

Pannon Egyetem

Villamosmérnöki és Információs Rendszerek

Tanszék



Digitális Áramkörök

(Villamosmérnök BSc /
Mechatronikai mérnök MSc)


10-11. hét – Sorrendi hálózatok tervezési lépései:
szinkron – aszinkron sorrendi hálózatok esetén

Előadó: Dr. Vörösházi Zsolt

voroshazi.zsolt@virt.uni-pannon.hu

Kapcsolódó jegyzet, segédanyag:

- <http://www.virt.uni-pannon.hu>
 - Oktatás → Tantárgyak → Digitális Áramkörök (Villamosmérnöki BSc / Mechatronikus BSc/MSc).
- Fóliák, óravázlatok (.ppt)
- Frissítésük folyamatosan



I. Szinkron sorrendi hálózatok tervezési lépéseinek bemutatása

Szinkron sorrendi hálózat főbb tervezési lépései

- 1. Logikai feladat megfogalmazása
 - 2. Előzetes állapottábla összeállítása
 - 3. Egyszerűsített (összevont) állapottábla
 - 4. Állapotkódolás
 - 5. Vezérlési tábla összeállítása
 - 6. Működés szemléltetése idődiagramon
-
- Mindezt *Mealy* ill. *Moore* modelleken is megvizsgálva és megvalósítva.



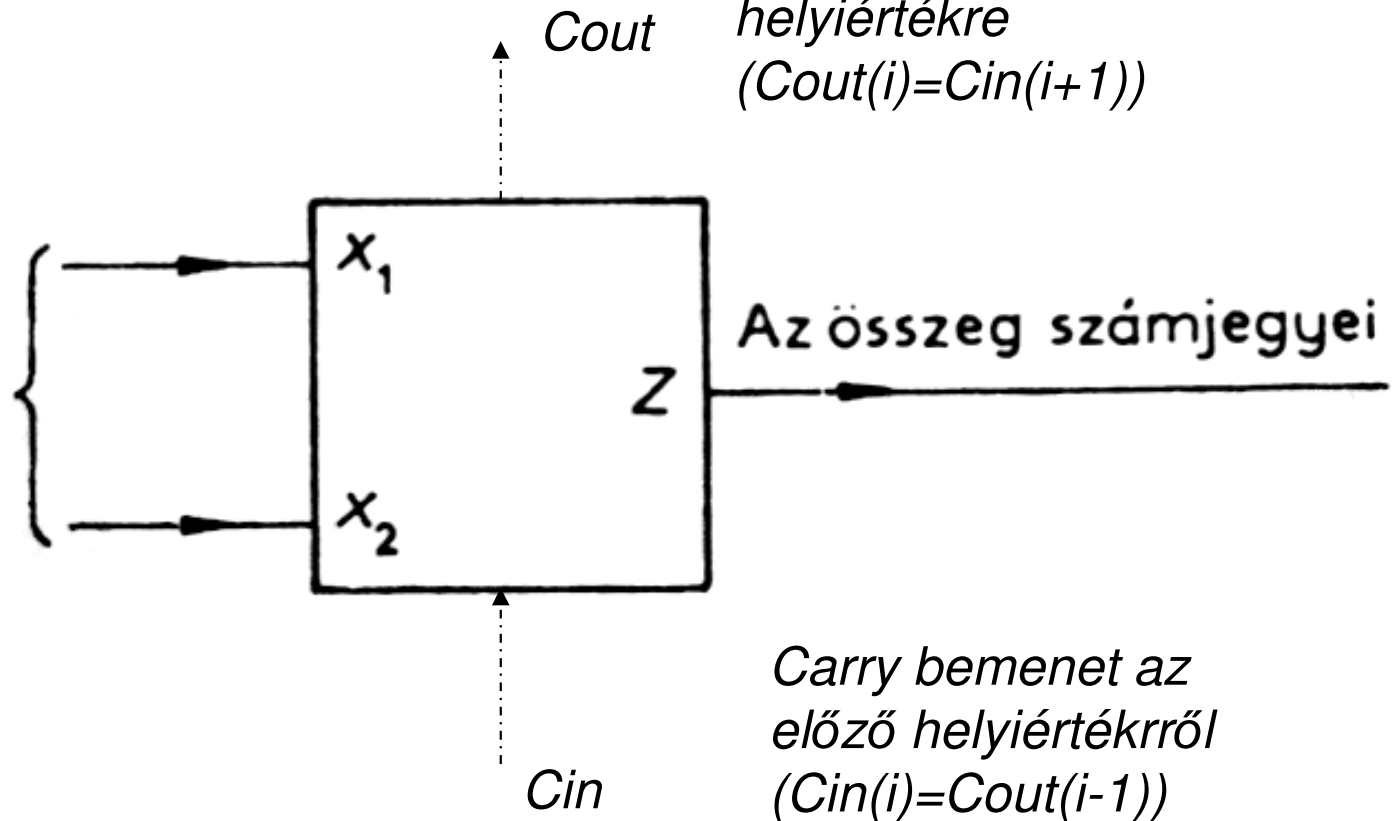
Tervezés: Mealy modell

1.) Logikai feladat megfogalmazása

- Tervezzünk egy **soros összeadót**, mint szinkron S.H.-ot, melynek 2 bemenete (X_1 , X_2) és egy kimenete van (Z). Bemenetekre, illetve a kimenetre órajel ütemezéssel érkezzenek az adatok.
 - 1-bites Full Adder realizációja
- Soros összeadó: **MSB** ← **LSB** felé haladva sorban érkeznek az összeadandó adatok (mint bináris számjegyek).
- Átvitelt (Carry) meg kell oldani az egyes helyiértékek között. Emlékeztető (Gyakorlat):
 - **Half Adder**: ha adott helyiértéken nem kezeljük a carry input-ot (C_{in}), de a kimeneti értéket (Z) és az átvitelt (C_{out}) előállítjuk.
 - **Full Adder**: carry be-,és kimeneteket is kezeljük, ill. generáljuk.

Soros összeadó

*A két összeadandó
bináris szám
összetartozó
helyiértékei*



$$Z = X_1 + X_2 + (C_{in})$$

2.) Előzetes állapottábla összeállítása (Mealy)

- A S.H.-nak különbözőképpen kell reagálnia ugyanarra a bemeneti kombinációra a megelőző bemeneti kombinációktól és a hálózat állapotától függően.
- Kérdés: min. mennyi állapot kell?
 - „Előzetes” állapottáblában ált. **több** állapot szerepel a szükségesnél.
 - Kimenet **pillanatnyi** értéke (Z)
 - **Előző** helyiértéken keletkezett átvitel (carry) értéke ($C_{in} \leftarrow C_{out}$)

Állapotoknak megfelelő szekunder kombinációk megadása:

Állapot jele (y)	Az előző helyiértéken keletkezett átvitel értéke (C _{in} =C _{out})	Z: A kimenet pillanatnyi értéke
a	0	0
b	0	1
c	1	0
d	1	1

↑
y: Szekunder kombinációk

Megj: soros összegzéskor a számított kimeneti érték független a szekunder kombinációkhoz tartozó a pillanatnyi kimeneti értéktől, csak a **bemeneti** kombinációktól, illetve az **előző** helyiértéken keletkezett átviteltől függ.

Előzetes állapottábla összeállítása

- Állapottábla: 4 sora (y), 4 oszlopa van (X_1X_2)

x_1x_2 y	00	01	11	10
a	yz $a0$	$b1$	$c0$	$b1$
b	$a0$	$b1$	$c0$	$b1$
c	$b1$	$c0$	$d1$	$c0$
d	$b1$	$c0$	$d1$	$c0$

Mealy modell:
soronként változó Z
kimeneti értékek

Kitöltés: (1. sor) *Kiindulási állapot bárhol lehet!*

•Tfh. $y=a$ és $X_1X_2 = '00'$ → $X_1+X_2+Cin = '0'$, és $Cout = '0'$ → Ekkor $Y='a'$ állapot és $Z='0'$ kimenetre: **a0**

•Tfh. $y=a$ és $X_1X_2 = '01'$ v $'10'$ → $X_1+X_2+Cin = '1'$ és $Cout = '0'$ → Ekkor $Y='b'$ állapot és $Z='1'$ kimenetre: **b1**

•Tfh. $y=a$ és $X_1X_2 = '11'$ → $Z=X_1+X_2+Cin = '10'$ és $Cout = '1'$ → Ekkor $Y='c'$ állapot és $Z='0'$ kimenetre: **c0**

Kitöltés: (4. sor) ...

•Tfh. $y=d$ és $X_1X_2 = '00'$ → $X_1+X_2+Cin = '1'$, és $Cout = '0'$ → Ekkor 'b' állapot és $Z='1'$ kimenetre: **b1**

•Tfh. $y=d$ és $X_1X_2 = '01'$ v $'10'$ → $X_1+X_2+Cin = '10'$ és $Cout = '1'$ → Ekkor 'c' állapot és $Z='0'$ kimenetre: **c0**

•Tfh. $y=d$ és $X_1X_2 = '11'$ → $Z=X_1+X_2+Cin = '11'$ és $Cout = '1'$ → Ekkor 'd' állapot és $Z='1'$ kimenetre: **d1**

3.) Összevont (egyszerűsített) állapotábra

- Előzetes állapotábra alapján cél, hogy megkeressük a lehető *legkevesebb megkülönböztetendő állapotot*.
- Ezek ismeretében kell létrehozni a lehető *legkevesebb, de azonos sort nem tartalmazó állapotábrát*, amelyet **összevont állapotábrának** nevezünk.
 - *nincs feleslegesen megkülönböztetett állapota (=sora)*

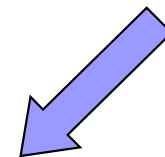
Összevont (egyszerűsített) állapotábra és állapotgráf

Előzetes állapotábra:

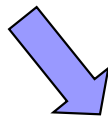
x_1x_2 y	00	01	11	10
a	$a0$	$b1$	$c0$	$b1$
b	$a0$	$b1$	$c0$	$b1$
c	$b1$	$c0$	$d1$	$c0$
d	$b1$	$c0$	$d1$	$c0$

■ **Állapot összevonások (szekunder állapotok):**

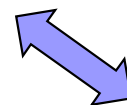
- $A := a, b$ (1.-, és 2. sor azonos)
- $B := c, d$ (3.-, és 4. sor azonos)



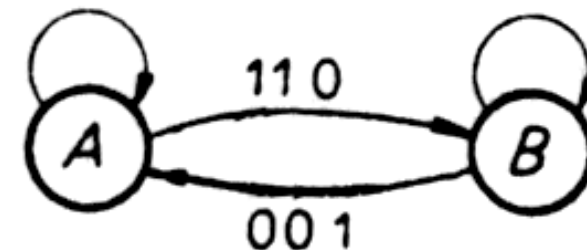
Összevont állapotábra:



x_1x_2 y	00	01	11	10
A	$A0$	$A1$	$B0$	$A1$
B	$A1$	$B0$	$B1$	$B0$



$x_1x_2 Z$
00 0 ; 01 1 ; 10 1 01 0 ; 11 1 ; 10 0



4.) Állapotkódolás

- Összevont állapottábla alapján a szimbolikusan összevonáshoz (A,B) jelölt állapotoknak meg kell feleltetni egy-egy szekunder kombinációt (y).
- Összevont állapottábla *sorainak (=állapotainak) számától függ*, hogy mennyi **szekunder** állapotot kell felvenni.
 - n sor esetén: $\lceil \log_2(n) \rceil$ szekunder állapot szükséges

Állapotkódolás (folyt.)

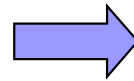
- Felvett állapotkódok: két összevont állapot
→ egyetlen szekunder (y) állapot (két lehetséges értékkel)
 - Legyen $y := '0'$, ha 'A' állapot
 - Legyen $y := '1'$, ha 'B' állapot(ez a fenti kódolás csak *egy lehetséges módja*, lehetett volna fordítva is definiálni, és egyáltalán **nem közömbös**, miként vesszük fel az állapotkódokat!)

Állapotkódolás (folyt.)

■ Kódolt állapottábla felírása:

*Összevont
állapottábla:*

x_1x_2 y	00	01	11	10
A	yz A0	A1	B0	A1
B	A1	B0	B1	B0



*Kódolt
állapottábla:*

x_1x_2 y	00	01	11	10
0	yz 00	01	10	01
1	01	10	11	10

$A := y = 0$

$B := y = 1$

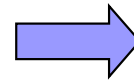
4.) Kimeneti függvény Karnaugh táblája (folyt.)

- Kódolt állapotábla felírása (Z) kimenetre:

Kódolt állapotábla:

x_1x_2 y	00	01	11	10
0	00	01	10	01
1	01	10	11	10

Soros összegző kódolt állapotáblája



Z

		x_2			
	x_1x_2	x_1			
y		00	01	11	10
0		0 ₀	1 ₁	0 ₃	1 ₂
1		1 ₄	0 ₅	1 ₇	0 ₆

$$Z = X_1 \oplus X_2 \oplus y \Rightarrow S^3_{1,3}(X_1, X_2, y)$$

$$f_z(X, y) \Rightarrow Z$$

Mealy leképezést megvalósító függvény

5.) Vezérlési tábla összeállítása

- Minden egyes szekunder (y) változóhoz hozzá kell rendelni egy Flip-flop-ot (pl. építőelem készletből). *FF-k működésének* ismeretében meghatározhatjuk, miként kell *vezérelni* azokat egy K.H. segítségével a felírt kódolt állapotábra alapján (itt egyetlen FF, mivel egy szekunder változó volt). Így kapjuk meg a **vezérlési táblát**.
 - Előfordulhat, ha több szekunder változónk van, hogy *eltérő* FF-kat rendelünk egy-egy változó szekunder állapot tárolásához!
 - FF-”Paraméter tábla” használata!

5.a.) Vezérlési tábla összeállítása – S-R tároló segítségével (folyt.)

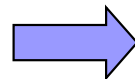
- Vezérlési tábla felírása (szekunder komb.):

*Kódolt állapottábla
(megvalósítandó):*

x_1x_2 y	00	01	11	10
0	yz 00	01	10	01
1	01	10	11	10

*S-R Vezérlési tábla
(megvalósító):*

x_1x_2 y	00	01	11	10
0	S R 0 -	0 -	1 0	0 -
1	0 1	- 0	- 0	- 0



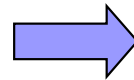
$yY = '00'$

Leolvasás a 3.31 „paraméter tábla” S-R sora alapján adott yY kombinációkra

5.a) Vezérlési tábla alapján (S-R **) Karnaugh táblák felírása

Vezérlési
tábla:

$x_1 x_2 \backslash y$	00	01	11	10
0	$\begin{matrix} S \\ 0- \end{matrix}$	$\begin{matrix} R \\ 0- \end{matrix}$	10	$\begin{matrix} R \\ 0- \end{matrix}$
1	$\begin{matrix} S \\ 01 \end{matrix}$	$\begin{matrix} R \\ -0 \end{matrix}$	$\begin{matrix} R \\ -0 \end{matrix}$	$\begin{matrix} R \\ -0 \end{matrix}$



S

	x_1			
			1	
y		-	-	-
	x_2			

R

	x_1			
	-	-		-
y	1			
	x_2			

Leolvasott DNF alakok:

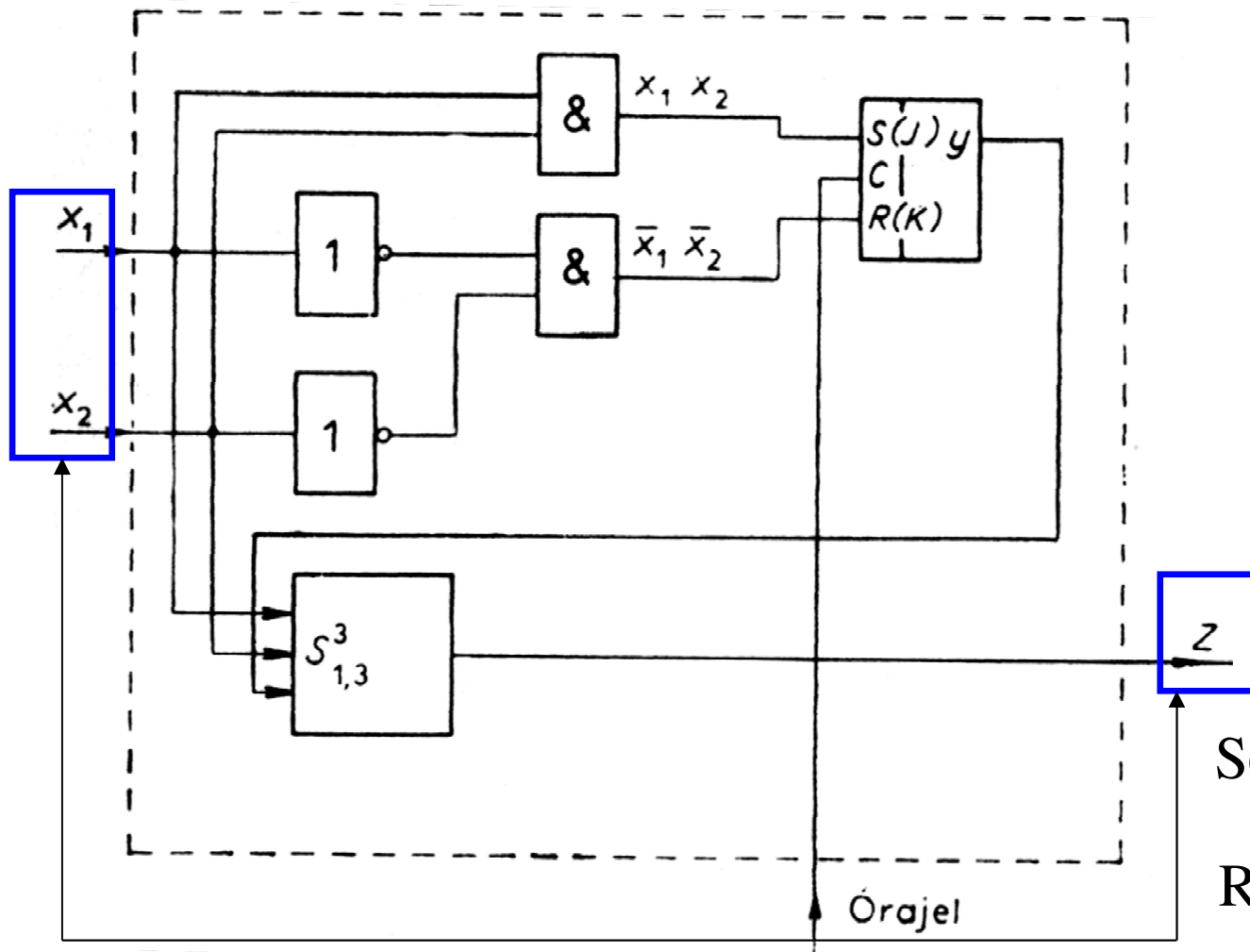
$$S = f_S(x_1, x_2, y) = x_1 \cdot x_2$$

$$R = f_R(x_1, x_2, y) = \overline{x_1} \cdot \overline{x_2}$$

$$Z = x_1 \oplus x_2 \oplus y \Rightarrow S_{1,3}^3(x_1, x_2, y)$$

***Megj: itt a **J-K tárolóval történő FF megvalósítás esetén is ugyanezt kaptuk volna** (mivel J-K értékek éppen olyan helyen adódtak '00' és '11', ahol a vezérlő K.H. legegyszerűbb DNF alakját nem befolyásolják)

6.a.) Szinkron soros összeadó ← S-R (vagy J-K) FF felhasználásával felépített elvi logikai rajza



*Z kimenet közvetlenül függ a bemenetektől és az állapottól is. Tehát **Mealy modell** adódott a Karnaugh egyszerűsítések és az ábra alapján is.*

$$S(= J) = f_S(x_1, x_2, y) = x_1 \cdot x_2$$

$$R(= K) = f_R(x_1, x_2, y) = \overline{x_1} \cdot \overline{x_2}$$

$$Z = x_1 \oplus x_2 \oplus y \Rightarrow S^3_{1,3}(x_1, x_2, y)$$

5.b.) Vezérlési tábla összeállítása – T tároló segítségével

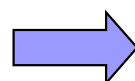
- Vezérlési tábla felírása (szekunder komb.):

*Kódolt állapottábla
(megvalósítandó):*

x_1x_2 y	00	01	11	10
0	00	01	10	01
1	01	10	11	10

*T Vezérlési tábla
(megvalósító):*

x_1x_2 y	00	01	11	10
0	T 0	0	1	0
1	1	0	0	0



$yY='00'$

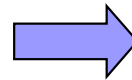
Leolvasás a 3.31 „paraméter tábla” T sora alapján adott yY kombinációkra

5.b.) Vezérlési tábla alapján (T **)

Karnaugh tábla felírása

Vezérlési
tábla:

x_1x_2 y	00	01	11	10
0	0	0	1	0
1	1	0	0	0



T

		x_2			
		x_1		x_1	
y	x_1x_2	00	01	11	10
0		0 ₀	0 ₁	1 ₃	0 ₂
1		1 ₄	0 ₅	0 ₇	0 ₆

Leolvasott DNF alakok:

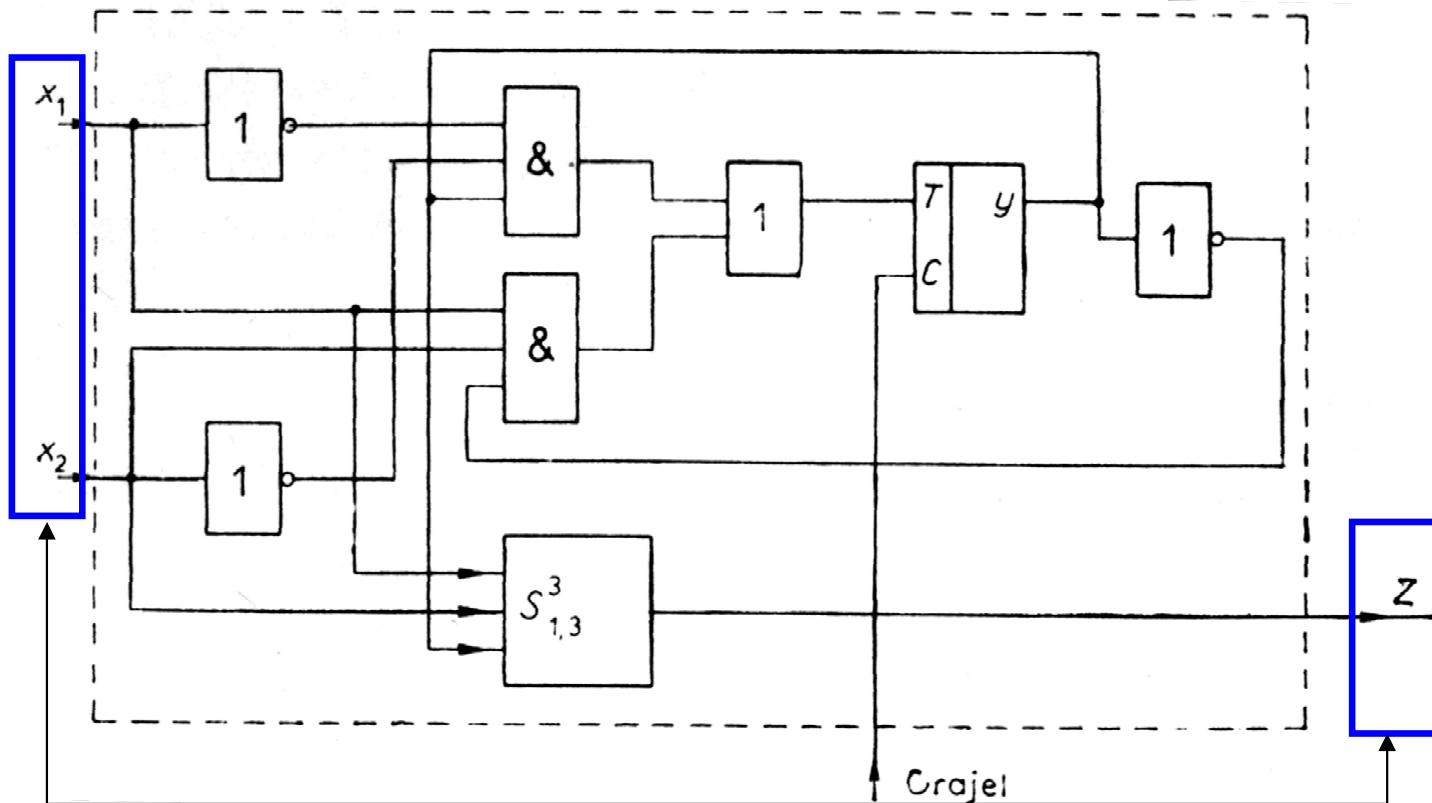
$$T = \overline{\overline{x_1} \overline{x_2} y} + x_1 x_2 \overline{y}$$

$$Z = x_1 \oplus x_2 \oplus y \Rightarrow S_{1,3}^{n=3}(x_1, x_2, y)$$

//a „Z”-t 4.) alapján kaptuk

*****Megj:** T tárolóval történő megvalósítás bonyolultabb lesz, mint az előző S-R vagy J-K tároló esetén volt.

6.b.) Szinkron soros összeadó ← T FF felhasználásával felépített elvi logikai rajza



*Z kimenet közvetlenül függ a bemenetektől és az állapottól is. Tehát **Mealy modell** adódott a Karnaugh egyszerűsítések és az ábra alapján is.*

$$T = \overline{x_1} \overline{x_2} y + x_1 x_2 \overline{y}$$

$$Z = x_1 \oplus x_2 \oplus y \Rightarrow S_{1,3}^{n=3}(x_1, x_2, y)$$

5.c.) Vezérlési tábla összeállítása – D-G tároló segítségével

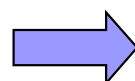
- Vezérlési tábla felírása (szekunder komb.):

*Kódolt állapottábla
(megvalósítandó):*

x_1x_2 y	00	01	11	10
0	yZ 00	01	10	01
1	01	10	11	10

*D-G Vezérlési tábla
(megvalósító):*

x_1x_2 y	00	01	11	10
0	D G 0 0 - 0	0 - - 0	11	0 - - 0
1	01	- 0 1 -	- 0 1 -	- 0 1 -



$yY='00'$

Leolvasás a 3.31 „paraméter tábla” **D-G** sora alapján adott yY kombinációkra

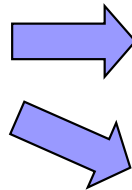
5.c.) Vezérlési tábla alapján (D-G **)

Karnaugh táblák felírása

Vezérlési tábla:

$x_1 x_2$ y	00	01	11	10
0	00 <u>0</u> <u>0</u>	0- -0	11	0- -0
1	01	-0 1-	-0 1-	-0 1-

Intuitív módszer!
(optimális összevonásra törekszünk)



D

	x_1			
y	0	1	0	1
0	<u>0</u>	<u>0</u>	1	<u>0</u>
1		<u>1</u>	<u>1</u>	<u>1</u>

x_2

G

	x_1			
y	0	1	0	1
0	<u>0</u>	<u>0</u>	1	<u>0</u>
1	<u>1</u>	<u>0</u>	<u>0</u>	<u>0</u>

x_2

Leolvasott DNF alakok:

$$D = x_1 y + x_2$$

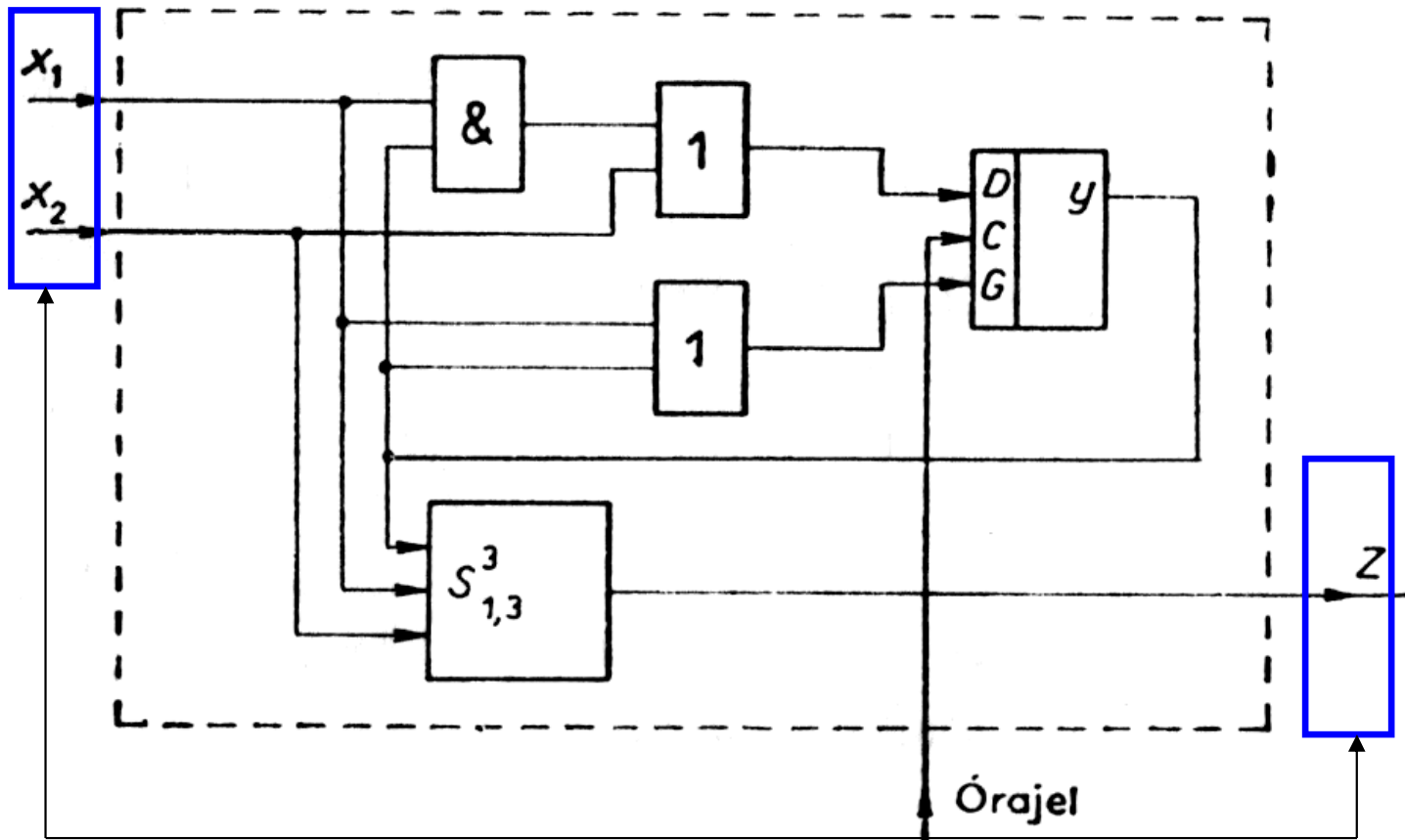
$$G = x_1 + y$$

$$Z = x_1 \oplus x_2 \oplus y \Rightarrow S_{1,3}^3(x_1, x_2, y)$$

//Z-t előző alapján kaptuk

***Megj: SR-JK tárolóval történő megvalósításnál is egyszerűbb megoldás lesz, de intuitív/próbálgatásos módszer az egyes rovatok alsó/felső értékeinek a megválasztása

6.c.) Szinkron soros összeadó ← D-G FF felhasználásával felépített elvi logikai rajza



*Z kimenet közvetlenül függ a bemenetektől és az állapottól is. Tehát **Mealy modell** adódott a Karnaugh egyszerűsítések és az ábra alapján is.*

$$D = x_1 y + x_2$$

$$G = x_1 + y$$

$$Z = x_1 \oplus x_2 \oplus y \Rightarrow S^3_{1,3}(x_1, x_2, y)$$

5.d.) Vezérlési tábla összeállítása – D tároló segítségével

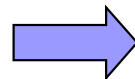
- Vezérlési tábla felírása (szekunder komb.):

*Kódolt
állapottábla:*

x_1x_2 y	00	01	11	10
0	yZ 00	01	10	01
1	01	10	11	10

*Vezérlési
tábla:*

x_1x_2 y	00	01	11	10
0	D 0	0	1	0
1	0	1	1	1



$yY='00'$

*Leolvasás a 3.31 „paraméter
tábla” D sora alapján adott yY
kombinációkra*

5.d.) Vezérlési tábla alapján Karnaugh táblák felírása

Vezérlési
tábla:

x_1x_2 y	00	01	11	10
0	0	0	1	0
1	0	1	1	1



D

		x_2			
		x_1		x_1	
y	x_1x_2	00	01	11	10
0		0	0	1	0
1		0	1	1	1

(Karnaugh map with cells numbered 0-7 and groups circled: a vertical group of 1s in column 11 (cells 3, 7) and a horizontal group of 1s in row 1 (cells 5, 6, 7).)

Leolvasott DNF alakok:

$$Y = D = x_1x_2 + x_1y + x_2y \Rightarrow S^3_{2,3}(x_1, x_2, y)$$

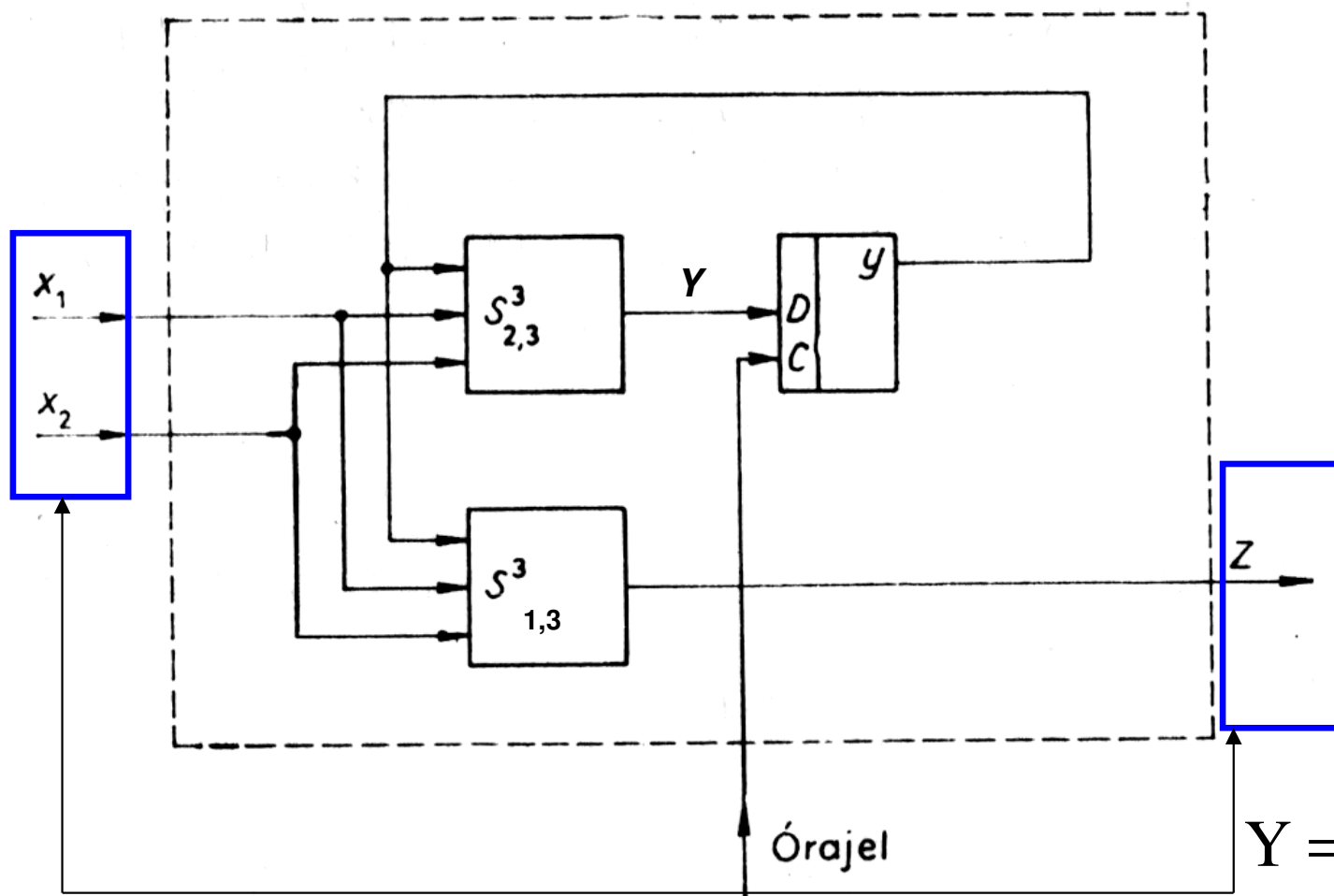
$$Z = x_1 \oplus x_2 \oplus y \Rightarrow S^3_{1,3}(x_1, x_2, y)$$

//Z-t előző alapján kaptuk

*****Megj:** D értéke minden egyes rovatban azonos a kódolt állapotábra ugyanazon rovatában szereplő Y értékkel (tehát $Y=D$).

D-FF a szinkron S.H-k visszacsatoló ágainak működésével azonos.

6.d.) Szinkron soros összeadó ← D FF felhasználásával felépített elvi logikai rajza



*Z kimenet közvetlenül függ a bemenetektől és a visszacsatolt állapottól is. Tehát **Mealy modell** adódott a Karnaugh egyszerűsítések és az ábra alapján is.*

$$Y = D = S_{2,3}^3(x_1, x_2, y)$$

$$Z = x_1 \oplus x_2 \oplus y \Rightarrow S_{1,3}^3(x_1, x_2, y)$$

*Visszacsatolt S.H.-val
realizálható!*

7. Működés szemléltetése idődiagramon

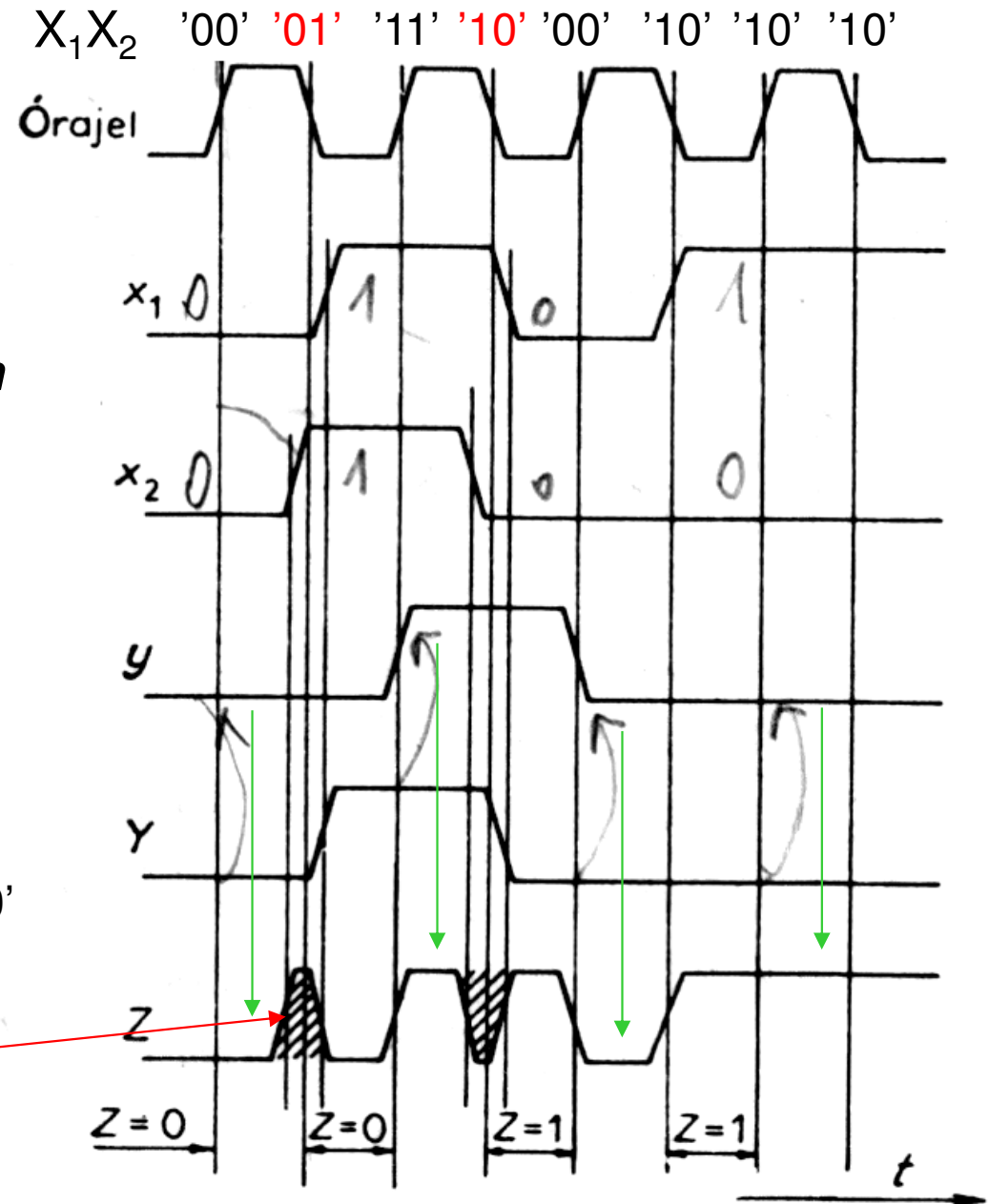
Kimeneti értékek értelmezési tartománya (szinkronizációs feltételként kell kijelölni). Valóságban véges meredekségűek.

- $X_1X_2 = '00' \rightarrow '11' \rightarrow '00' \rightarrow '10'$ komb.
- X bemeneti változás: lefutó élre +/- dt ↘
- y állapot változás ($y \leq Y$): felfutó élre ↗

„Csak szomszédos” bemeneti változásokat feltételezve ('01' ill. '10' közbeiktatásával):

$X_1X_2 = '00' \rightarrow '01' \rightarrow '11' \rightarrow '10' \rightarrow '00' \rightarrow '10'$
 //Tfh. X_2 előbb változzon dt-vel előbb X_1 -nél

Ne alakuljon ki bizonytalan átmeneti állapot (satírozott) ezért a Z kimenetek értékét csak órajel-impulzus „szünetekben” értelmezzük.





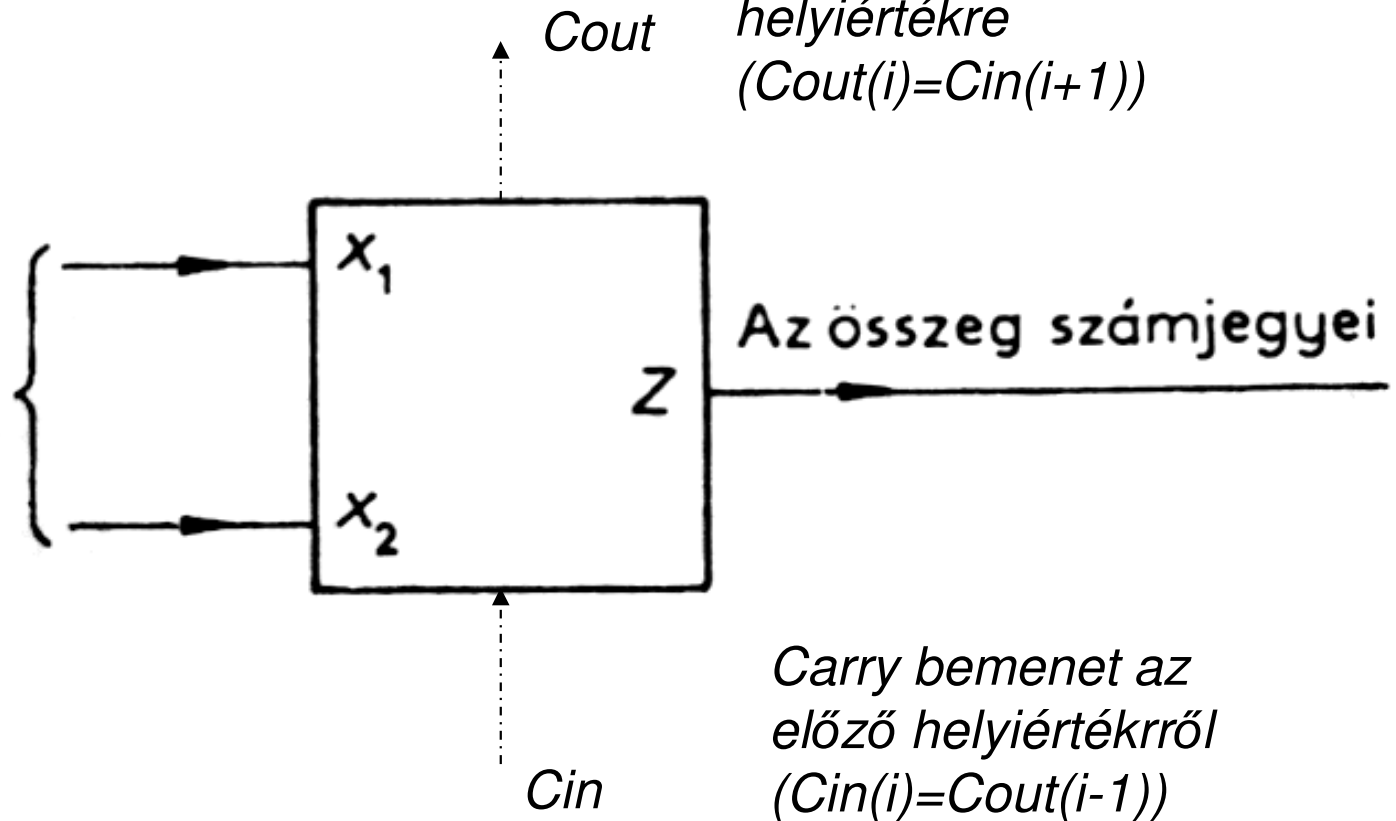
Tervezés: Moore modell

1.) Logikai feladat megfogalmazása

- Tervezzünk egy **soros összeadót Moore modell alapján**, mint szinkron S.H.-ot, melynek 2 bemenete (X_1 , X_2) és egy kimenete van (Z). Bemenetekre, illetve a kimenetre órajel ütemezéssel érkezzenek az adatok.
- Soros összeadó: **MSB** ← **LSB** felé haladva sorban érkeznek az összeadandó adatok (mint bináris számjegyek).
- Átvitelt (Carry) meg kell oldani az egyes helyiértékek között.

Soros összeadó

*A két összeadandó
bináris szám
összetartozó
helyiértékei*



$$Z = X_1 + X_2 + (C_{in})$$

2.) Előzetes állapottábla összeállítása (Moore)

- A S.H.-nak különbözőképpen kell reagálnia ugyanarra a bemeneti kombinációra a megelőző bemeneti kombinációktól és a hálózat állapotától függően.
- DE: az új bemeneti kombináció hatása csak azután fog érvényesülni a kimeneten ($X \rightarrow Z$), miután az általa létrehozott Y visszahat y -ként a bemenetre ($X \rightarrow Y \rightarrow y = Z$)! (Moore szinkronizálás)
- Kérdés: min. mennyi állapot kell?
 - Előzetes állapottáblában általában **több állapot** szerepel a szükségesnél.
 - Kimenet **előző/korábbi** értéke (Z)
 - Előző helyiértéken keletkezett átvitel (carry) értéke ($C_{in} \leftarrow C_{out}$)

Lényeges eltérés Mealy-Moore modellek működése között

- Mealy-Moore állapottábla közötti fő eltérés, hogy
 - a *Mealy* esetén mindig az **új** kimeneti értéket kell bejegyezni a **következő állapot** mellé, míg
 - *Moore* esetén a **kiindulási** kimeneti értéket kell bejegyezni a **következő állapot** mellé!

Állapotoknak megfelelő szekunder kombinációk megadása (nem változott):

Állapot jele (y)	Az előző helyiértéken keletkezett átvitel értéke (C _{in} =C _{out})	A kimenet pillanatnyi értéke !
a	0	0
b	0	1
c	1	0
d	1	1

↑
Szekunder kombinációk

Megj: soros összegzéskor a számított kimeneti érték független a pillanatnyi kimenet értéktől, csak a bemeneti kombinációktól, illetve az előző helyiértéken keletkezett átviteltől függ.

Előzetes állapottábla összeállítása

- Állapottábla: 4 sora (y), 4 oszlopa van (X_1X_2)

$\begin{matrix} X_1X_2 \\ y \end{matrix}$	00	01	11	10
a	$\begin{matrix} yz \\ a0 \end{matrix}$	$b0$	$c0$	$b0$
b	$a1$	$b1$	$c1$	$b1$
c	$b0$	$c0$	$d0$	$c0$
d	$b1$	$\epsilon 1$	$d1$	$c1$

↓
Moore modell:
soronként azonos
kimeneti Z értékek

Kitöltés: (1. sor) *Kiindulási állapot bárhol lehet!*

•Tfh. $y=a$ és $X_1X_2 = '00'$ → $X_1+X_2+Cin = '0'$, és $Cout = '0'$ → Ekkor $Y='a'$ állapot és $Z='0'$ kimenetre: **a0**

•Tfh. $y=a$ és $X_1X_2 = '01'$ v $'10'$ → $X_1+X_2+Cin = '1'$ és $Cout = '0'$ → Ekkor $Y='b'$ állapot, de a **korábbi!** $Z='0'$ kimenetre: **b0**

•Tfh. $y=a$ és $X_1X_2 = '11'$ → $Z=X_1+X_2+Cin = '10'$ és $Cout = '1'$ → Ekkor $Y='c'$ állapot, de a $Z='0'$ kimenetre: **c0**

...

Kitöltés: (4. sor)

•Tfh. $y=d$ és $X_1X_2 = '00'$ → $X_1+X_2+Cin = '1'$, és $Cout = '0'$ → Ekkor 'b' állapot, de a $Z='1'$ kimenetre: **b1**

•Tfh. $y=d$ és $X_1X_2 = '01'$ v $'10'$ → $X_1+X_2+Cin = '10'$ és $Cout = '1'$ → Ekkor 'c' állapot, de a **korábbi** $Z='1'$ kimenetre: **c1**

•Tfh. $y=d$ és $X_1X_2 = '11'$ → $Z=X_1+X_2+Cin = '11'$ és $Cout = '1'$ → Ekkor 'd' állapot és $Z='1'$ kimenetre: **d1**

3.) Összevont (egyszerűsített) állapotábra

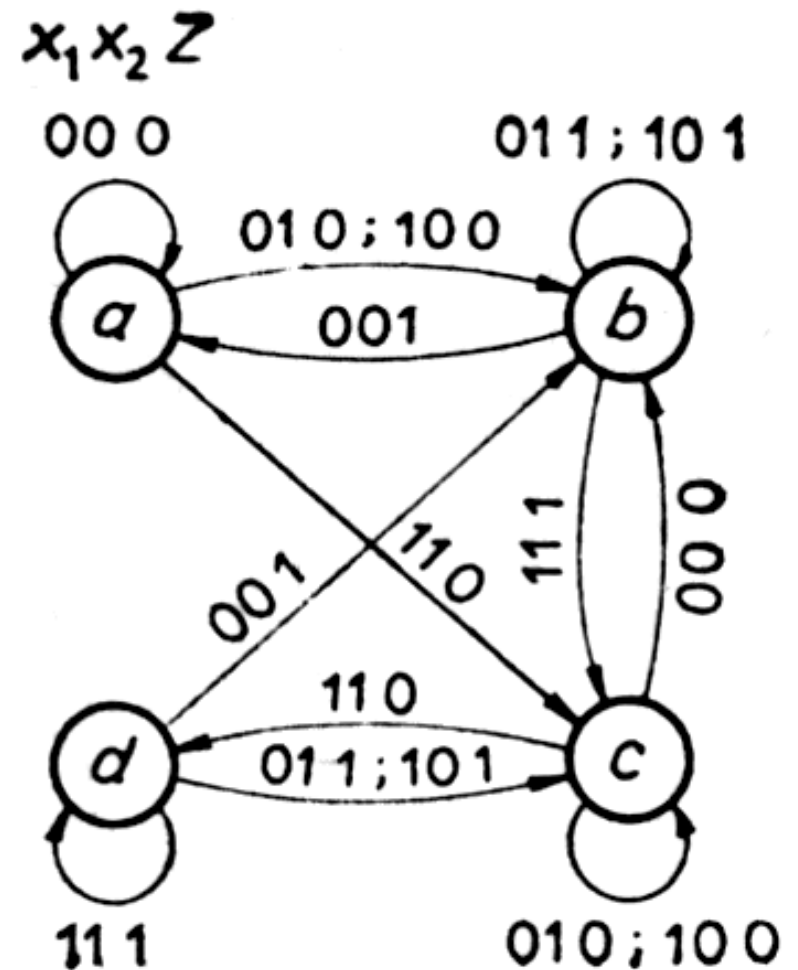
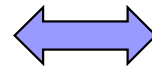
- Következmény: a jelenlegi példánkban az előzetes állapotábrán *nincsenek feleslegesen megkülönböztetett állapotok*.
 - Így előzetes = összevont állapotábra
- Oka(i): Kimeneti (Z) értéket direkt módon a mindenkori állapot határozza meg (y)!
- Az állapotokat (Y) tehát a kimenet értéke szerint is meg kell különböztetnünk, emiatt sem lehetséges további összevonás. Így az előzetes állapotábra, egyben összevont állapotábra is (esetünkben).

Összevont (egyszerűsített) állapottábla és állapotgráf

- Állapotgráf:

Összevont = Előzetes állapottábla:

x_1x_2 $y \setminus$	00	01	11	10
a	$a0$	$b0$	$c0$	$b0$
b	$a1$	$b1$	$c1$	$b1$
c	$b0$	$c0$	$d0$	$c0$
d	$b1$	$c1$	$d1$	$c1$



4.) Állapotkódolás

- Felvett állapot kódok: $n=4$ - állapot \rightarrow 2 szekunder y_1 - y_2 állapot (2-2 lehetséges értékkel). Tfh:

	<u>y_1</u>	<u>y_2</u>
a	0	0
b	0	1
c	1	0
d	1	1

(ez a fenti kódolás csak egy lehetséges módja, lehetett volna másként is definiálni, és **egyáltalán nem közömbös, miként vesszük fel az állapot kódokat!**)₄₀

Állapotkódolás (folyt.)

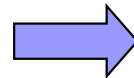
■ Kódolt állapottábla felírása:

**Összevont
állapottábla:**

x_1x_2 $y \setminus$	00	01	11	10
a	$y z$ $a 0$	$b 0$	$c 0$	$b 0$
b	$a 1$	$b 1$	$c 1$	$b 1$
c	$b 0$	$c 0$	$d 0$	$c 0$
d	$b 1$	$c 1$	$d 1$	$c 1$

**Kódolt
állapottábla:** $Y_1 Y_2 Z$

x_1x_2 $y_1y_2 \setminus$	00	01	11	10
00	$Y_1 Y_2 Z$ $00 0$	010	100	010
01	00 1	011	10 1	011
10	010	100	11 0	100
11	011	10 1	11 1	10 1



$Tfh: a:=y_1y_2=00$

$b:=y_1y_2=01$

$c:=y_1y_2=10$

$d:=y_1y_2=11$

4.) Kimeneti függvény Karnaugh táblája

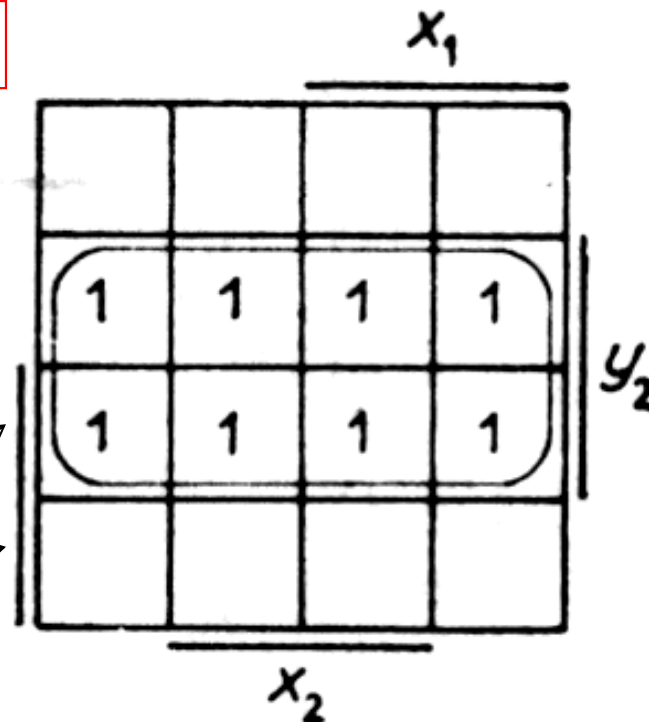
- Kódolt állapotábla felírása (Z) kimenetre:

Kódolt állapotábla:

x_1x_2 y_1y_2	00	01	11	10
00 $y_1y_2 Z$ 00 0	010	100	010	
01	00 1	01 1	10 1	01 1
10	01 0	10 0	11 0	10 0
11	01 1	10 1	11 1	10 1

Soros összegző kódolt állapotáblája

Z



$$Z = f(y_1, y_2) = y_2$$

$$f'_z(y) \Rightarrow Z$$

5.) Vezérlési tábla összeállítása

- Minden egyes szekunder (y_1 - y_2) állapothoz hozzá kell rendelni egy *Flip-flop-ot* (pl. építőelem készletből).
- FF-k működésének ismeretében meghatározzuk, miként kell *vezérelni* azokat egy K.H. segítségével a kódolt állapottábla alapján:
 - (példánkban két FF lesz, mivel 2 szekunder változó volt). Így kapjuk meg a **vezérlési táblá(k)at!**
 - Előfordulhat, ha több szekunder változónk van, hogy eltérő FF-kat rendelünk egy-egy változó szekunder állapot tárolásához!
 - Paraméter tábla használata!

5.) Vezérlési tábla összeállítása – J-K tároló segítségével (folyt.)

- Vezérlési tábla felírása (szekunder komb.):

Kódolt állapottábla:

x_1x_2 y_1y_2	00	01	11	10
00	y_1y_2Z 00 0	010	100	010
01	00 1	011	10 1	011
10	010	100	11 0	100
11	011	101	11 1	10 1

$y_1Y_1='00'$
 $y_2Y_2='00'$

Vezérlési tábla:

x_1x_2 y_1y_2	00	01	11	10
00	J_1K_1 J_2K_2 0 - 0 -	0 - 1 -	1 - 0 -	0 - 1 -
01	0 - -1	0 - -0	1 - -1	0 - -0
10	-1 1 -	-0 0 -	-0 1 -	-0 0 -
11	-1 -0	-0 -1	-0 -0	-0 -1

Leolvasás a 3.31 „paraméter tábla” J-K sora alapján adott yY kombinációkra

5.) Vezérlési tábla alapján (J-K) Karnaugh táblák felírása

Vezérlési tábla:

$x_1 x_2$ $y_1 y_2$	00	01	11	10
00	$J_1 K_1$ 0 -	$J_2 K_2$ 0 -	0 - 1 -	1 - 0 -
01	0 - - 1	0 - - 0	1 - - 1	0 - - 0
10	- 1 1 -	- 0 0 -	- 0 1 -	- 0 0 -
11	- 1 - 0	- 0 - 1	- 0 - 0	- 0 - 1



J_1

		x_1	
		0	1
y_1	0		1
	1		1
y_2	0	-	-
	1	-	-
		x_2	

K_1

		x_1	
		0	1
y_1	0	-	-
	1	1	-
y_2	0	-	-
	1	1	-
		x_2	

J_2

		x_1	
		0	1
y_1	0	-	1
	1	-	-
y_2	0	-	-
	1	1	1
		x_2	

K_2

		x_1	
		0	1
y_1	0	-	-
	1	1	1
y_2	0	-	-
	1	1	1
		x_2	

Leolvasott DNF alakok:

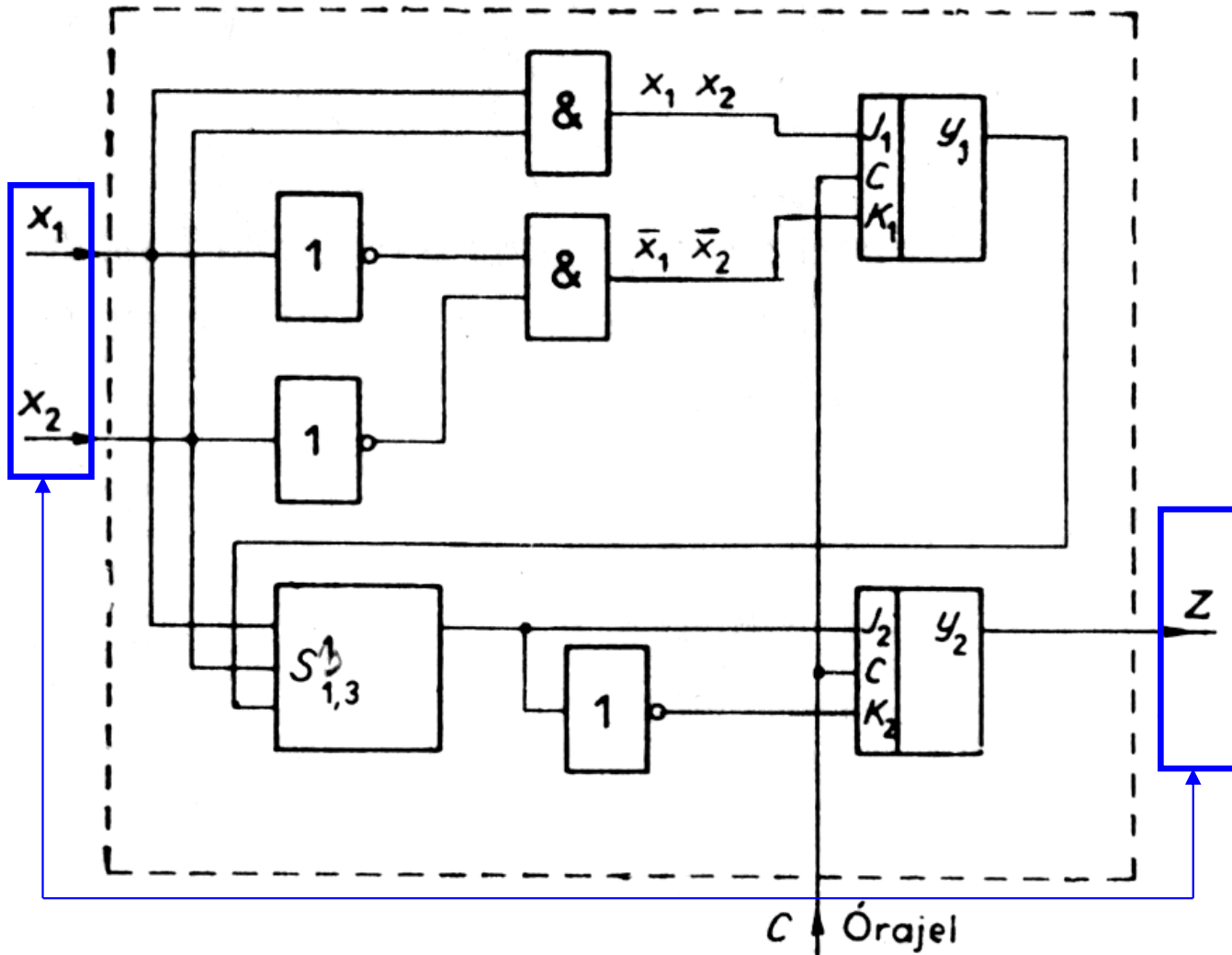
$$J_1 = x_1 \cdot x_2 \quad K_1 = \overline{x_1} \cdot \overline{x_2} \quad Z = y_2$$

$$J_2 = \overline{x_1} \overline{x_2} y_1 + \overline{x_1} x_2 y_1 + x_1 \overline{x_2} y_1 + x_1 x_2 y_1$$

$$\Rightarrow S^3_{1,3}(x_1, x_2, y_1)$$

$$K_2 = J_2 = S^3_{1,3}(x_1, x_2, y_1) = S^3_{0,2}(x_1, x_2, y_1)$$

6.) Szinkron soros összeadó ← J-K FF felhasználásával felépített elvi logikai rajza



Z kimenet közvetve függ a bemenetektől, és közvetlenül az állapottól is. Tehát **Moore modell** adódott a Karnaugh egyszerűsítések és az ábra alapján is.

$$J_1 = x_1 \cdot x_2$$

$$K_1 = \overline{x_1} \cdot \overline{x_2}$$

$$J_2 = S_{1,3}^3(x_1, x_2, y_1)$$

$$K_2 = \overline{J_2} = S_{0,2}^3(x_1, x_2, y_1)$$

$$Z = y_2$$

7. Működés szemléltetése idődiagramon

Kimeneti értékek értelmezési tartománya (szinkronizációs feltételként kell kijelölni). Valóságban véges meredekségűek.

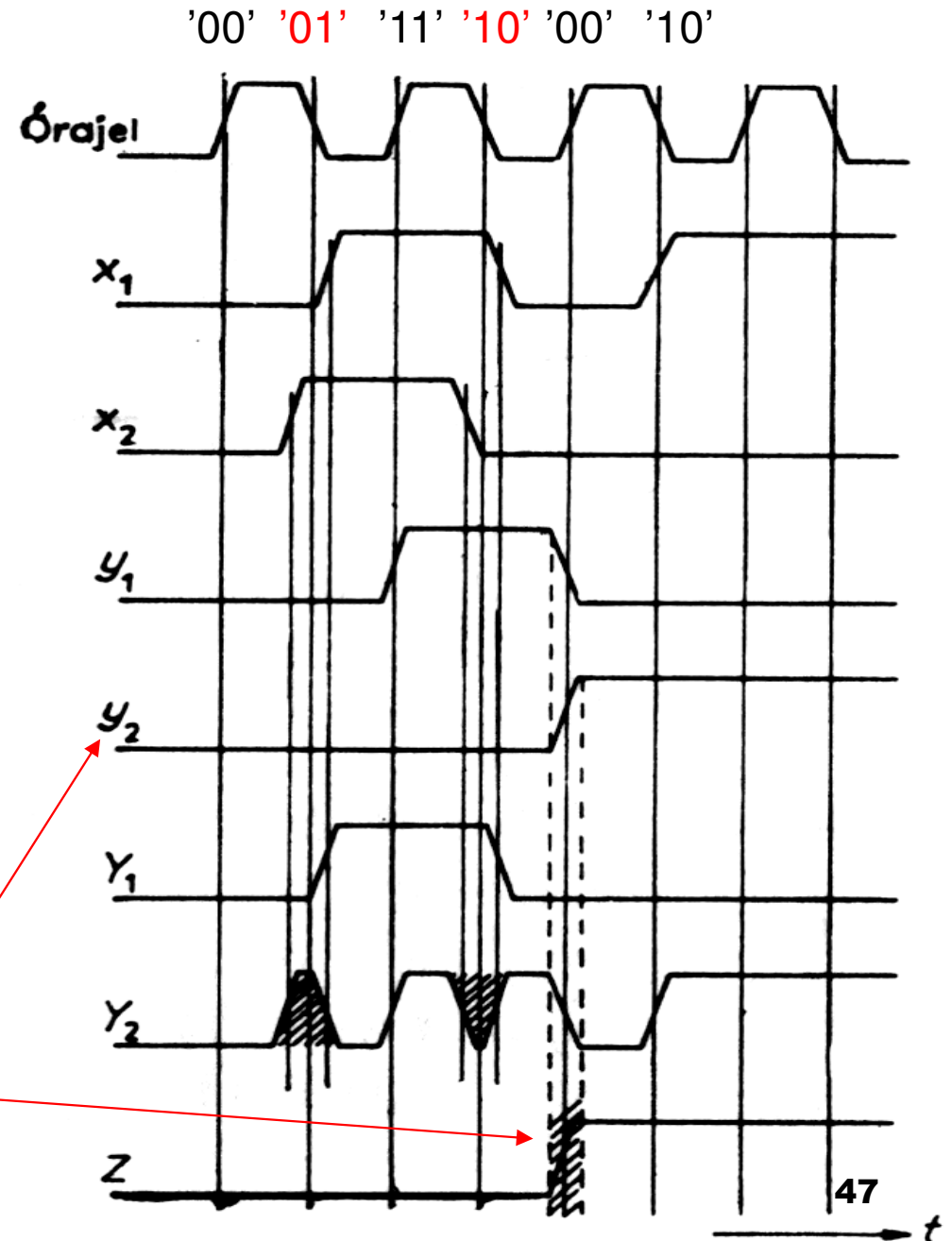
• $X_1X_2 = '00' \rightarrow '11' \rightarrow '00' \rightarrow '10'$ komb.

• **X** bemeneti változás: lefutó élre +/- dt ↘

• **y** állapot változás: felfutó élre ↗

„Csak szomszédos” bemeneti változásokat feltételezve ('01' ill. '10' közbeiktatásával):

$X_1X_2 = '00' \rightarrow '01' \rightarrow '11' \rightarrow '10' \rightarrow '00' \rightarrow '10'$



Ne alakuljon ki bizonytalan átmeneti állapot (satírozott) -> y_2 biztosítja a kimenet érzéketlenségét a két felfutó él közötti kombinációkra

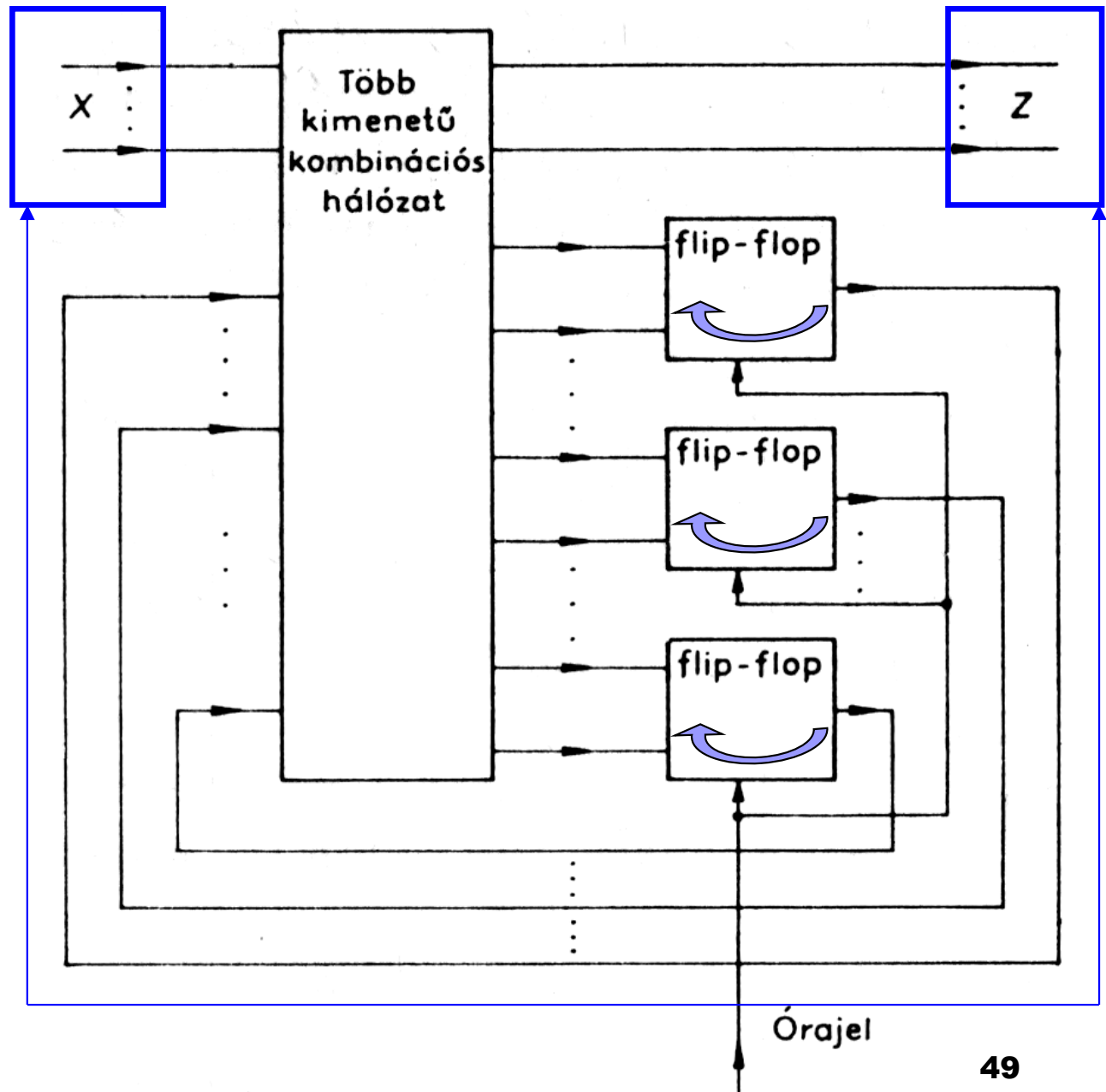
$$Z = y_2$$


Összefoglalás: Szinkron sorrendi hálózat főbb tervezési lépései

- 1. Logikai feladat megfogalmazása
 - 2. Előzetes állapottábla összeállítása
 - 3. Egyszerűsített (összevont) állapottábla
 - 4. Állapotkódolás megválasztása
 - 5. Vezérlési tábla összeállítása
 - 6. Vezérlési K.H. és kimenetet előállító K.H. realizációja
 - 7. Működés szemléltetése idődiagramon
- Mindezt *Mealy* ill. *Moore* modelleken is megvizsgálva és megvalósítva.

Szinkron S.H. általános vázlat szerinti felépítése

- Ábra: (Sorrendi működést az egyes FF-ok állapotainak visszacsatolásai valósítják meg.)
- Valóságban azonban a sorrendi működést az egyes FF-okon **belüli** visszacsatolások biztosítják!!





II. **Aszinkron** sorrendi hálózatok tervezési lépéseinek bemutatása

Aszinkron S.H. tervezési lépései:

- **1. Feladat megfogalmazása**
- **2. Az előzetes állapotábra felírása**
 - (aszinkron viselkedés, szomszédos bemeneti változások engedélyezése, minden sorban csak egy stabil állapot lehet)
- **3. Összevont állapotábra**
- **4. Alkalmazandó FF-típus megválasztása** (vagy visszacsatolt K.H-al valósítjuk meg!)
- **5. Állapotkódolás:** elsődleges cél a kritikus versenyhelyzet kiküszöbölése (ez nem feltétlenül a legegyszerűbb hálózat megtervezését jelenti)
- **6. Vezérlési tábla felírása és elvi logikai rajz**
- **7. „Lényeges hazard” ellenőrzése** (ha szükséges a megfelelő szekunder változók késleltetése)

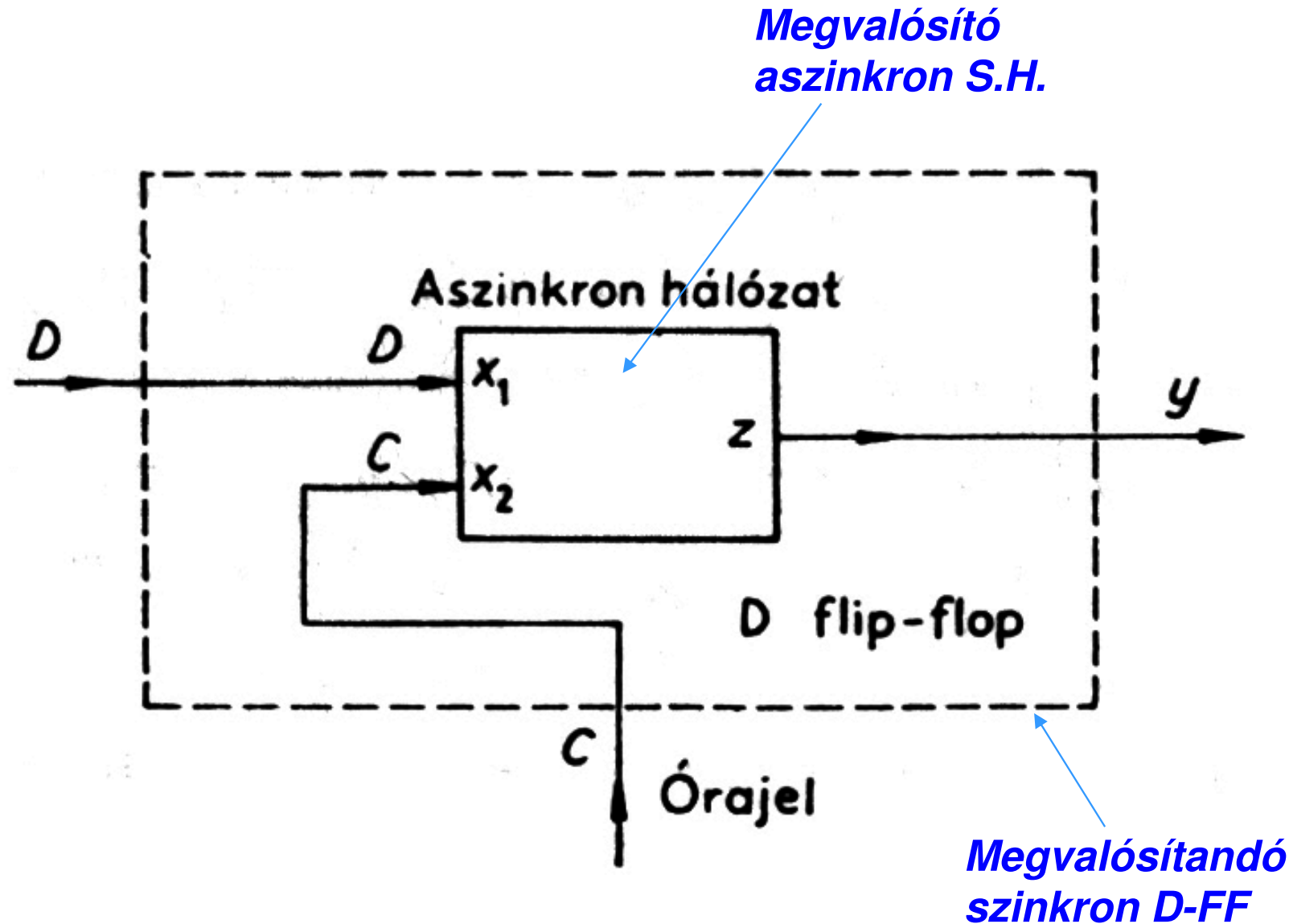
Aszinkron S.H.:

- **Feltételezés**: egymás után következő bemeneti kombinációk esetén ***csak szomszédos*** bemeneti változások megengedettek.
 - E nélkül ugyanis, az **aszinkron** S.H. működése nem lenne egyértelműen követhető az állapottáblán, mert *funkcionális hazard* jönne létre,
 - Pl. $x_1x_2 = 00 \rightarrow 11$ helyett $00 \rightarrow 01 / 10 \rightarrow 11$, attól függően, hogy x_1 vagy x_2 változását érzékeli előbb (akár egy *másik stabil, de nem az előírt stabil állapotba is kerülhet* a sorrendi hálózatunk)

1.) Feladat megfogalmazása

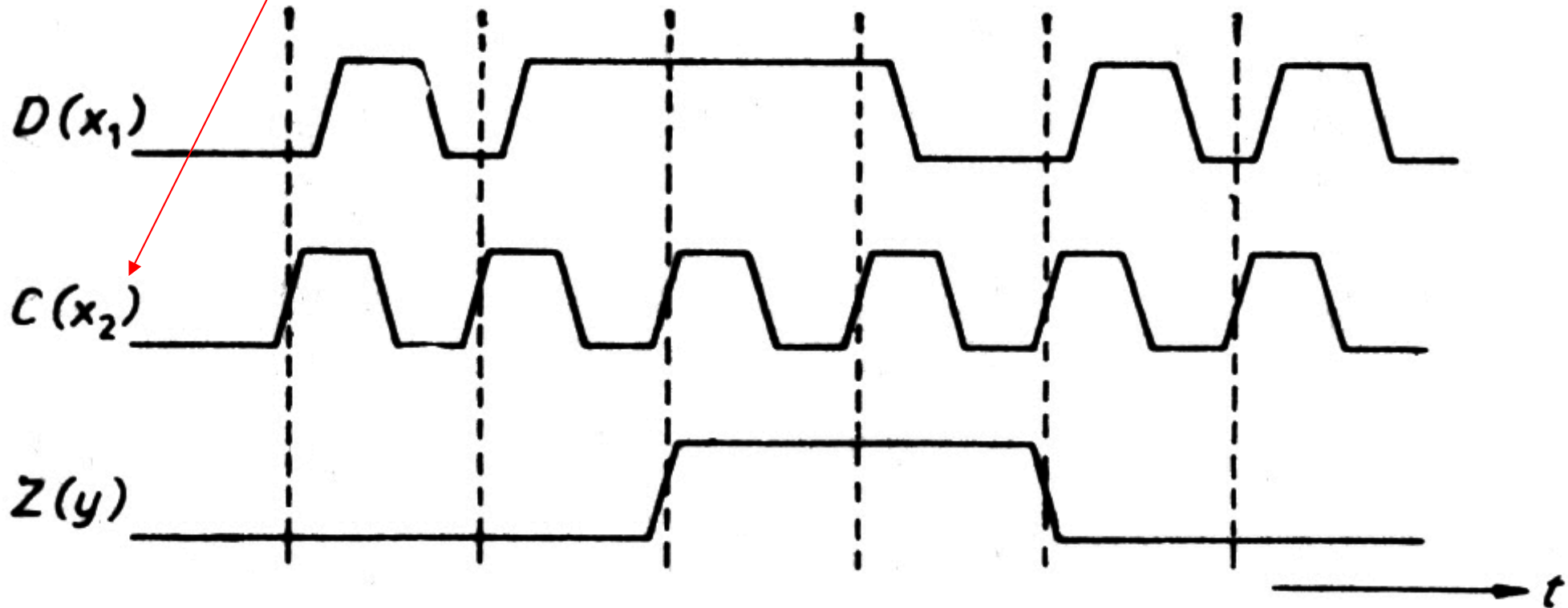
- **Megvalósító aszinkron** S.H. hálózat tervezése, amely „kívülről” **szinkron tárolóként** viselkedik.
 - **Megvalósítandó szinkron** tároló legyen: D-FF
 - „Szinkronizációs feltételek” biztosítása
 - CLK (C) órajelet tekintsük független bemeneti jelnek
- **Működés:** Tfh. Z kimenet csak akkor változhat, amikor $C(x_2)$ értéke '0' → '1' (felfutó él ↗), és ekkor Z vegye fel a D bemenetnek a pillanatnyi értékét, $Z=D(x_1)$.
 - Megj: a ()-ben megadott kifejezések a belső, felhasznált aszinkron hálózat jeleit definiálják.

Szinkron D tároló ← aszinkron S.H. hálózattal történő megvalósítása: „rendszer technikai rajz”



Aszinkron hálózattal megvalósítandó szinkron D-FF működésének szemléltetése

C(x₂) CLK bemenetet nem, mint szinkron órajelet, hanem mint független (aszinkron) bemeneti jelet értelmezzük!



A nem-szomszédos $D(x_1)$ - $C(x_2)$ bemeneti változásokat kizártnak tekintjük!

2.) Előzetes állapotábra felírása

$\begin{matrix} DC \\ x_1, x_2 \\ y \end{matrix}$	00	01	11	10
a	a0	b0	-	c0
b	a0	b0	d0	-
c	a0	-	e-	c0
d	-	b0	d0	c0
e	-	g1	e1	f1
f	h1	-	e1	f1
g	h1	g1	e1	-
h	h1	b-	-	f1

Lehetőség szerint minden sorban pontosan **egy stabil állapot** (kör), a vele nem szomszédos változást **don't care-nek** '–' rögzítjük. Ez jelentősen növelheti az esetleges állapot-összevonást!

Ism: csak stabil állapotból lehet a bemeneti kombinációknak változnia („vízszintesen” haladva), különben csak y-mentén haladhatunk („függőlegesen”).

Kitöltés:

1. Tfh. $D(X_1); C(X_2) = '00'$ stabil **kiindulási állapot**, ahol: $y:=a$ és $Z:=0$. Ekkor: a0. (stabil $y \leq Y:=a$)

2. Tfh: $D(X_1); C(X_2) = '01'$ -re vált. (Ekkor $D=0$, $C=0 \rightarrow 1$, tehát a kimenet pill. értéke $Z=0$ lesz, felveszi D-t). Legyen b instabil állapot. Ekkor: b0. ($y \neq Y=b$ instabil)

2.) Előzetes állapotábra felírása (folyt.)

$\begin{matrix} DC \\ x_1, x_2 \\ y \end{matrix}$	00	01	11	10
a	(a)0	b0	-	c0
b	a0	(b)0	d0	-
c	a0	-	e-	(c)0
d	-	b0	(d)0	c0
e	-	g1	(e)1	f1
f	h1	-	e1	(f)1
g	h1	(g)1	e1	-
h	(h)1	b-	-	f1

- Tfh: a0 stabil állapotban vagyunk és $D(X1);C(X2) = '10'$ -ra vált. (Ekkor $D=0 \rightarrow 1$, $C=0$, tehát a kimenet pill. értéke $Z=0$ marad változatlanul). c instabil állapot. Ekkor: c0. ($y \neq Y=c$ instabil). //3. sor c0 stabil állapotába viheti át//
- Ha a 2. lépésben kialakuló b0 mint $Y \Rightarrow y$ visszahat a bemenetre, akkor a 2. sorba jutunk, ahol b0 **stabil** állapot lesz ($D(X1);C(X2) = '01'$ és $y=b$)
- Ez azt jelenti, hogy $D(X1);C(X2) = '10'$ közömbösnek rögzítendő „-” (funkc. hazard mentes).
- Tfh: b0 stabil állapotban vagyunk és $D(X1);C(X2) = '11'$ -re vált. (Ekkor $D=0 \rightarrow 1$, $C=1$, tehát a kimenet pill. értéke $Z=0$ marad változatlanul). d instabil állapot. Ekkor: d0. ($y \neq Y=d$ instabil).
- Tfh: b0 stabil állapotban vagyunk és $D(X1);C(X2) = '00'$ -ra vált. (Ekkor $D=0$, $C=1 \rightarrow 0$, tehát a kimenet pill. értéke $Z=0$ marad változatlanul). a instabil állapot. Ekkor: a0. ($y \neq Y=a$ instabil).
- Az a0 instabil állapotból (2. sor) az a0 stabil állapotba kerülhetünk vissza (1. sorba)

2.) Előzetes állapotábra felírása (folyt.)

$\begin{matrix} DC \\ x_1, x_2 \\ y \end{matrix}$	00	01	11	10
a	(a)0	b0	-	c0
b	a0	(b)0	d0	-
c	a0	-	e-	(c)0
d	-	b0	(d)0	c0
e	-	g1	(e)1	f1
f	h1	-	e1	(f)1
g	h1	(g)1	e1	-
h	(h)1	b-	-	f1

9. Tfh: a 3.sorban c0 stabil állapotban vagyunk azaz $y=c$, és $D(X1);C(X2) = '11'$ -ra vált. (Ekkor $D=1, C=0 \rightarrow 1$, tehát a kimenet pill. értéke $Z=1$ kell legyen). Az '11' oszlopon belül olyan stabil állapotba kell jutni!, amelyhez $Z='1'$ kimeneti érték tartozik. Ezt jelöljük e-vel. **Kimenetként** azért kell írjunk **közömbös** '-' értéket, mivel a c0 stabil állapotból e- köztes instabil állapoton keresztül e1 stabil állapotba kerülhet a rendszer, miközben a kimeneti érték megváltozik (c0 \rightarrow e1!). Ilyen esetekben, amikor a stabil – (instabil állapothoz tartozó kimeneti érték párok eltérők, közömbös kimeneti értéket kell rögzíteni a köztes instabil állapothoz!

10. c0 stabil állapot azt jelenti, hogy $D(X1);C(X2) = '01'$ közömbösnek „-” (funkc. hazard mentes) kell venni.
11. Tfh: c0 stabil állapotban vagyunk és $D(X1);C(X2) = '00'$ -re vált. (Ekkor $D=1 \rightarrow 0, C=0$, tehát a kimenet pill. értéke $Z=0$ marad változatlanul). a instabil állapot. Ekkor: a0. ($y \neq Y=c$ instabil).
12. Amelyből az a0 stabil állapotba kerülhetünk vissza (1. sorba)

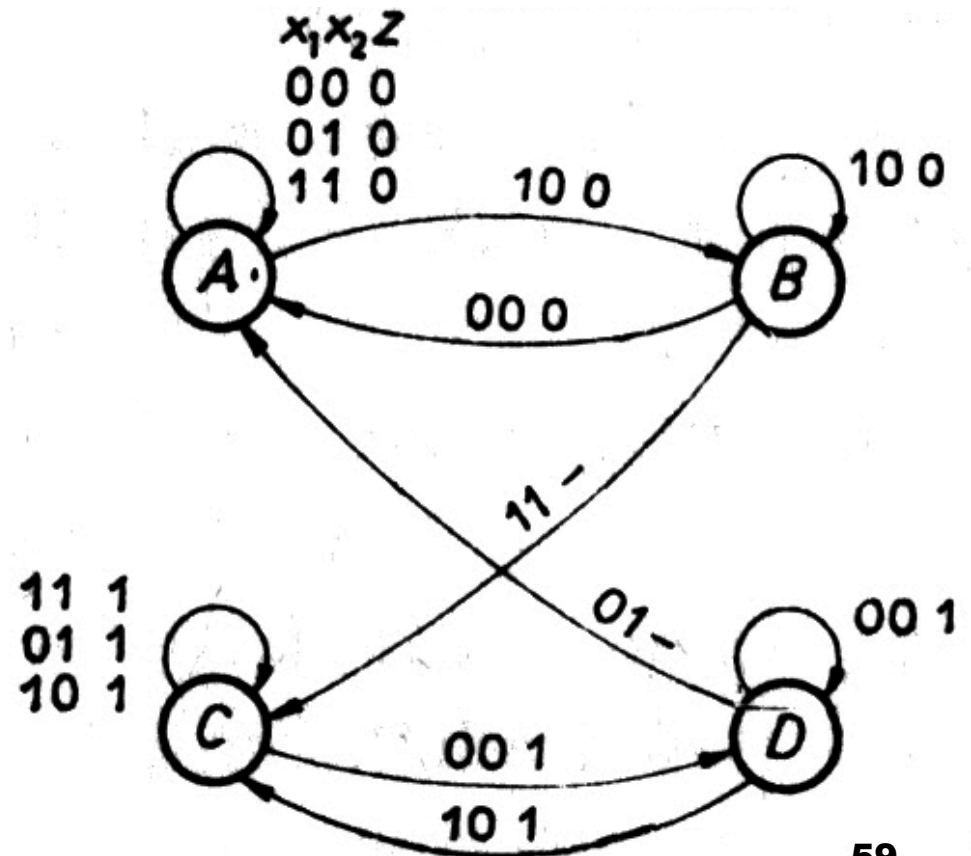
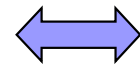
És így tovább... ☺

3.) Összevont (egyszerűsített) állapot-tábla és állapot-gráf:

■ Állapot összevonások (egy lehetséges módja!):

- A:= a,b,d (1.-,2.-, és 4. sor)
- B:= c (3. sor)
- C:= e,f,g (5.-,6.- és 7. sor)
- D:= h (8. sor)

x_1x_2 y	00	01	11	10
A	(A)0	(A)0	(A)0	B0
B	A0	-	C-	(B)0
C	D1	(C)1	(C)1	(C)1
D	(D)1	A-	-	C1



4. Alkalmazandó FF-típus megválasztása

- Alkalmazandó, megvalósítandó FF típus:
 - Adott: szinkron D-(C) megvalósító tároló
 - Feladat megfogalmazásának lépésében már meghatároztuk

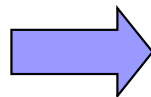
5. Állapotkódolás (I. mód):

- Összevont állapottábla 4 különböző állapotának megkülönböztetése miatt legalább 2 szekunder változó definiálása szükséges (legyen: y_1 ill. y_2)
- **Állapotkódolás** (több lehetséges mód is van!):
amely meghatározza a hálózat helyes, vagy hibás működését is!

Kódolt állapottábla:

Tfh:

	y_1	y_2
A	0	0
B	0	1
C	1	0
D	1	1



$x_1 x_2$	00	01	11	10
00	$y_1 y_2 z$ 00 0	00 0	00 0	01 0
01	00 0	—	10 —	01 0
10	11 1	10 1	10 1	10 1
11	11 1	00 —	—	10 1

Állapotkódolás hatása egy S.H. ‘helyes’ vagy ‘hibás’ működésére:

„**Kritikus versenyhelyzet**” (rendszer hazárd)

- DEF: Az aszinkron S.H. ‘helyes’ vagy ‘hibás’ működése (a hibás stabil állapotba jutás is) a y szekunder változók egymáshoz képesti működési sebességétől függhet. Ezt a sebességarányt a hálózatban fellépő jelterjedési késleltetések befolyásolják, ezért a hálózat viselkedése véletlenszerű lehet.
- **Kritikus versenyhelyzet (vagy Rendszer Hazárd)**: két szekunder állapot (pl. y_1 és y_2) közötti verseny (sebesség arányaik eltérőek)
- Tehát az **állapotkódolás** megválasztása nemcsak a megvalósítandó S.H. egyszerűségét, hanem annak helyes/hibás működését is döntően befolyásolja.

Példa: Kritikus versenyhelyzetre (I. mód)

Kódolt állapottábla vizsgálata:

x_1x_2	00	01	11	10
00	00 0	00 0	00 0	01 0
01	00 0	—	10 —	01 0
10	11 1	10 1	10 1	10 1
11	11 1	00 —	—	10 1

Egyidejű változás: Tfh: $X_1X_2 = '10'$ mellett az $y_1y_2 = '01'$ stabil állapotban van a rendszer (2.sor)

1.) Változzon a bemenet $X_1X_2 = '11'$ -re. Ekkor a hálózatnak az '11' oszlopában $y_1y_2 = '10'$ stabil állapot van előírva (3. sor), melyet a $y_1y_2 Y_1Y_2 = '01'10'$ instabil állapoton keresztül ér el (azaz $Y = '10'$ visszahat y-ra). Feltétel ekkor az, hogy a bemeneti változás hatására a szekunder változóknak $y_1 = '0' \rightarrow '1'$ míg $y_2 = '1' \rightarrow '0'$ **kell egyszerre változnia!**

Ha mégsem egyszerre változnak:

2.) Tfh. y_1 **gyorsabban** változik, mint y_2 . Először $y_1 = '1' \rightarrow '0'$. Ekkor átmenetileg létrejön az $y_1y_2 = '00'$ szek. kombináció. Innen viszont $Y ('00') \rightarrow y$ visszahatására az '11' oszlop '00' cellájába (1. sor) kerülünk, amely szintén stabil, de 'hibás' stabil állapot (az előírt $y_1y_2 = '10'$ helyett!)

Nem kritikus versenyhelyzet

- A **szekunder változók közötti sebességarány** nem minden esetben hoz létre *hibás* stabil állapotot.
 - Előfordulhat, hogy a szekunder változók egymáshoz képesti működési sebessége *csak azt befolyásolja, hogy a következő helyes stabil állapotot milyen más lehetséges instabil állapotokon keresztül értük el.* Ebben az esetben a **versenyhelyzet nem kritikus (nem vezet rendszer házárdhoz).**

5. Állapotkódolás (II. mód):

- Összevont állapottábla 4 állapotának megkülönböztetése miatt legalább 2 szekunder változó definiálása szükséges (legyen y_1 ill. y_2)
- ***Új állapotkódolás (több lehetséges mód közül!): meghatározza a hálózat helyes vagy hibás működését is!**

Kódolt állapottábla:

Tfh*:

	y_1	y_2
A	0	0
B	0	1
C	1	1
D	1	0



***Állapot kódolás az I. módhoz képest**

	DC		$x_1 x_2$			
	$y_1 y_2$		00	01	11	10
A	00	$y_1 y_2 z$ 00 0	00 0	00 0	01 0	01 0
B	01	00 0	—	11 —	01 0	—
C	11	10 1	11 1	11 1	11 1	11 1
D	10	10 1	00 —	—	11 1	—

Példa: Kritikus versenyhelyzetre (II. mód)

Kódolt állapottábla vizsgálata:

$\begin{matrix} DC \\ x_1 x_2 \\ y_1 y_2 \end{matrix}$	00	01	11	10
00	$\begin{matrix} y_1 y_2 z \\ \textcircled{00} 0 \end{matrix}$	$\textcircled{00} 0$	$\textcircled{00} 0$	01 0
01	00 0	—	11 —	$\textcircled{01} 0$
11	10 1	$\textcircled{11} 1$	$\textcircled{11} 1$	$\textcircled{11} 1$
10	$\textcircled{10} 1$	00 —	—	11 1

Előző módszerhez (I.mód) képest itt a szekunder változók eltérő sebességarányának változása ellenére **sem alakul ki kritikus versenyhelyzet**, mivel mindig *csak egyetlen szekunder változó* változtatja az értékét ennél a II. mód. állapotkódolásnál!

1.) Y1 vált: $\textcircled{01} \rightarrow '11' \rightarrow \textcircled{11}$

6. Vezérlési tábla összeállítása –

- **Aszinkron S.H.:** elvégezhetjük a megvalósítást
 - a.) akár *tisztán visszacsatolt* K.H. segítségével,
 - b.) vagy *adott megvalósító FF felhasználásával* is, (lásd későbbi S-R, vagy D-G aszinkron FF-kal)
- „Tisztán visszacsatolt” hálózat esetén a ***vezérlési tábla azonos*** az állapottáblával!
 - (mivel Y_1 –et ill. Y_2 -t előállító függvényeket kell megvalósítani, majd visszacsatolni)
 - Kódolt állapottábláról kell leolvasni az Y_1 –re , Y_2 -re illetve Z-re vonatkozó Karnaugh táblákat.

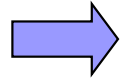
6. a.) Vezérlési tábla → Karnaugh táblák

tisztán visszacsatolt K.H.-al

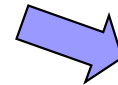
Vezérlési tábla (=kódolt állapotábra II. mód):

DC x ₁ x ₂ y ₁ y ₂	00	01	11	10
00	$\textcircled{00} \text{ } 0$	$\textcircled{00} \text{ } 0$	$\textcircled{00} \text{ } 0$	01 0
01	00 0	—	11 —	$\textcircled{01} \text{ } 0$
11	10 1	$\textcircled{11} \text{ } 1$	$\textcircled{11} \text{ } 1$	$\textcircled{11} \text{ } 1$
10	$\textcircled{10} \text{ } 1$	00 —	—	11 1

Karnaugh táblái



Paraméter tábla D sora, VAGY pusztán önvisszacsatolással (aszinkron!)



Z

	x ₁		
	—	—	
y ₁	1	1	1
	1	—	1
	x ₂		

Y₁

	x ₁		
	—	1	
y ₁	1	1	1
	1	—	1
	x ₂		

Y₂

	x ₁		
			1
	—	1	1
y ₁	1	1	1
		—	1
	x ₂		

Leolvasott DNF alakok:

Közös primimplikánsok

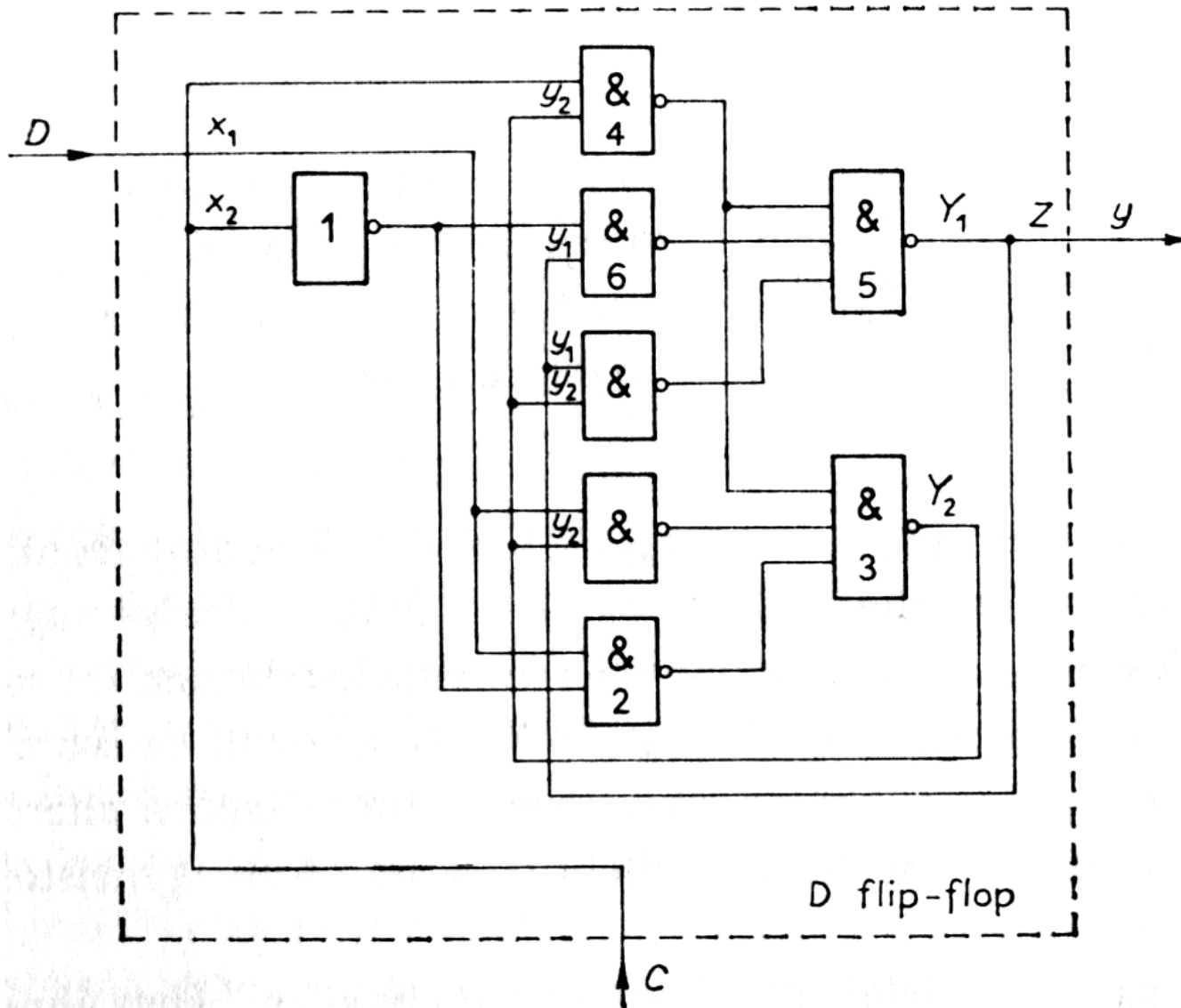
$$Z = y_1$$

$$Y_1 = \textcircled{x_2 y_2} + \overline{x_2} y_1 + \textcircled{y_1 y_2}$$

$$Y_2 = \textcircled{x_2 y_2} + \overline{x_1} \overline{x_2} + \textcircled{x_1 y_2}$$

Hazárd-mentesítés!

6.a.) D-FF-ot megvalósító aszinkron S.H. tisztán visszacsatolt K.H.-al felépített elvi NAND kapus logikai rajza



$y=Z$ kimenet csak közvetve függ a bemenetektől! Így tehát a **Moore modell** adódott a Karnaugh egyszerűsítések és ábra alapján.

$$f'_z(y) \Rightarrow Z$$

$$Z = y_1$$

Aszinkron: Mealy vs. Moore modell szerinti működése

- Ha az aszinkron S.H. *előzetes állapottáblájának* minden sorába csak egyetlen stabil állapotot definiáltunk, még nem döntöttünk annak *Mealy, vagy Moore modell-szerű* működéséről. Ez csak az **összevont állapottábla** alapján dönthető el!
- Csak akkor tételezhetünk fel **Mealy** modellt, ha összevonunk olyan állapotokat is, amelyekhez tartozó bemeneti kombinációtól **függően** eltérő kimeneti értékek tartoznak, illetve az egy sorban lévő közömbös kimeneti értékeket egymástól eltérő módon rögzítjük (választjuk meg) azaz **soronként nem azonos kimeneti értékek**.

6. Vezérlési tábla összeállítása – b.) megvalósító FF felhasználásával

- Aszinkron: elvégezhetjük a megadását adott megvalósító *FF felhasználásával* is (vagy *tisztán visszacsatolt* K.H. segítségével, lásd **a.**) esetben)
- Ekkor a **vezérlési tábla leolvasható az** állapottábláról a „paraméter tábla” segítségével!
- Majd a vezérlési tábláról kell leolvasni az Y_1 -re , Y_2 -re illetve Z-re vonatkozó Karnaugh táblákat.

b.) D-FF-ot megvalósító aszinkron S.H. vezérlési táblája (aszinkron!) **S-R** FF felhasználásakor

Kódolt állapottábla:

DC $y_1 y_2$ \ $x_1 x_2$	00	01	11	10
00	$y_1 y_2$ z 00 0	00 0	00 0	01 0
01	00 0	—	11 —	01 0
11	10 1	11 1	11 1	11 1
10	10 1	00 —	—	11 1

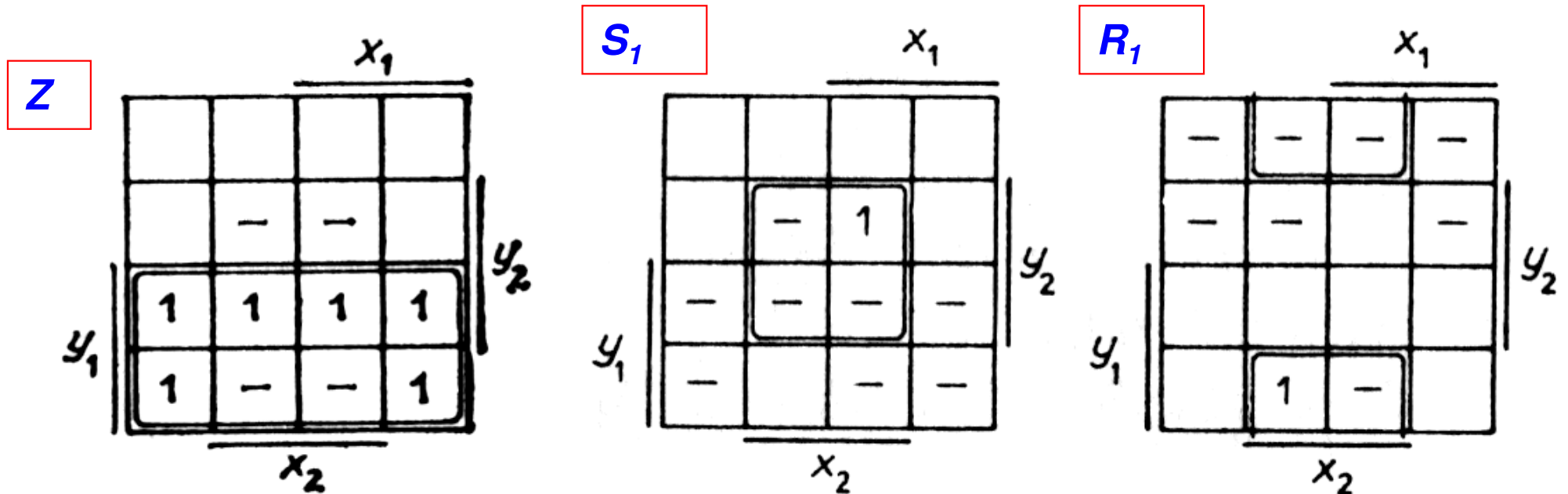
Vezérlési tábla:

S-R

DC $y_1 y_2$ \ $x_1 x_2$	00	01	11	10
00	$S_1 R_1$ $S_2 R_2$ 0 - 0 -	0 - 0 -	0 - 0 -	0 - 10
01	0 - 01	-- --	10 -0	0 - -0
11	-0 01	-0 -0	-0 -0	-0 -0
10	-0 0 -	01 0 -	-- --	-0 10

Leolvasás a 3.31 „paraméter tábla” **S-R** sora alapján adott yY kombinációkra

6. b.) Vezérlési tábla → Karnaugh táblák



Leolvasott DNF alakok:

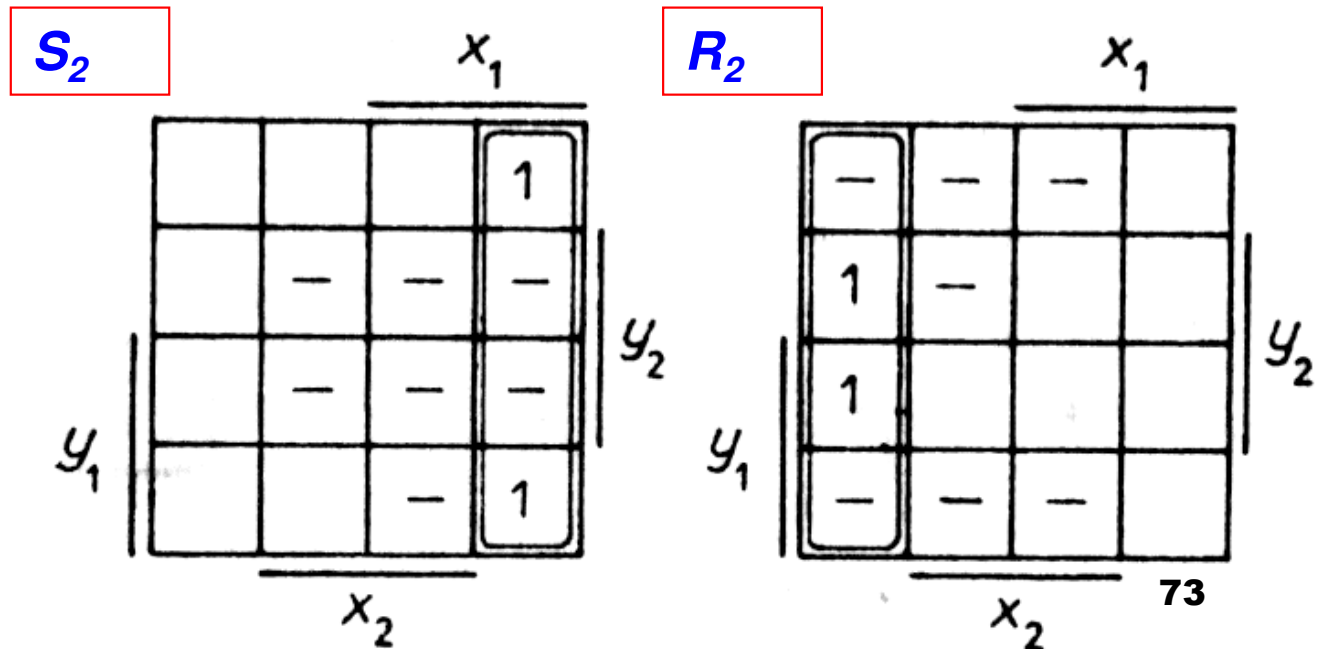
$$Z = y_1 \quad // \text{változatlan}$$

$$S_1 = x_2 y_2 = C y_2$$

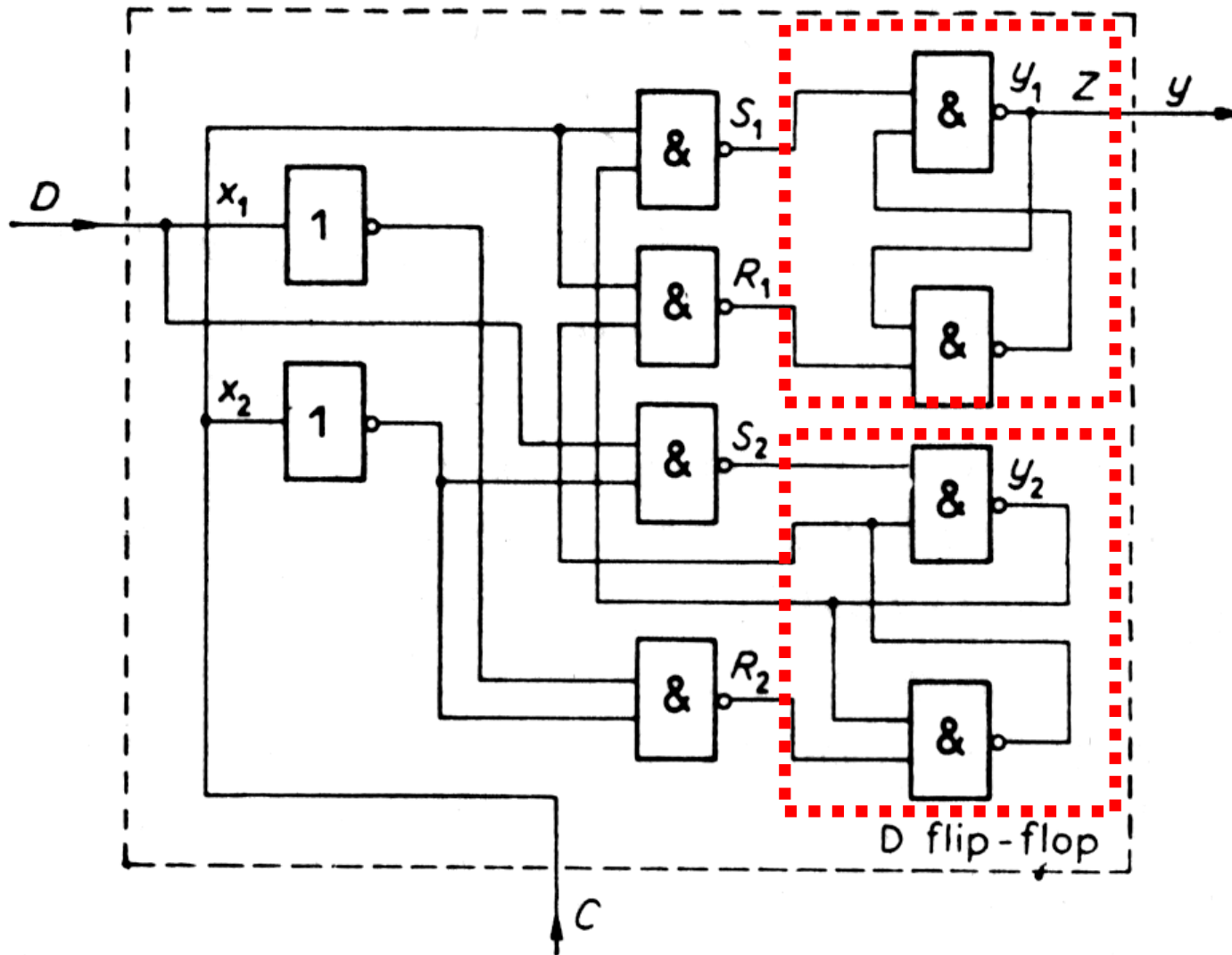
$$R_1 = x_2 \bar{y}_2 = C \bar{y}_2$$

$$S_2 = x_1 \bar{x}_2 = D \bar{C}$$

$$R_2 = \bar{x}_1 \cdot \bar{x}_2 = \bar{D} \cdot \bar{C}$$



6.b.) D-FF-ot megvalósító aszinkron S.H. ← S-R FF felhasználásával felépített elvi NAND kapus logikai rajza



$y=Z$ kimenet független
a bemenetektől (csak
közvetve függ). Tehát
Moore modell adódott
a Karnaugh
egyszerűsítések és
ábra alapján.

$$f'_z(y) \Rightarrow Z$$

$$Z = y_1$$

7. Lényeges hazárd ellenőrzése

- Eddigiekben megtervezett (1-6. lépés) aszinkron S.H. még mindig nem biztos, hogy biztonságosan (hazárdmentesen) működik.

$y_1 y_2$ \ $x_1 x_2$ DC	00	01	11	10
00	$y_1 y_2 z$ 00 0	00 0	00 0	01 0
01	00 0	—	11 —	01 0
11	10 1	11 1	11 1	11 1
10	10 1	00 —	—	11 1

Handwritten annotations: A blue arrow labeled '1' points from the '11' column to the '10' column in the '00' row. A red arrow labeled '2.' points from the '10' column to the '11' column in the '01' row.

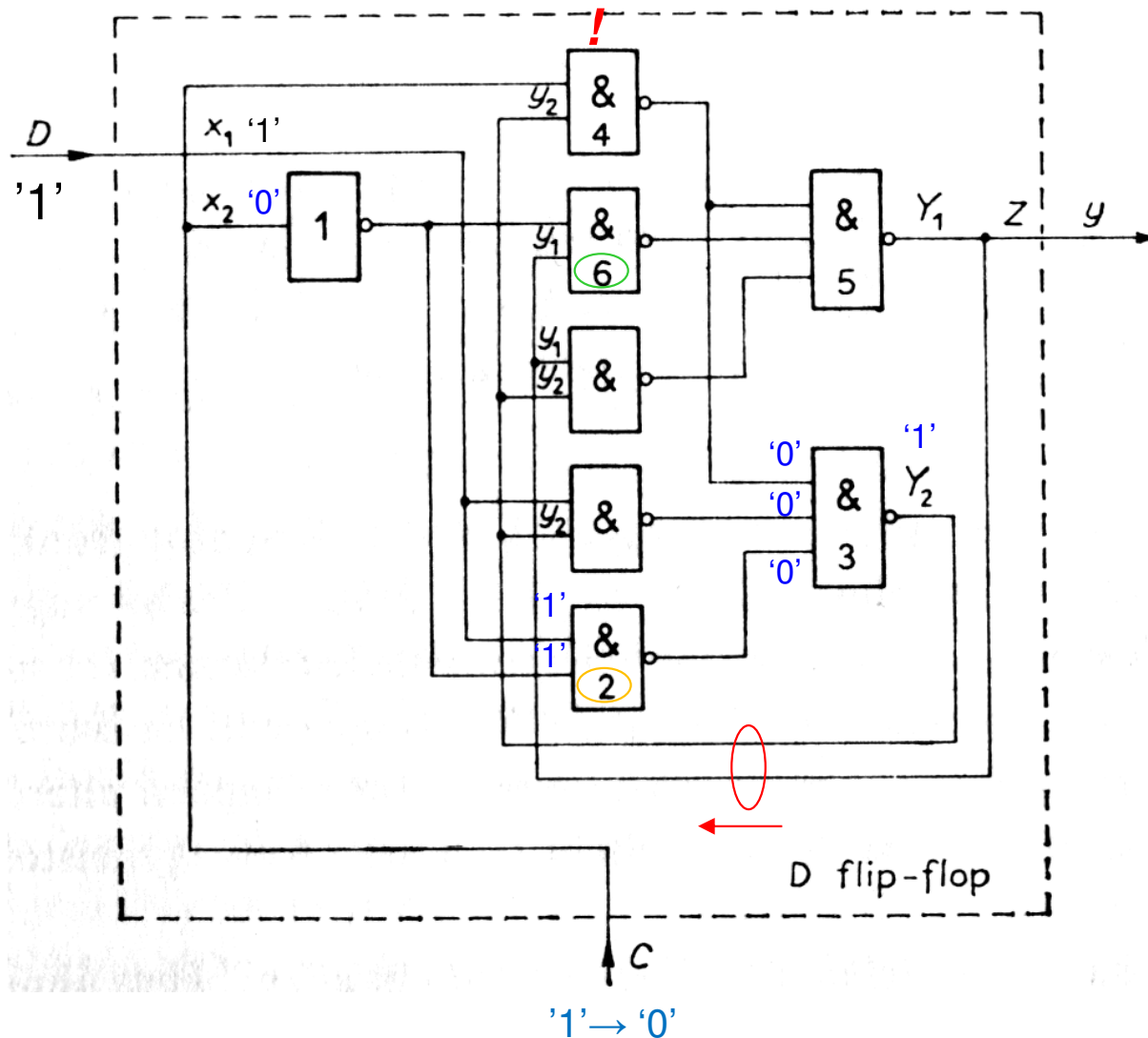
Tfh: $X_1 X_2 = '11'$ mellett az $y_1 y_2 = '00'$ **stabil** állapotban van a rendszer (1.sor)

1.) Változzon $X_1 X_2 = '10'$ –ra. Az előírás szerint ennek az oszlopnak a $y_1 y_2 = '01'$ **stabil** állapotába kerül majd a rendszer.

2.) De bizonyos **késleltetési viszonyok** miatt, akár ugyanezen $X_1 X_2 = '10'$ oszlop $y_1 y_2 = '11'$ **stabil** állapotába is kerülhetünk (hiba, nem ez volt előírva!). →

7. Lényeges hazárd ellenőrzése

(6.a. ábra alapján)



Tfh: $X_1 (D) = '1'$ és $X_2 (C)$ bemenet $'1' \rightarrow '0'$ előírt változását először az inverteren keresztül a 2-jelű NAND kapu észleli (de a 6-os jelű még nem).

Majd ennek hatása eljut a 3-as NAND kapura is, és $Y_2 = '1'$ jön létre, ami visszahatva $y_2 = '1'$ -et hoz létre pl. 4-es jelű NAND kapu bemenetén.

Tfh. pl. a 4-es jelű NAND kapu a másik bemenetén (nagy Δt késleltetéssel) csak ezután észleli a $X_2 (C)$ bemenet $'1' \rightarrow '0'$ korábbi változását, akkor az 5-ös jelű NAND-n keresztül $Y_1 = '1'$ jön létre. Ez visszahatva $y_1 = '1'$ -et hoz létre pl. a 6-os jelű NAND kapun.

Mivel az $Y_2 = '1'$ -et az $X_1 X_2 = '10'$ kombinációra a 2-jelű NAND kapun keresztül állandóan fenntartja, kialakul egy $y_1 y_2 Y_1 Y_2 = '11 11'$ stabil, de hibás állapot (nem volt előírva)

Hibás állapotátmenet

- A hibás állapotátmenet (hibás stabil állapot) létrejöttének oka, hogy van(nak) az aszinkron S.H.-nak olyan kapubemenete(i), amelyeken a szekunder (y) állapotok értékváltozása **előbb** érvényesül (aktualizálódik), mint a bemeneti (X) változók értékváltozása. (pl. 4-es NAND).
 - Emiatt olyan instabil állapot jön létre ($Y_1Y_2='11'$) amely hibás (nem előírt) stabil állapotba vezeti az S.H.-t.

DEF: Lényeges hazard

- Alakulhat ki, ha az *aszinkron S.H.* az összevont állapottáblától eltérően / vagy annak megfelelően 'hibás' / 'helyes' módon működik. Lényeges hazard attól függ, hogy a **bemeneti** (X) és a **szekunder** (y) változások egymáshoz képest milyen időrendben játszódtak le (**sebességaránya**). Ha a sorrendi hálózat ettől az eltérő sebességaránytól függően is hibás stabil állapotba kerülhet – akár tetszőleges állapotkódolást választva – **lényeges hazard** alakul ki.

Lényeges hazard kiküszöbölése

- Az állapotkódolás változtatásával *sem* lehetséges kiküszöbölni!
- Lehet viszont a **szekunder változók** megfelelő **késleltetésével**:
 - Szekunder változók ($y \leq Y$) visszahatását késleltetjük, ahhoz hogy a **bemeneti változás elsődleges** legyen. Pl: a visszacsatoló ág(ak)ba páros számú INVERTER használatával.

Összefoglaló táblázat: Jelterjedési késleltetések hatása (hiba) az *aszinkron sorrendi hálózatokban*

Jelenség oka	Jelenség elnevezése	Okozott hiba	Kiküszöbölés módja
A bemeneti változók (pl. X1,X2) közötti versenyhelyzet. (Nem szomszédos bemeneti kombináció változásokat tekintve)	Funkcionális hazard	Előre <i>nem meghatározható</i> tranzien jelértékek lehetősége miatt a hálózat működése nem definiálható egyértelműen	<u>Szomszédos bemeneti kombináció</u> változások biztosítása, vagy <u>szinkronizáció</u> (feltételezve hogy szomszédos változás esetén statikus v. dinamikus hazard ne alakuljon ki)
A szekunder változók (pl. y1, y2) közötti versenyhelyzet	Kritikus versenyhelyzet	A szekunder változók <i>változási sebesség-arányától</i> függően hibás stabil állapot kialakulásának lehetősége	Az <u>állapotkódolás</u> megfelelő megválasztásával
A bemeneti változók (pl. X1,X2) és a szekunder változók (pl. y1,y2) közötti versenyhelyzet.	Lényeges hazard	A bemeneti és a szekunder változók <i>változási sebesség-arányától</i> függően hibás stabil állapot kialakulásának lehetősége	A megfelelő visszacsatoló ágakba megfelelő <u>késleltető elemek</u> (pl. páros-számú inverter) beépítése

Aszinkron S.H tervezési lépéseinek összefoglalása:

- **1. Feladat megfogalmazása**
- **2. Az előzetes állapottábla felírása**
 - (aszinkron viselkedés, szomszédos bemenetek engedélyezése, minden sorban csak egy stabil állapot lehet)
- **3. Összevont / egyszerűsített állapottábla**
- **4. Alkalmazandó FF-típus megválasztása** (vagy visszacsatolt K.H-al valósítjuk meg!)
- **5. Állapotkódolás:** elsődleges cél a *kritikus versenyhelyzet* kiküszöbölése (és nem a legegyszerűbb hálózat megtervezése)
- **6. Vezérlési tábla felírása és elvi logikai rajz**
- **7. Lényeges hazard ellenőrzése** (ha szükséges a megfelelő szekunder változók késleltetése)