



M Ű E G Y E T E M 1 7 8 2

Budapesti Műszaki és Gazdaságtudományi Egyetem

Villamosmérnöki és Informatikai Kar

Irányítástechnika és Informatika Tanszék

Dr. Pilászy György

Digitális technika 1

14. előadás

(Számlálók)

Lektorálta: Dr. Horváth Tamás

Minden jog fenntartva. Jelen könyvet, illetve annak részleteit a szerzők írásbeli engedélye nélkül tilos reprodukálni, adatrögzítő rendszerben tárolni, bármilyen formában vagy eszközzel elektronikus vagy más módon közölni.

Számlálók

A sorrendi hálózatok világában igen gyakori feladat külső impulzusok számlálása, frekvenciák leosztása, különböző időzítések előállítása. Ezekre a feladatokra számlálót használunk. A sorrendi hálózatokról tanultak alapján magunk is tervezhetünk ilyen egységeket. A következőkben egy egyszerű példán keresztül mutatjuk meg a számláló áramkör felépítését.

Tervezzünk egy olyan Moore-modell szerint működő szinkron sorrendi hálózatot, amely egy bemenettel (E) és három kimenettel (Z_2, Z_1, Z_0) rendelkezik. $E=0$ esetén a hálózat kimenete nem változik. $E=1$ esetén három bites bináris felfele számlálást valósít meg oly módon, hogy a legnagyobb érték elérése után $Z_2, Z_1, Z_0 = 000$ -ról újraindul. Z_0 kimenet jelenti a legkisebb helyi értéket.

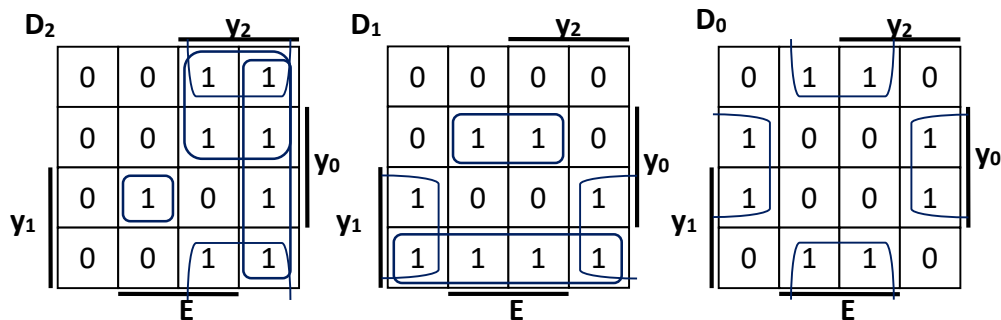
Először készítsük el a hálózat állapotábráját. Mivel Moore modell szerint tervezünk, a kimeneteket minden sor utolsó cellájában tüntessük fel. Így elegendő az egyes cellákban az Y értékeket feltüntetni. A kimeneteket növekvő számsorrendben írtuk fel. Az $E=0$ vezérlés esetén $Y=y$ vezérlés szükséges a „változatlanság” megvalósításához. $E=1$ esetén mindig a következő állapotba vezetjük a hálózatot, míg a legnagyobb kimenet előállítása után visszatérünk a kezdőállapotba (a).

$E \backslash y$	0	1	Z_2, Z_1, Z_0
a	a	b	000
b	b	c	001
c	c	d	010
d	d	e	011
e	e	f	100
f	f	g	101
g	g	h	110
h	h	a	111

Az állapotkódolást valósítsuk meg úgy, hogy a flip-flopok kimenetei közvetlenül szolgáltatassák a kimeneteket ($Z=y$). A kódokat szomszédos sorrendben tüntessük fel, hogy később könnyen lehessen Karnaugh táblát képezni a vezérléshez. A kódolt állapotábra:

$E \backslash y_2 y_1 y_0$	0	1	Z_2, Z_1, Z_0
$y_2 y_1 y_0$	$Y_2 Y_1 Y_0$		
000	000	001	000
001	001	010	001
011	011	100	011
010	010	011	010
100	100	101	100
101	101	110	101
111	111	000	111
110	110	111	110

Valósítsuk meg a hálózatot D-flip-flopok felhasználásával. Mivel a D flip-flopok karakterisztikus egyenlete: $Y=D$, ezért nincs szükség külön vezérlési táblára, mert az megegyezik a kódolt állapotábrával. Határozzuk meg a vezérlési egyenleteket.



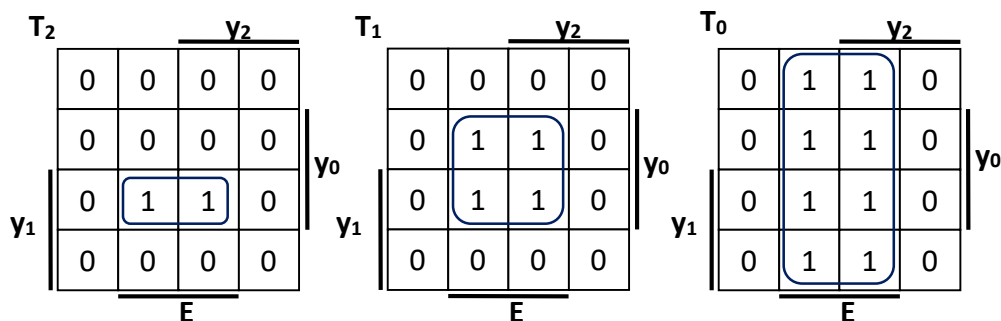
$$D_2 = E \cdot \overline{y_2} \cdot y_1 \cdot y_0 + \overline{E} \cdot y_2 + y_2 \cdot \overline{y_0} + y_2 \cdot \overline{y_1} = y_2 \oplus (E \cdot y_1 \cdot y_0)$$

$$D_1 = E \cdot \overline{y_1} \cdot y_0 + \overline{E} \cdot y_1 + y_1 \cdot \overline{y_0} = y_1 \oplus (E \cdot y_0)$$

$$D_0 = E \cdot \overline{y_0} + \overline{E} \cdot y_0 = y_0 \oplus E$$

A kétszintű diszjunktív algebrai alakokat szándékosan rendeztük át úgy, hogy jól látszódjon benne az XOR kapcsolat. A flip-flopok átalakításának tárgyalásakor láttuk, hogy egy olyan D flip-flop, amelynek a kimenete XOR kapun keresztül van visszacsatolva T flip-flopként működik. Ezen a felismerésen elindulva valósítsuk meg ugyanezt a három bites számlálót T flip-flopok felhasználásával. Első lépésként állítsuk elő a T realizációhoz szükséges vezérlési táblát.

	E	
$y_2 y_1 y_0$	0	1
	$T_2 T_1 T_0$	
000	000	001
001	000	011
011	000	111
010	000	001
100	000	001
101	000	011
111	000	111
110	000	001

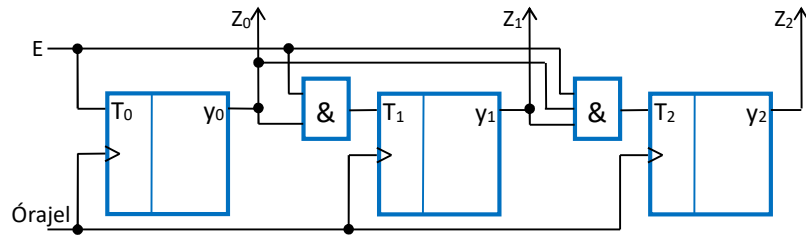


$$T_2 = E \cdot y_1 \cdot y_0$$

$$T_1 = E \cdot y_0$$

$$T_0 = E$$

Figyeljük meg, hogy lényegesen egyszerűbb vezérlési egyenletek adódtak, mint a D flip-flop realizáció esetében. Készítsük el az elvi logikai rajzát a T flip-flopos realizációnak.



Lefelé számlálás megvalósítása

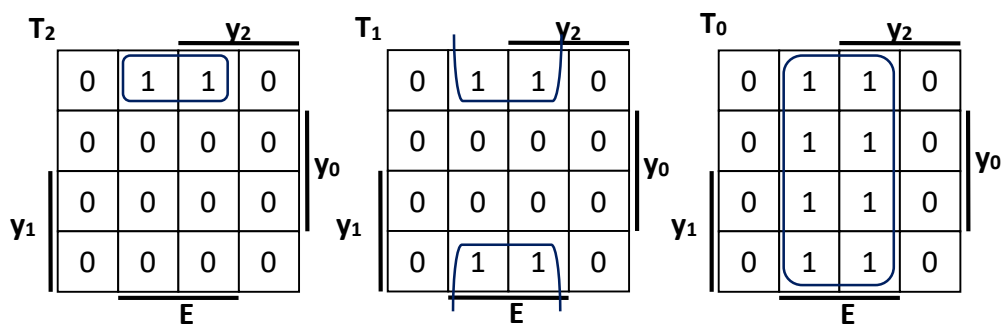
Valósítsuk meg ugyanezt a feladatot, de visszafele számlálással. Ehhez állítsuk elő a számláló hálózat kódolt állapotábráját, majd a T flip-flop alkalmazásához szükséges vezérlési tábláját:

Kódolt állapotábra

$y_2 y_1 y_0 \backslash E$	0	1	Z_2, Z_1, Z_0
	$Y_2 Y_1 Y_0$		
000	000	111	000
001	001	000	001
011	011	010	011
010	010	001	010
100	100	011	100
101	101	100	101
111	111	110	111
110	110	101	110

Vezérlési tábla

$y_2 y_1 y_0 \backslash E$	0	1
	$T_2 T_1 T_0$	
000	000	111
001	000	001
011	000	001
010	000	011
100	000	111
101	000	001
111	000	001
110	000	011

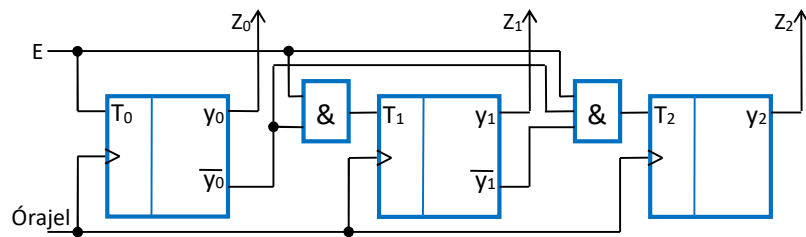


$$T_2 = E \cdot \overline{y_1} \cdot \overline{y_0}$$

$$T_1 = E \cdot \overline{y_0}$$

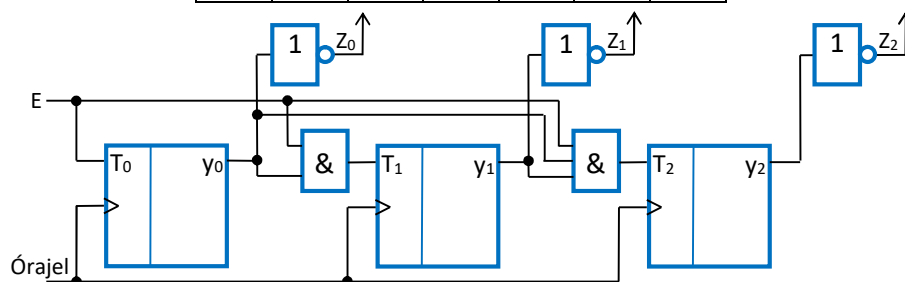
$$T_0 = E$$

Figyeljük meg, hogy a vezérlésekben a negált kimenetek kerültek felhasználásra. Ennek oka, hogy a tervezés során ragaszkodtunk ahhoz, hogy az állapotkód legyen egyben a hálózat kimenete is.



A visszafelé számlálást egy már meglévő „hagyományos” számláló felhasználásával is megvalósíthatjuk, ha a számláló kimeneteit invertáljuk. Az alábbi táblázatban három biten ábrázoltuk a „hagyományos” számláló és a kimenetek invertálásával kapott számláló kimeneti értékeit.

Z_2	Z_1	Z_0	\rightarrow	$\overline{Z_2}$	$\overline{Z_1}$	$\overline{Z_0}$
0	0	0		1	1	1
0	0	1		1	1	0
0	1	0		1	0	1
0	1	1		1	0	0
1	0	0		0	1	1
1	0	1		0	1	0
1	1	0		0	0	1
1	1	1		0	0	0



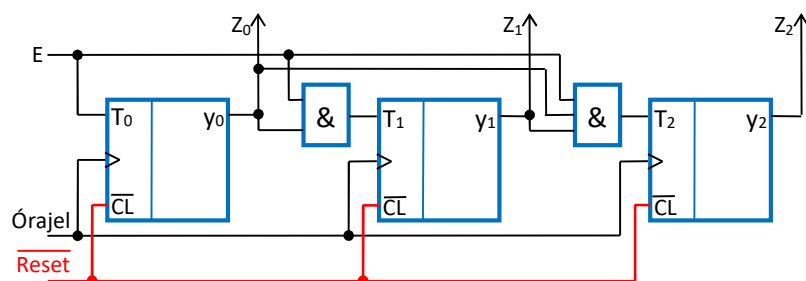
A kimenetek invertálásához a fenti elvi logikai rajzon külön invertereket használtunk, de megjegyezzük, hogy bizonyos esetekben a flip-flopok rendelkezhetnek invertált kimenettel is, ilyenkor azok használata kevesebb külső alkatrészt igényel.

Számlálók alaphelyzetbe állítása

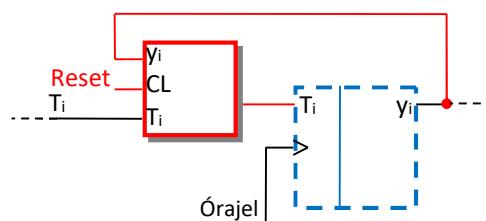
Az alábbi fejezetben összefoglaljuk, hogy milyen kezdőérték beállítási lehetőségeket szokás alkalmazni a számlálóáramkörökben. Számláló felépítéstől függően az alaphelyzetbe állítás lehet törlés (clear, nullázás) vagy tetszőleges kezdeti érték beállítási lehetőség. Mindkettő működhet aszinkron vagy szinkron módon attól függően, hogy a flip-flopok melyik bemenetén keresztül avatkozunk be a működésbe.

Számláló törlése

Aszinkron törlés megvalósításakor a számlálót alkotó flip-flopok törlés (clear) bemeneteit vezetjük ki. Az aszinkron törlés a számláló órajelétől és engedélyezésétől függetlenül azonnal módosítja a számláló értékét. A következő ábra az alacsony aktív működésű aszinkron törlés megvalósítását szemlélteti.



Szinkron törlés kialakításához a flip-flopok „normál” bemenetein keresztül avatkozunk be úgy, hogy az órajel hatására a flip-flop kimenetének új értéke zérus legyen. D-flip-flop esetén elegendő egy ÉS kapuval nullába vezérelni a D-flip-flopok bemenetét. T flip-flop esetén a vezérléshez szükséges a flip-flop kimeneti értékének ismerete is. Tervezzünk egy ilyen kombinációs hálózatot, melyet a számláló eredeti struktúráját megtartva, közvetlenül a T flip-flopok bemenetére téve megvalósítja a számláló törlését. Az áramkör beépítését a következő ábra mutatja.



Szinkron törlés elve

A flip-flop működésének ismeretében meghatározhatjuk a vezérlő hálózat igazságtáblázatát, majd a Karnaugh táblázatát. Ezek alapján szisztematikus tervezéssel elkészíthetjük a szinkron törlést megvalósító vezérlő hálózatot.

CL	y_i	T_i	T	megjegyzés
0	0	0	0	Számolás $T=T_i$
0	0	1	1	
0	1	0	0	
0	1	1	1	
1	0	0	0	Törlés
1	0	1	0	
1	1	0	1	
1	1	1	1	

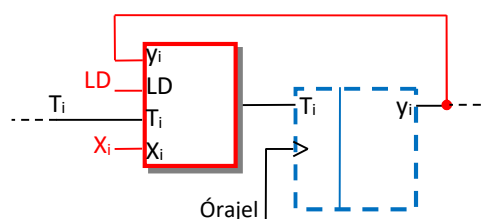
		Y _i		
		0	1	1
CL	T	0	1	1
	0	0	1	1

$$T = \overline{CL} \cdot T_i + CL \cdot y_i$$

Tetszőleges érték betöltése

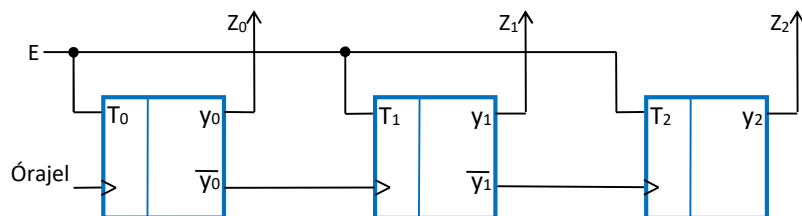
Tetszőleges érték betöltésekor az egyes flip-flopok vezérlését olyan kombinációs hálózattal egészítjük ki, amely biztosítja, hogy egy betöltés engedélyező jel (LD) aktiválásakor egy érték megadó bemenet (X_i) segítségével a flip-flop új kimeneti értéke X_i értékével megegyező legyen.

Szinkron betöltés megvalósításának egy lehetséges elrendezését mutatja az alábbi ábra. A számlálót alkotó összes T flip-flop bemenetének vezérlésébe az alábbi kombinációs hálózatot építjük be a fenti kiegészítő elemeken felül. A kombinációs hálózat négy bemenettel (y_i , LD, T_i , X_i) és egy kimenettel (T) rendelkezik. Ha LD=0, akkor $T=T_i$. Ha LD=1, akkor úgy vezéreljük a T kimenetet, hogy az y_i és X_i bemenet felhasználásával, hogy az órajelet követően $y_i=X_i$ legyen. Másszóval az órajelet hatására x_i bemenetet átmásoljuk a flip-flop kimenetére. Adjuk meg a kombinációs hálózat igazságtáblázatát és Karnaugh táblázatát.

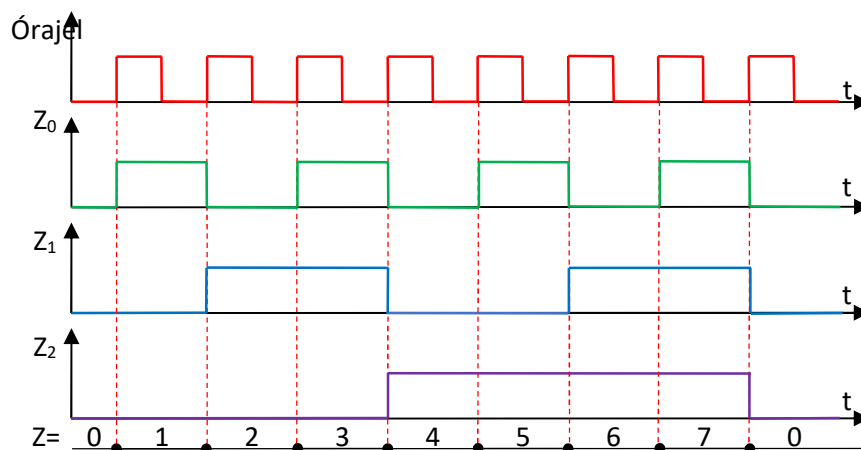


Aszinkron számláló

Számlálási funkciót megvalósíthatunk úgy is, ha a flip-flopok kimenetei szolgáltatják az órajelet a következő fokozat számára. Ilyenkor a lánc elején lévő flip-flop kimenetének megváltozása indítja a következő flip-flopot és így tovább. Kívülről szemlélve úgy látjuk, hogy a kimeneteken végig terjed a változási hullám. Az ilyen számlálók angol elnevezése is ezt sugallja: ripple counter. A következő ábra egy engedélyező bemenettel ellátott felfele számoló aszinkron számláló elvi logikai rajzát mutatja, melyet T-flip-flopokból alakítottunk ki.



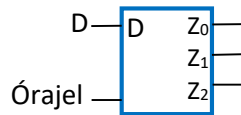
Amennyiben nincs szükség engedélyező bemenetre, úgy a T flip-flopok bemeneteit fix „1” logikai szintre köthetjük. A következő idődiagram szemlélteti a számláló működését felfutó-él vezérelt flip-flopok és $E=1$ vezérlés esetén.



Ha a következő fokozat órajelét az előző fokozat ponált kimenete szolgáltatja, lefelé számláló számlálót kapunk. Ennek ellenőrzését az olvasóra bízuk.

Léptető regiszter

Tervezzünk olyan szinkron Moore modell szerint működő sorrendi hálózatot, amely egy bemenettel (D) és három kimenettel (Z_0, Z_1, Z_2) rendelkezik. A hálózat a D bemeneten megjelenő értéket először a Z_0 kimeneten, majd a Z_1 végül a Z_2 kimenetén jeleníti meg. Másként fogalmazva minden órajel ütemben egy helyi értékkel feljebb lépjen a D bemeneten érkező értékek.



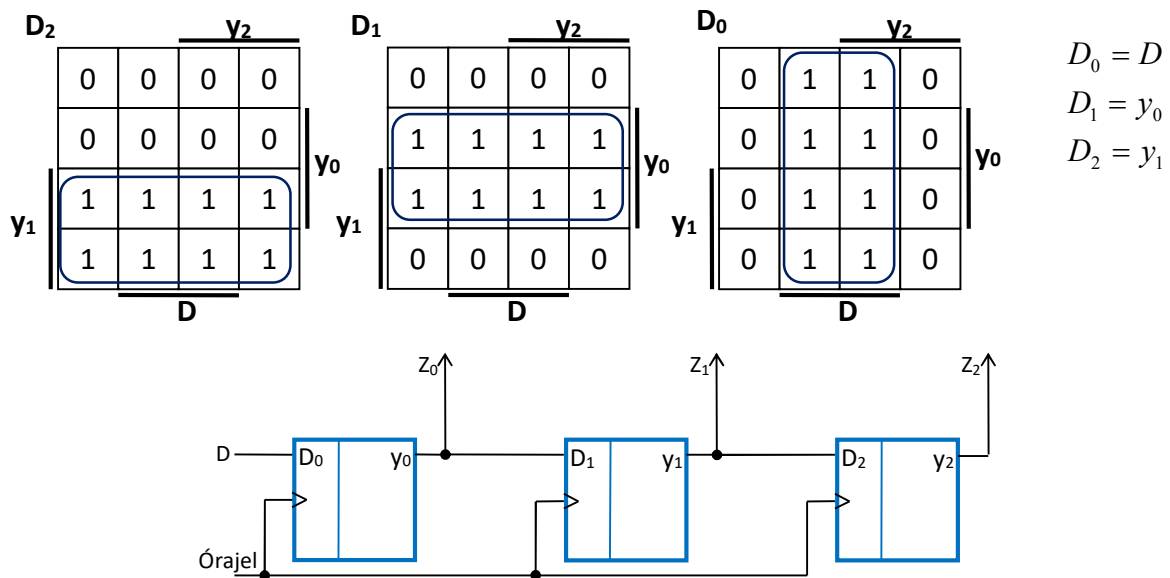
A hálózat előzetes állapotábrája:

$\begin{matrix} D \\ y \end{matrix}$	0	1	Z_2, Z_1, Z_0
a	a	b	000
b	c	d	001
c	e	f	010
d	g	h	011
e	a	b	100
f	c	d	101
g	e	f	110
h	g	h	111

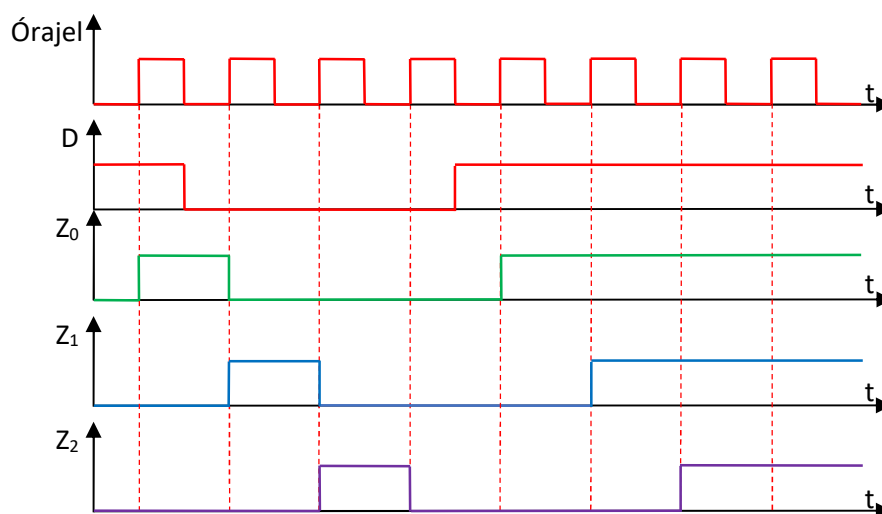
Moore-modell szerinti működés miatt válasszunk állapotkódot a kimenet alapján ($Z=y$). A kódolt állapotábra:

$\begin{matrix} D \\ y_2 y_1 y_0 \end{matrix}$	0	1	Z_2, Z_1, Z_0
$y_2 y_1 y_0$	$Y_2 Y_1 Y_0$		
000	000	001	000
001	010	011	001
011	110	111	011
010	100	101	010
100	000	001	100
101	010	011	101
111	110	111	111
110	100	101	110

Valósítsuk meg a hálózatot D flip-flopok felhasználásával. Mivel a D flip-flop karakterisztikus egyenlete alapján $Y=D$, ezért nincs szükség külön vezérlési táblára, mert az megegyezik a kódolt állapotábrával. Ez alapján a három flip-flop vezérlési egyenleteit meghatározhatjuk.



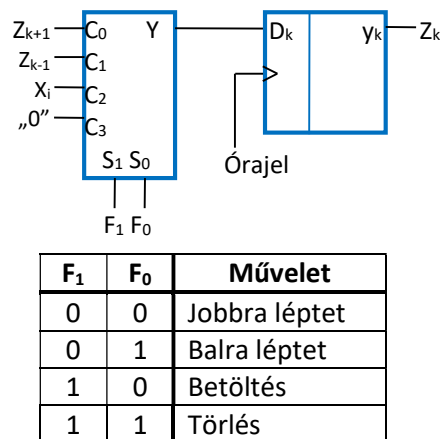
Amennyiben a hálózatot egyszerű él vezérelt flip-flopokkal valósítjuk meg, úgy rendszerhazárdot tartalmaz. A fenti hálózatban jól megfigyelhetjük egy sorrendi hálózat „emlékezetét”, hiszen az egyre korábbi időpontban érkezett bemeneti kombináció az egyre magasabb indexű kimeneten érhető el. Ez a struktúra tetszőleges bitszámúra bővíthető. A fenti áramkört soros bemenetű, párhuzamos kimenetű léptető regiszternek hívjuk. Az áramkör működését szemlélteti az alábbi idődiagram. Az ábrán feltételeztük, hogy a flip-flopok nulla kezdeti értékből indultak, majd a léptetőregiszter bemenetére az alábbi jelsorozat érkezett: $D=100011111$.



Általános regiszter

Multiplexer és D flip-flop összekapcsolásával kialakíthatunk egy olyan regisztert, amelynél vezérlő bemenetekre adott bináris kombináció segítségével választhatjuk ki az órajel hatására végrehajtandó műveletet. A kívánt bitszámnak megfelelően ilyen fokozatok egymás után kapcsolásával tetszőlegesen bővíthető a struktúra.

Néhány lehetséges művelet: fix „0”-ba állítás, fix „1”-be állítás, tetszőleges X_i bemenet betöltése, jobbra léptetés, balra léptetés, tétlen állapot. A megoldás elvi vázlatát mutatja az alábbi ábra.



A multiplexer bemeneteinek száma a megvalósítani kívánt funkciók számától függ. A fenti példában az F_1, F_0 bemenetek segítségével négyféle műveletből választhatunk.