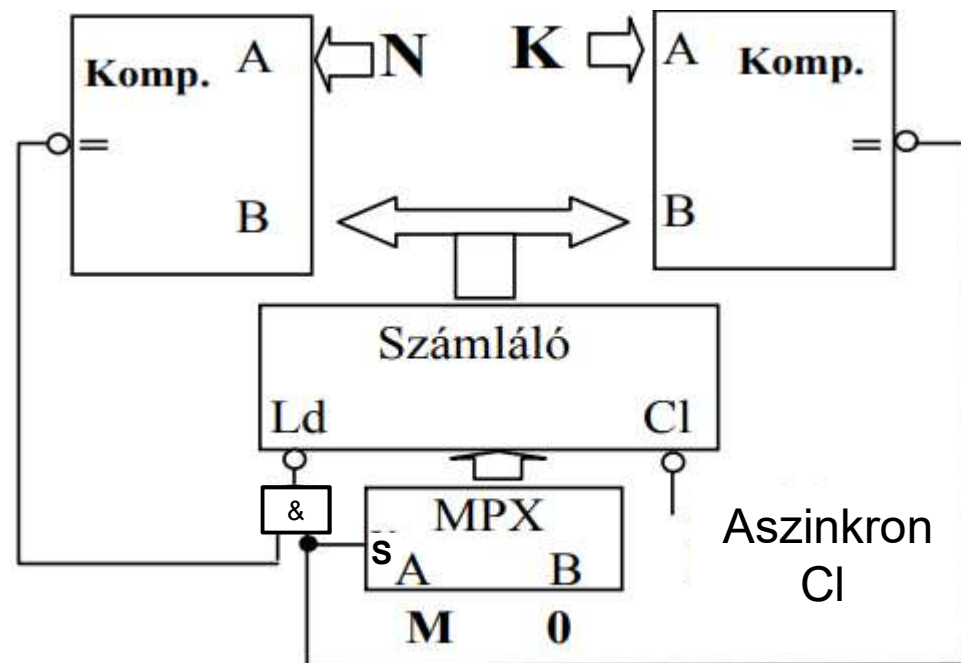
Számlálási ciklus módosítása $0 \dots N, M \dots K$ ($N < M < K$) Aszinkron CI

Törlés nem használható

→ Töltés multiplexerrel

- ha elértük N-t: M-et tölt
- ha elértük K-t: 0-t tölt

Blokk vázlat



Számláló típusok

- Aszinkron
- Szinkron
- Bináris
Modulo N számlálás: $0 \dots N-1$ (n bites számláló, $N = 2^n$)
- BCD
 $0 \dots 9$
- Gray
3 bites: $000 \rightarrow 001 \rightarrow 011 \rightarrow 010 \rightarrow 110 \rightarrow 111 \rightarrow 101 \rightarrow 100 \rightarrow 000 \dots$
- Egyirányú
Számlálási irány megfordítása a kimenet invertálásával
- Reverzibilis
MIN/MAX – számlálási iránytól függő jelzés a következő fokozat felé
- Alaphelyzetbe állítás
 - Törlés (Cl): aszinkron, szinkron
 - Töltés (Ld): szinkron

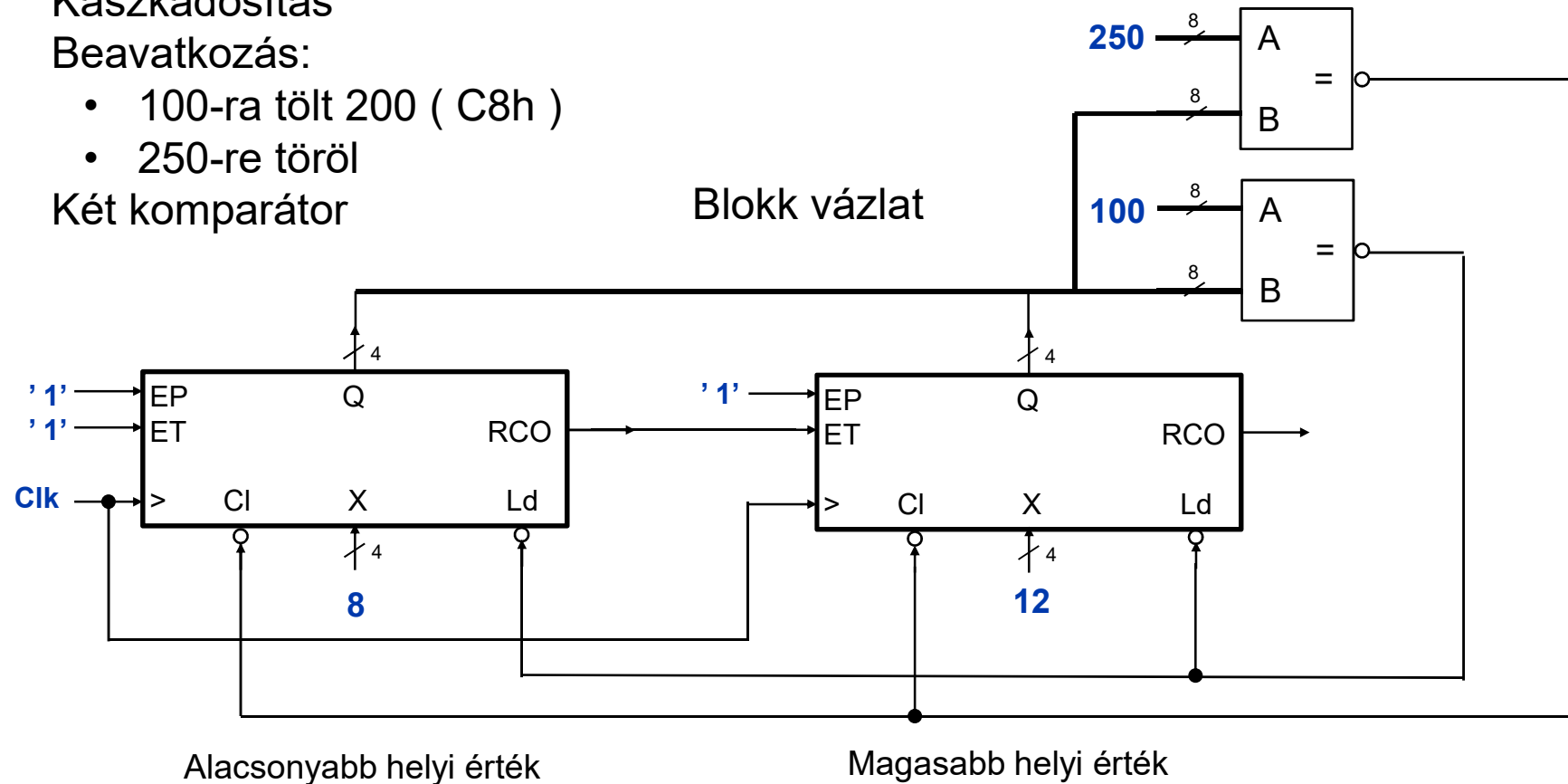
8 bites szinkron törölhető/tölthető számláló és 8 bites komparátor kell

Beavatkozás:

- 100-ra tölt 200 (C8h)
- 250-re töröl

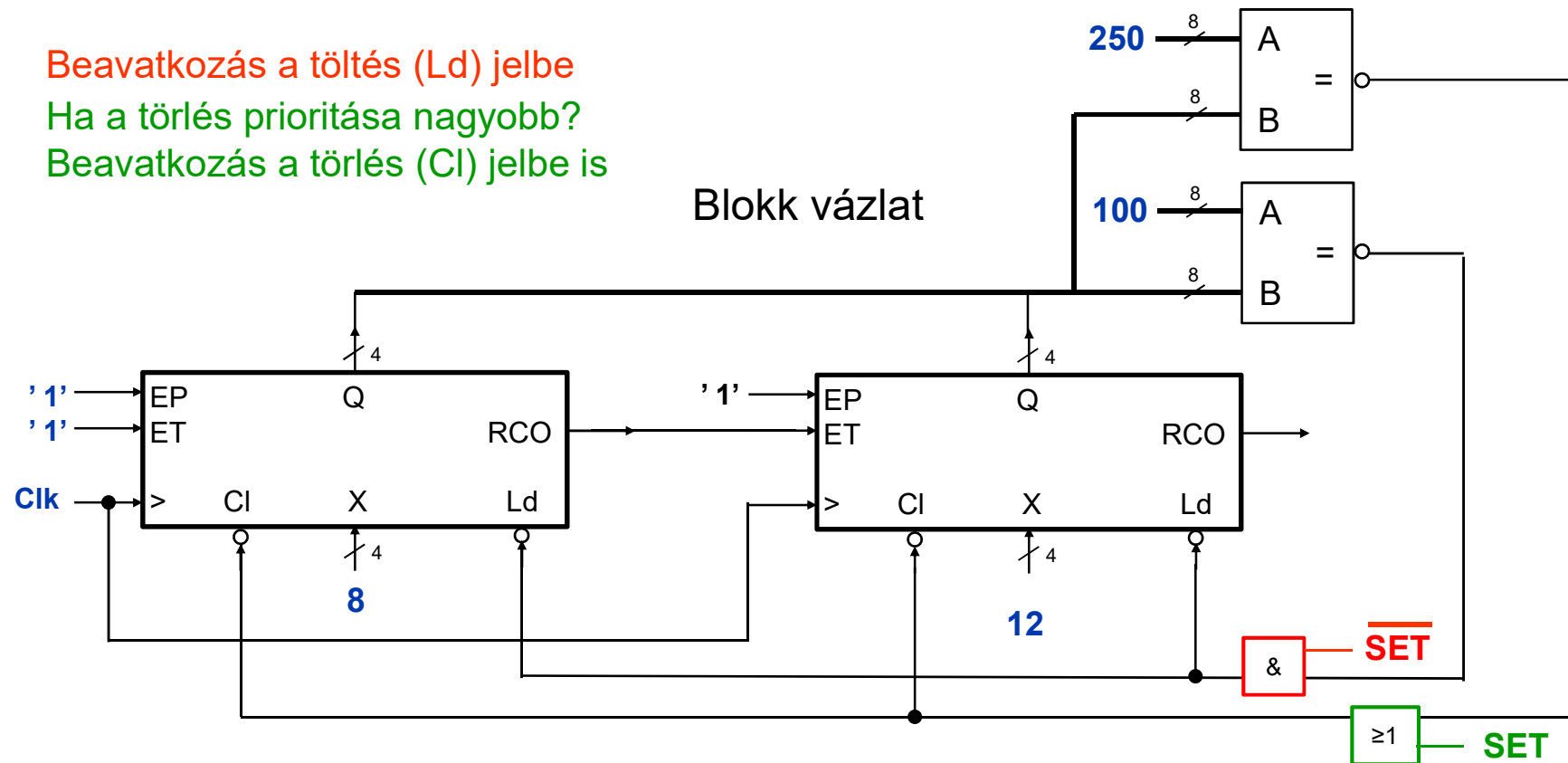
Két komparátor

Blokk vázlat



Legyen egy külső SET jel, amellyel a számláló 200-ról indítható

Beavatkozás a törlés (Cl) jelbe is



Legyen egy külső SET jel, amellyel a számláló 200-ról indítható
Legyen egy külső RESET jel, amellyel a számláló 0-ról indítható

(Cl és Ld közül Cl prioritása nagyobb)

