

*Laboratóriumi gyakorlatok*

Fehér Gyula  
Kóré László

# Szekvenciális hálózatok

GYAKORLATOK



GÁBOR DÉNES  
FŐISKOLA



# 1. BEMUTATÓ VIZSGÁLATOK

A vizsgálat tárgya: *Sorrendi hálózatok működésének vizsgálata*

A vizsgálat célja: *A sorrendi hálózatok néhány kiemelt sajátosságának és gyakorlati vizsgálati módszerének bemutatása.*

Elméleti alapok: *Dr. Szittyá Ottó: Bevezetés az elektronikába.*

Az állományok helye: *C:\MC5DEMO\SZEKV\*

## 1.1 Ismert funkciójú kapcsolás működésének vizsgálata animációval

### 1.1.1 NOR kapus RS flip-flop vizsgálata

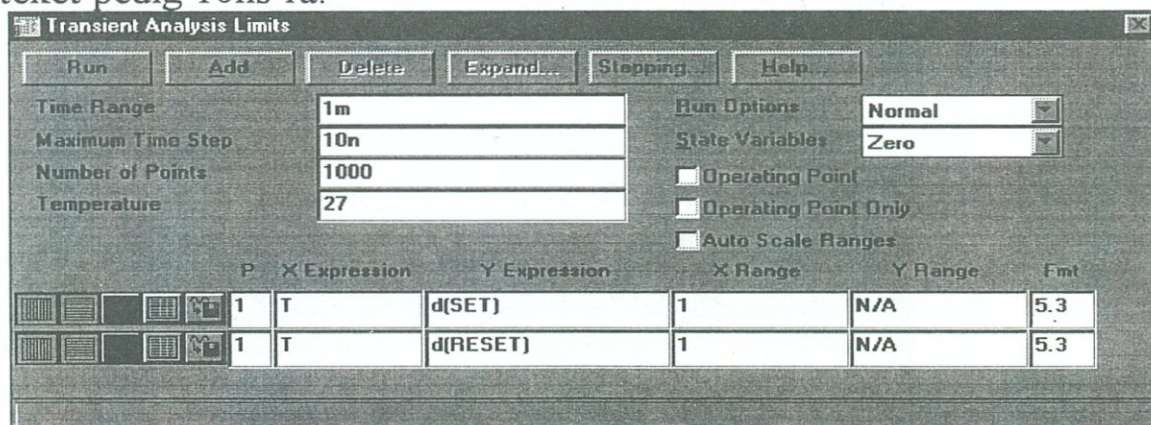
#### 1. lépés

Az MC5 indítását követően **töltse be** a NORFF1.CIR állományt. A betöltés után megjelenik a képernyőn a flip-flop működésének vizsgálatára alkalmas kapcsolás:

#### 2. lépés

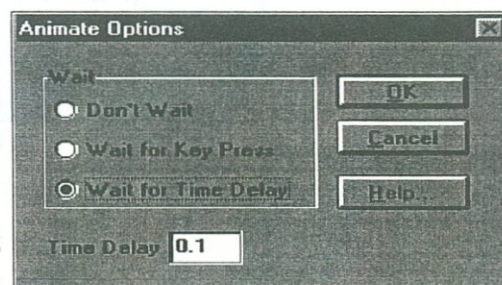
Az ábra jobb sarkában lévő felirat szerint **válassza ki a tranziens analízist!** Állítsa be a szimulációs paraméterek megfelelő értékeit! Mivel a kapcsolás vizsgálatát az animációs opció alkalmazásával kell elvégezni, a megjelenítés időzítéséhez egy olyan animációs stimulus generátort célszerű alkalmazni, mely a jelét 10 ns-onként változtatja, ezért a Time Range értékét állítsa be 1ms-ra, a Maximum Time Step értékét pedig 10ns-ra.

#### [1] 5.5.





#### 3. lépés

**Kapcsolja be az animációs opciót,** állítsa be annak paramétereit! Válassza ki a *Wait for Time Delay* működési módot, a *Time Delay* dobozba a 0.1 s értéket írja be!



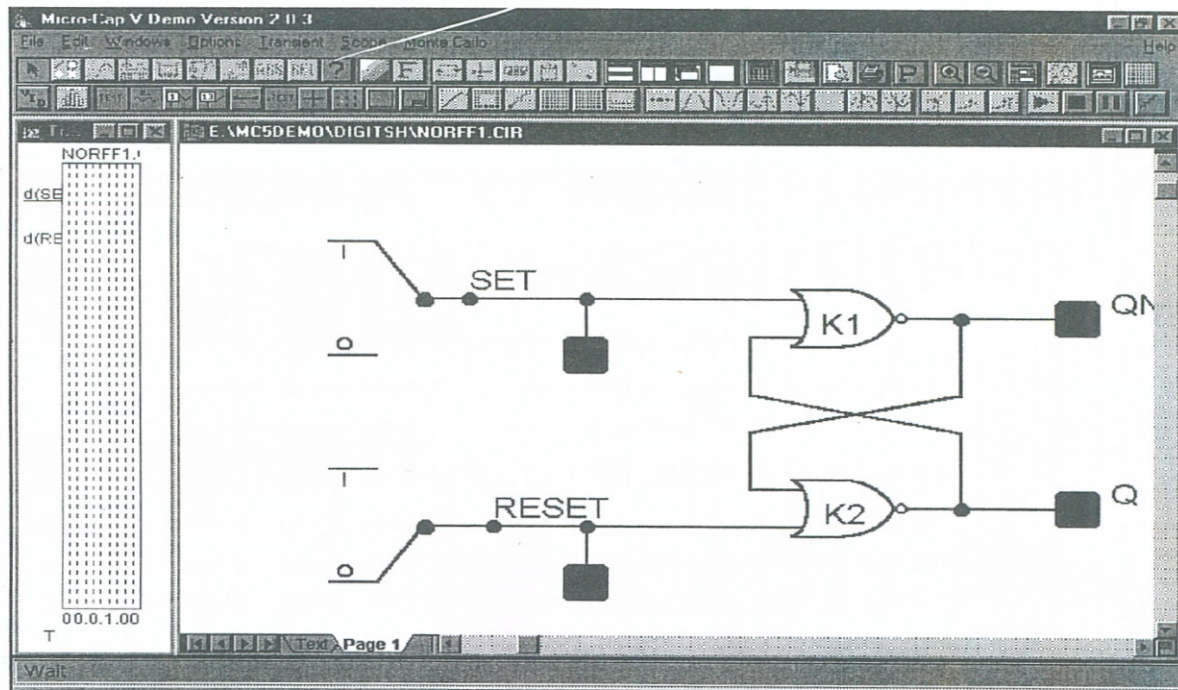
#### 4. lépés

**Indítsa el** a szimulációt! Amennyiben az indítás után a képernyőn nem látható a kapcsolási rajz, hanem csak az idődiagramok ablaka, kattintson a vízszintes  vagy a függőleges  ablakmegosztás ikonra, majd az idődiagram ablakot húzza össze kisebbre. Ügyeljen arra, hogy a

tényleges vizsgálat előtt mindegyik kapcsolót egyszer váltsa át, hogy a program a megfelelő kapcsoló-állást érzékelni tudja!

## 5. lépés

Vizsgálja meg a flip-flop működését, a kapcsolók segítségével állítsa a flip-flopot 0-ba, majd 1-be, végül ismét 0-ba! Adjon a bemenetre 11 jelkombinációt, vizsgálja meg, miért tiltott ez a kombináció, hogyan reagál erre az kapcsolás!



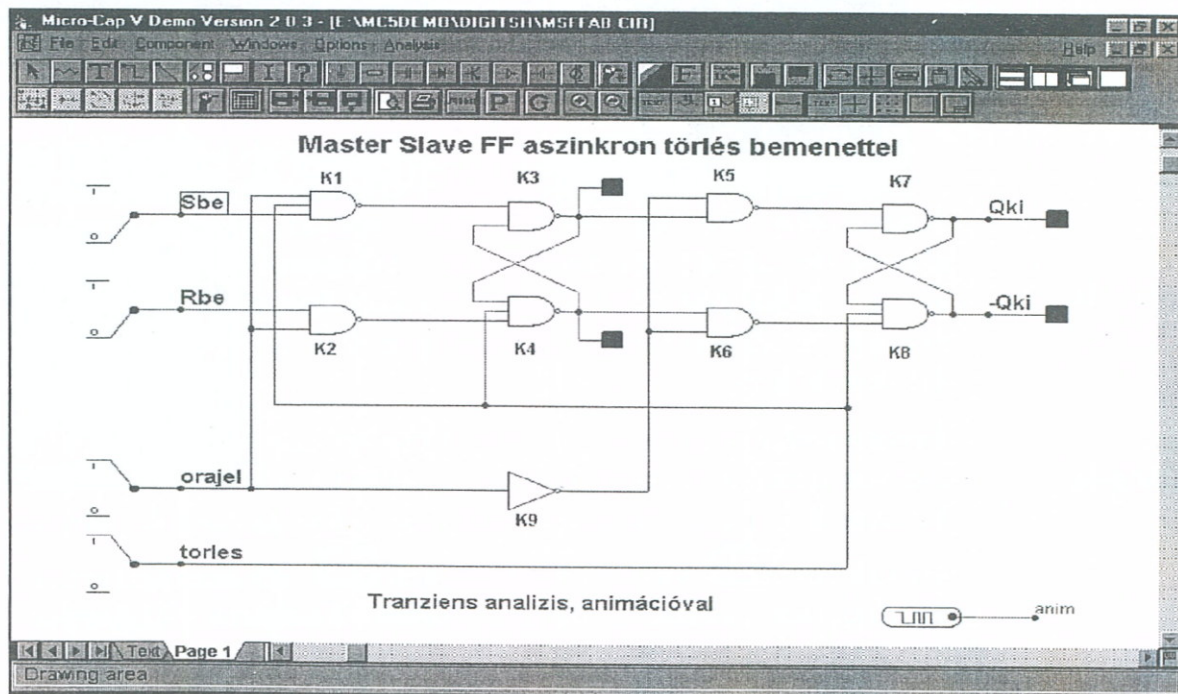
[1] 5.5.2.

**Vizsgálati eredmény:** A tiltott bemeneti jelkombináció esetén a flip-flop mindkét kimenete 1-be áll be, ezáltal nem teljesül az a követelmény, hogy a két kimeneti jel egymás negáltja legyen.

## 1.1.2 Master-Slave flip-flop vizsgálata

### 1. lépés

Töltse be az MSFFAB.CIR állományt!



[1] 5.5.2.2.

- 2. lépés** A menüből **válassza ki** a tranziens analízist! **Állítsa be** a szimulációs paraméterek megfelelő értékeit (lásd az előző vizsgálatot)!
- 3. lépés** **Kapcsolja be** az animációs opciót, **állítsa be** az animációs paramétereket (szintén az előző vizsgálatnál megismert módon)!
- 4. lépés** **Indítsa el** a szimulációt! Vizsgálja meg a flip-flop működését! Elsőként ellenőrizze az aszinkron törlést, majd bemenet a kapcsoló segítségével állítsa a flip-flopot 0-ba. Győződjön meg róla, nem lesz-e átlátszó a flip-flop az órajel 1 értéke esetén! Az S és R bemenetek vezérlésével vegye fel a flip-flop állapot-táblázatát! Készítse el ennek alapján a működési táblázatot!

**Vizsgálati eredmény:** Az RS flip-flop *állapot-táblázata*:

n.ütem	n+1. ütem			
	RS=00	RS=01	RS=11	RS=10
0	0	1	tiltott	0
1	1	1	tiltott	0

Az RS flip-flop *működési táblázata*:

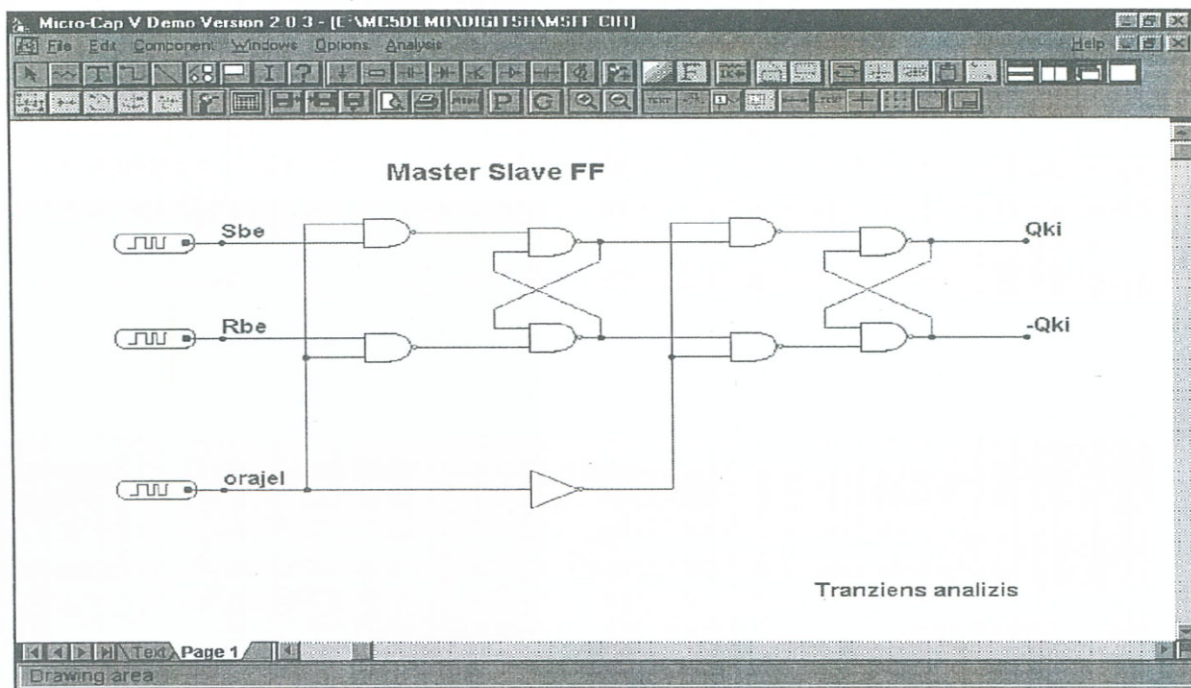
n.ütem → n+1.ütem	R	S
0 → 0	X	0
0 → 1	0	1
1 → 0	1	0
1 → 1	0	X

- 5. lépés** **Vizsgálja meg** a tiltott bemeneti jelkombináció hatását!

**Vizsgálati eredmény:** Az órajel 1 szintje esetében a Master mindkét kimenete 1 értéket vesz fel, majd miután az órajel visszaáll 0-ra, oszcillálni kezd a Master és a Slave kimenete is. Ha változatlan bemeneti jelkombináció mellett az órajel ismét 1 értékű lesz, a Master mindkét kimenete újra állandó 1 értékre áll be. Ez a helytelen működési állapot csak akkor szűnik meg, ha töröljük a flip-flopot, vagy megengedett bemeneti jelkombinációt írunk a flip-flopba.

## 1.2 Ismert funkciójú kapcsolás működésének vizsgálata a jelek időfüggvényei alapján

- 1. lépés** **Töltse be** az MSFF.CIR állományt!



## 2. lépés

**Ellenőrizze** a vizsgálathoz szükséges stimulus generátorok paramétereinek beállítását, ha nem megfelelő értékűek, változtassa meg őket! A flip-flopot alkotó kapuk jelterjedési késési idejét a Text lapon a modell-paraméterek között megtalálhatja, ez átlagosan 10ns. Az előző animációs vizsgálat alapján a leghosszabb jelút 4 kapun vezet keresztül, a flip-flop várható jelterjedési késési ideje az órajel bemenettől a kimenetig 40ns. Ennek megfelelően válassza az órajel periódusidejét 200 ns-ra, kitöltési tényezőjét pedig 50%-ra. Állítsa be az S és az R jelet szolgáltató stimulus generátor paramétereit úgy, hogy az összes megengedett bemeneti jelkombináció hatását meg tudja vizsgálni!

```

Micro-Cap V Demo Version 2.0.3 - [C:\MS5DEMO\DIGITS\HMSTT.CIR]
File Edit Component Windows Options Analysis Help
*****
.MODEL DLY_TTL UGATE (TPLHTY=11NS TPLHMx=22NS TPHLTY=8NS TPHLMx=15NS)

.DEFINE OJEL
+LABEL=ASTART
+0nS 0
+50nS 1
+150nS 0
+200nS GOTO ASTART 100 TIMES

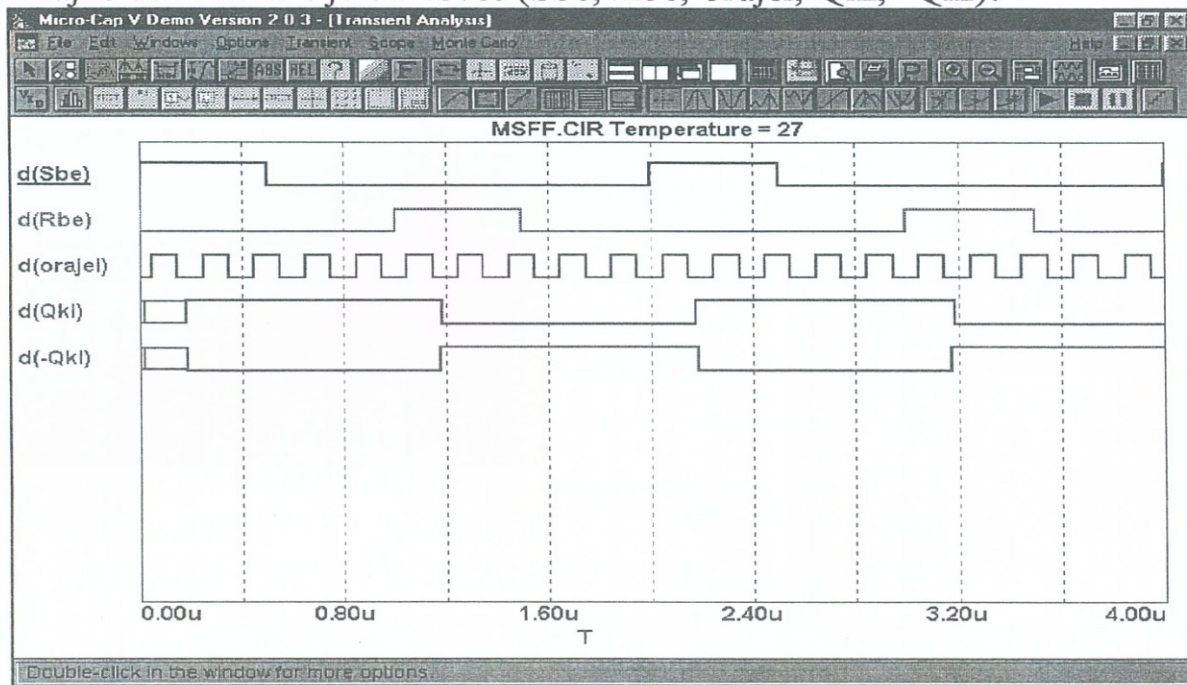
.DEFINE SJEL
+LABEL=BSTART
+0nS 1
+0.5uS 0
+2uS GOTO BSTART 100 TIMES

.DEFINE RJEL
+LABEL=CSTART
+0nS 0
+1uS 1
+1.5uS 0
+2uS GOTO CSTART 100 TIMES
    
```

## 3. lépés

A menüből **válassza ki a tranziens analízist!** Állítsa be a szimulációs paraméterek megfelelő értékeit! A szimulációs vizsgálat idejét (*Time Range*) vegye az S és R jelek periódusidejének kétszeresére (4μs)! Írja

be a tranziens analízis párbeszéd-ablak *Y Expression* dobozaiba a kirajzoltatni kívánt jelek nevét (Sbe, Rbe, órajel, Qki, -Qki)!



#### 4. lépés

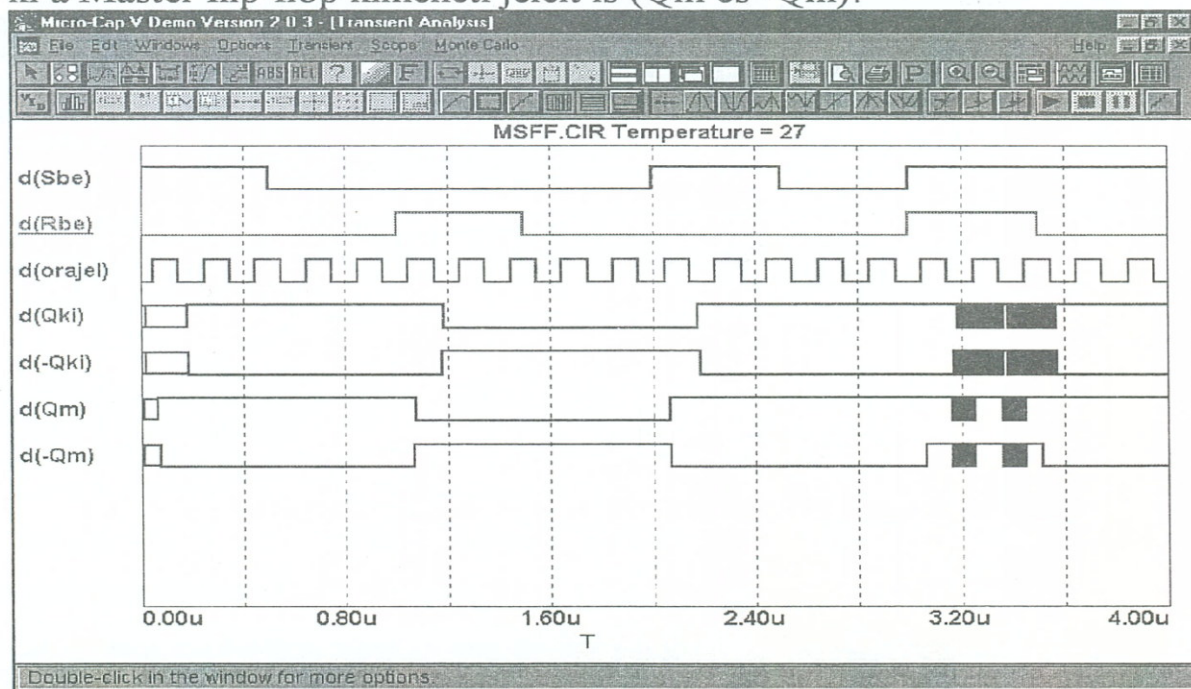
Értékelje a kapott eredményeket:

**Vizsgálati eredmény:**

- Mivel a flip-flopnak nincs alapállapot-beállító bemenete, ezért a kimenet mindaddig határozatlan állapotú, amíg az első órajel hatására a bemeneti jeleknek megfelelően nem áll be a Slave flip-flop kimenete.
- Az időfüggvények alapján ellenőrizve az animációs vizsgálattal felvett állapot-táblázatot, látható, hogy a kétféle vizsgálati módszerrel kapott eredmény megegyezik.

#### 5. lépés

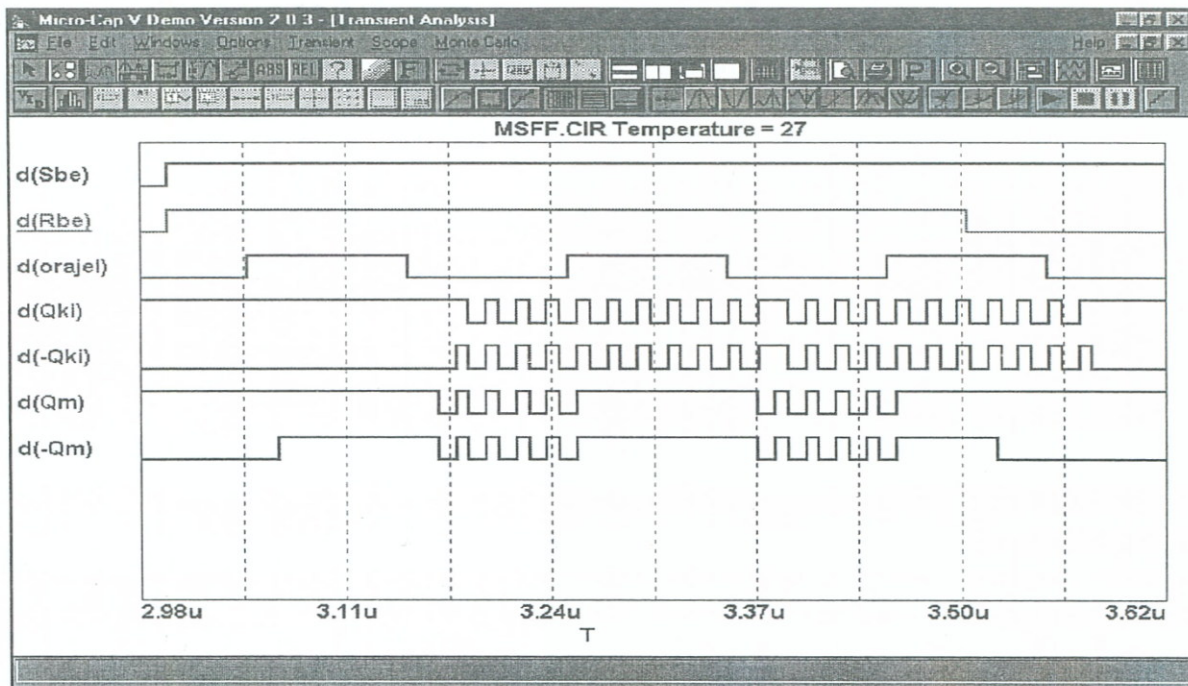
Változtassa meg az S és az R jelet szolgáltató stimulus generátor paramétereit úgy, hogy a tiltott bemeneti jelkombináció hatása is megfigyelhető legyen! Futtassa le újra a tranziens analízist! Rajzoltassa ki a Master flip-flop kimeneti jeleit is (Qm és -Qm)!



**Vizsgálati eredmény:** Az ábrán látszik, hogy a tiltott bemeneti jelkombináció hatására a flip-flop helytelenül működik, gerjed.

**6. lépés**

A pontosabb kiértékelés érdekében nagyítsa ki a tiltott bemeneti kombinációhoz tartozó jelszakaszt!



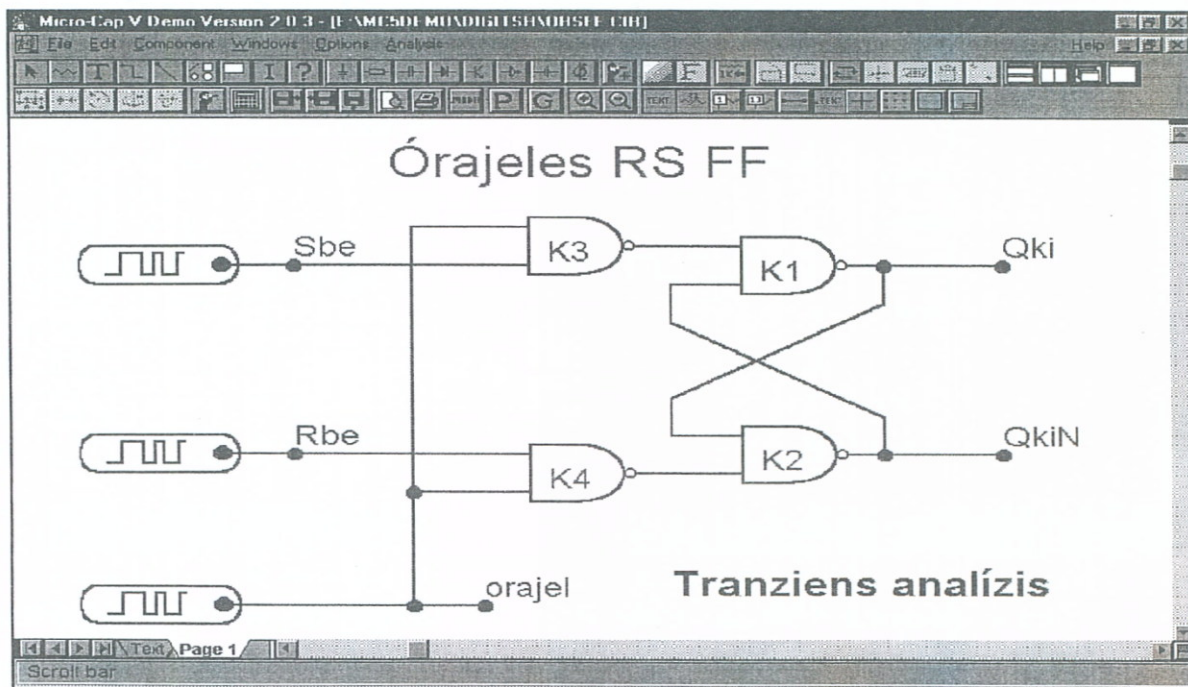
**Értékelje** a kapott eredményeket:

**Vizsgálati eredmény:** Az ábrán jól megfigyelhető az a gerjedés, amit már az animációs vizsgálat is kimutatott.

**1.3 Ismert funkciójú kapcsolás sebességi jellemzőinek meghatározása**

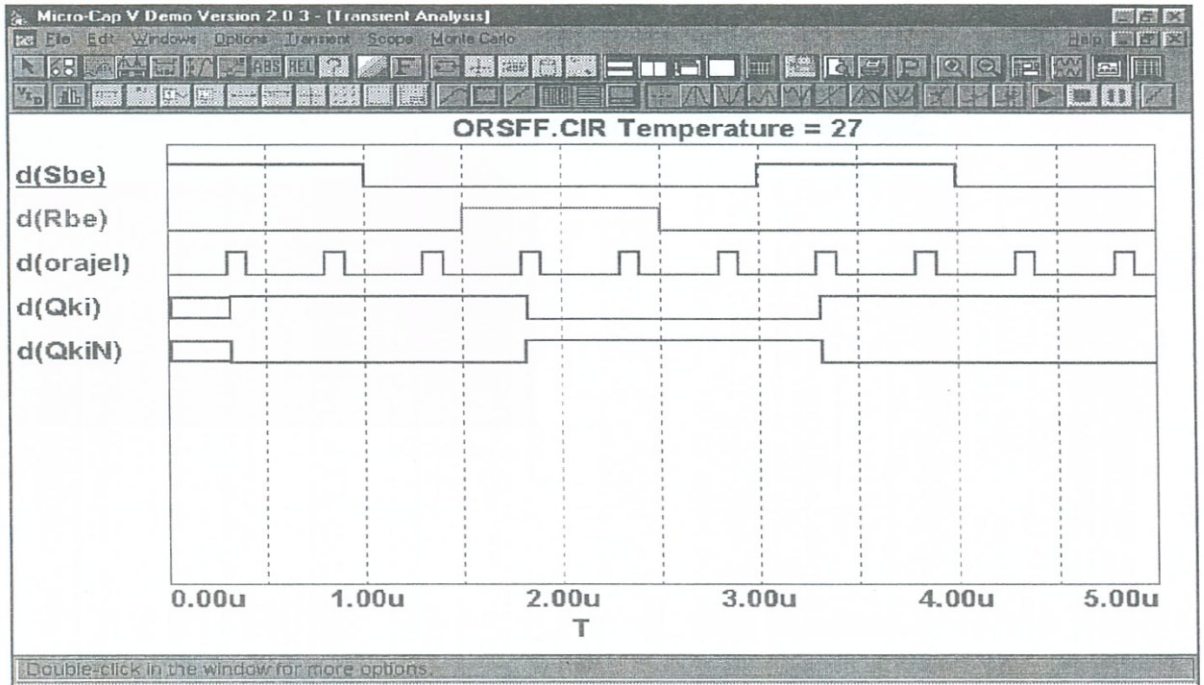
**1. lépés**

Töltse be az ORSFF1.CIR állományt:



2. lépés

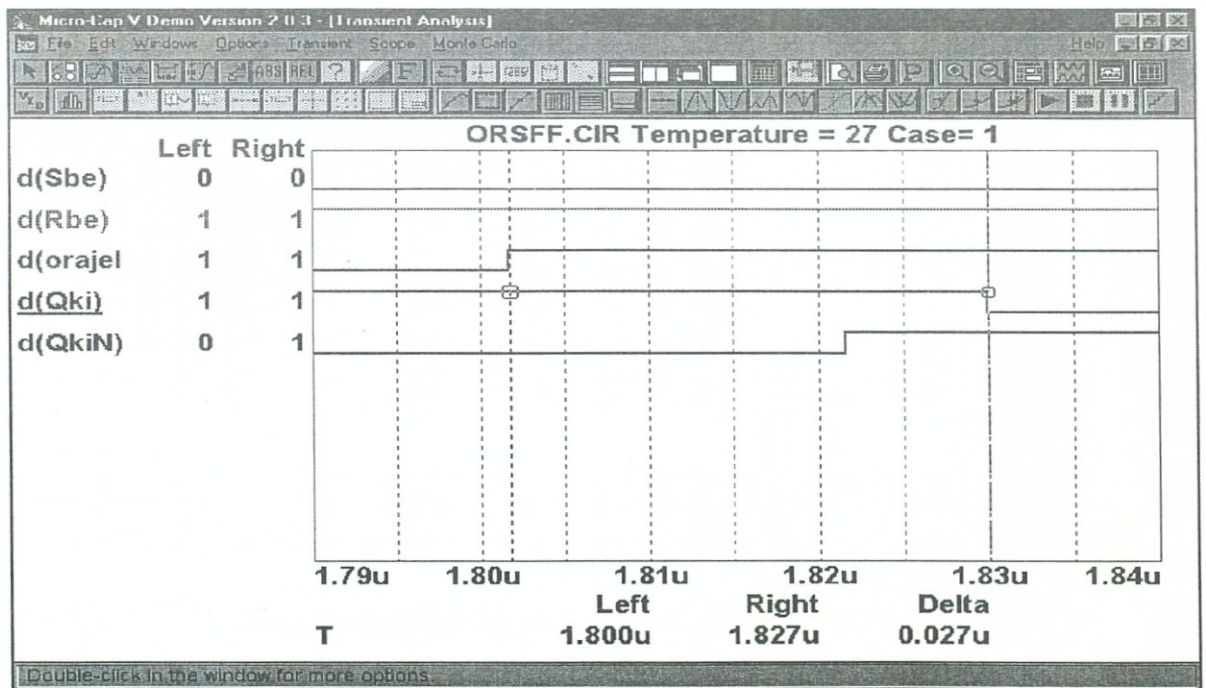
Válassza ki a tranziens analízist! Indítsa el a szimulációt! A szimuláció befejeződése után az idődiagramok alapján ellenőrizze a működést!



3. lépés

Nagyítsa ki az ábrának azt a részletét, ahol az órajel H szintre váltása után megváltozik a Qki kimeneti jel! **Kapcsoljon át** kurzor módba, és határozza meg a flip-flop órajel-bemenete és Qki kimenet között mérhető jelterjedési késési időt! **Állítsa** a bal kurzort az órajel felfutó élére, a jobb kurzort pedig a Qki kimeneti jel ehhez legközelebb eső éléhez (jelen ábrán ez a lefutó él). **Olvassa le** az ábra alatt a Delta T értéket!

**Vizsgálati eredmény:** A jelterjedési késési idő  $0.027\mu$  azaz  $27\text{ns}$ .

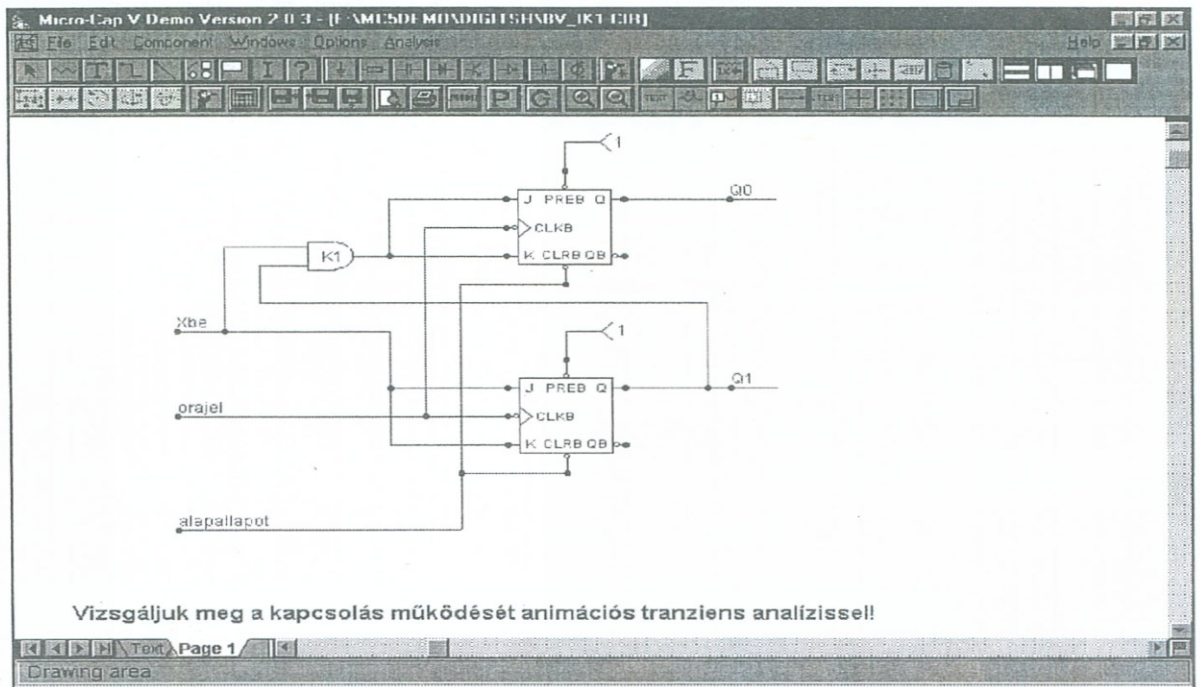




## 1.4 Ismeretlen funkciójú kapcsolás működésének vizsgálata animációval

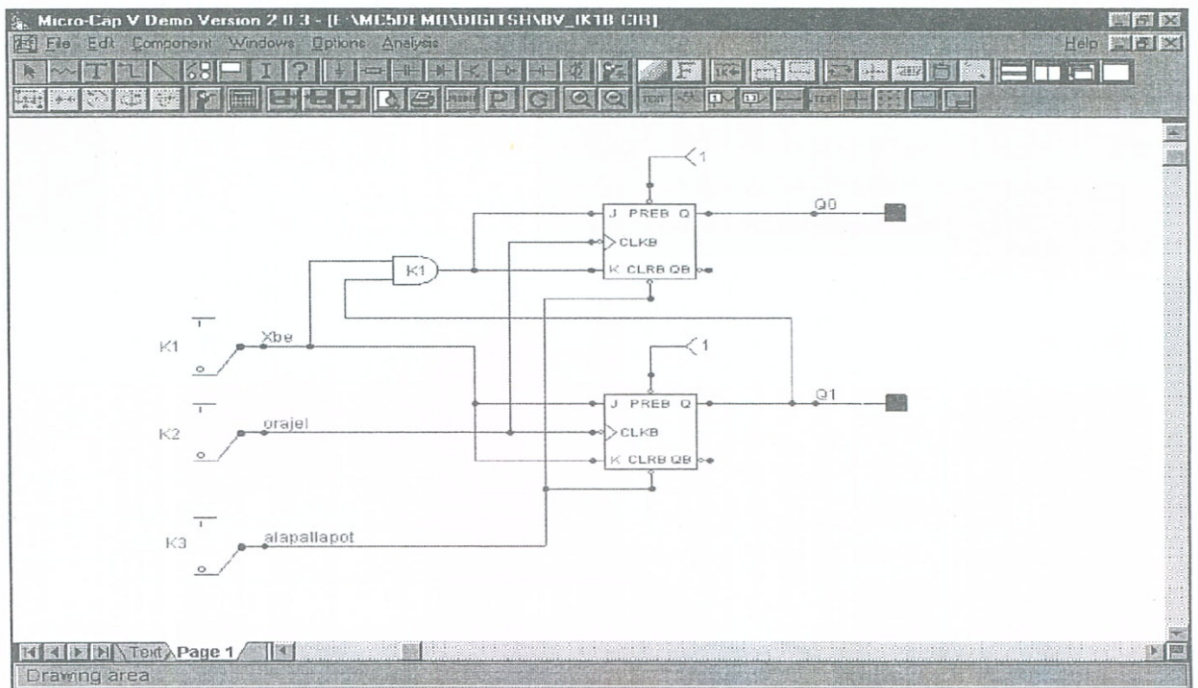
### 1. lépés

Töltse be a BV\_IK1.CIR állományt:



### 2. lépés

Egészítse ki a kapcsolást a bemeneti jelet szolgáltató kapcsolókkal és a jelek vizsgálatát lehetővé tevő LED-ekkel!



### 3. lépés

Rajzolja be a kapcsolásba az animációs stimulus generátort! A generátor paramétereit átmásolhatja valamelyik másik, animációs vizsgálattal dolgozó kapcsolás Text lapjáról:

### 4. lépés

Válassza ki a tranziens analízist és az animációs módot, állítsa be a szimulációs és animációs paramétereiket, **indítsa el** a szimulációt!

## 5. lépés

Állítsa alapállapotba a kapcsolást a K3 kapcsolóval, majd a K1 és K2 kapcsolót működtetve vizsgálja meg a kapcsolást, vegye fel a kódolt állapottáblázatot!

**Vizsgálati eredmény:**

A szimulációs vizsgálat alapján az alábbi állapot-táblázat kapható:

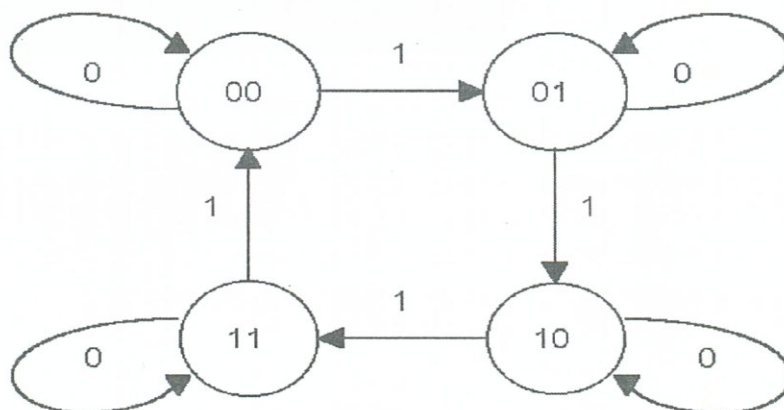
n.ütem $Q_1^n Q_0^n$	n+1.ütem	
	$Q_1^{n+1} Q_0^{n+1}$	$Q_1^{n+1} Q_0^{n+1}$
	X=0	X=1
00	00	01
01	01	10
10	10	11
11	11	00

## 6. lépés

Rajzolja meg a hálózat állapot-diagramját! Értékelje ki az eredményeket, állapítsa meg, milyen funkciót valósít meg ez a hálózat!

**Eredmény:**

Az állapot-táblázat alapján az alábbi állapot-diagram rajzolható meg:

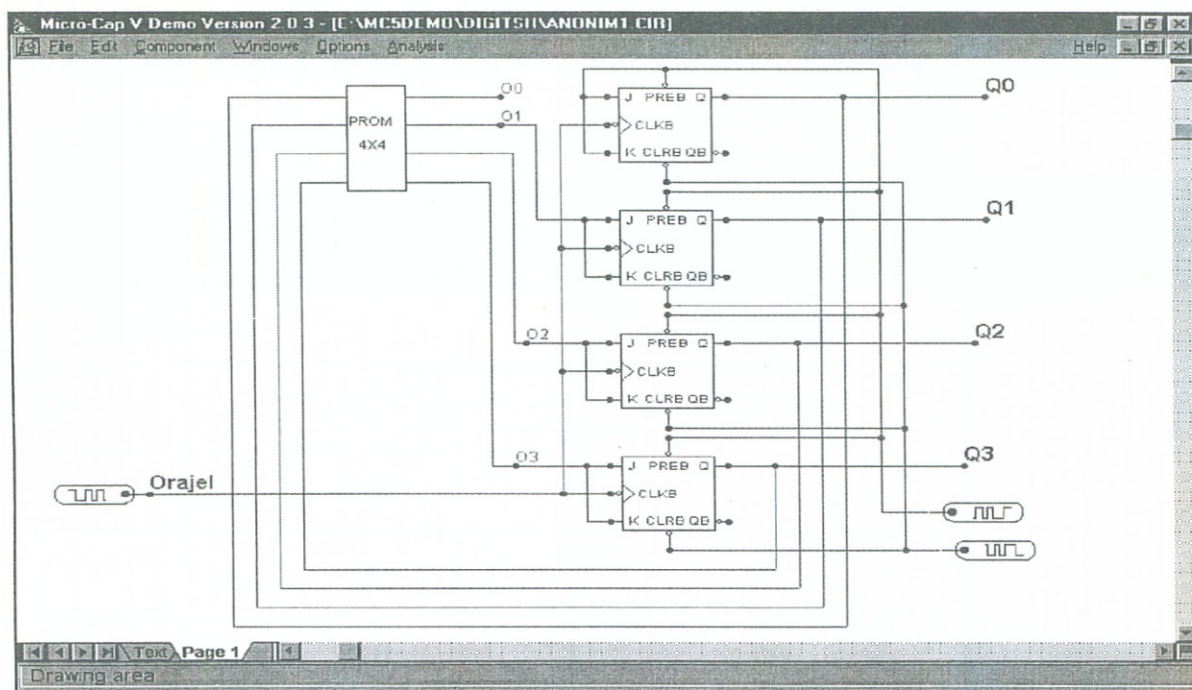


Ennek alapján megállapítható, hogy ez a hálózat egy 2 bites bináris start-stop számláló.

## 1.5 Ismeretlen funkciójú kapcsolás működésének vizsgálata a jelek időfüggvényei alapján

## 1. lépés

Töltse be az ANONIM1.CIR állományt!



## 2. lépés

**Értelmezze** a hálózat működését a kapcsolási rajz alapján!

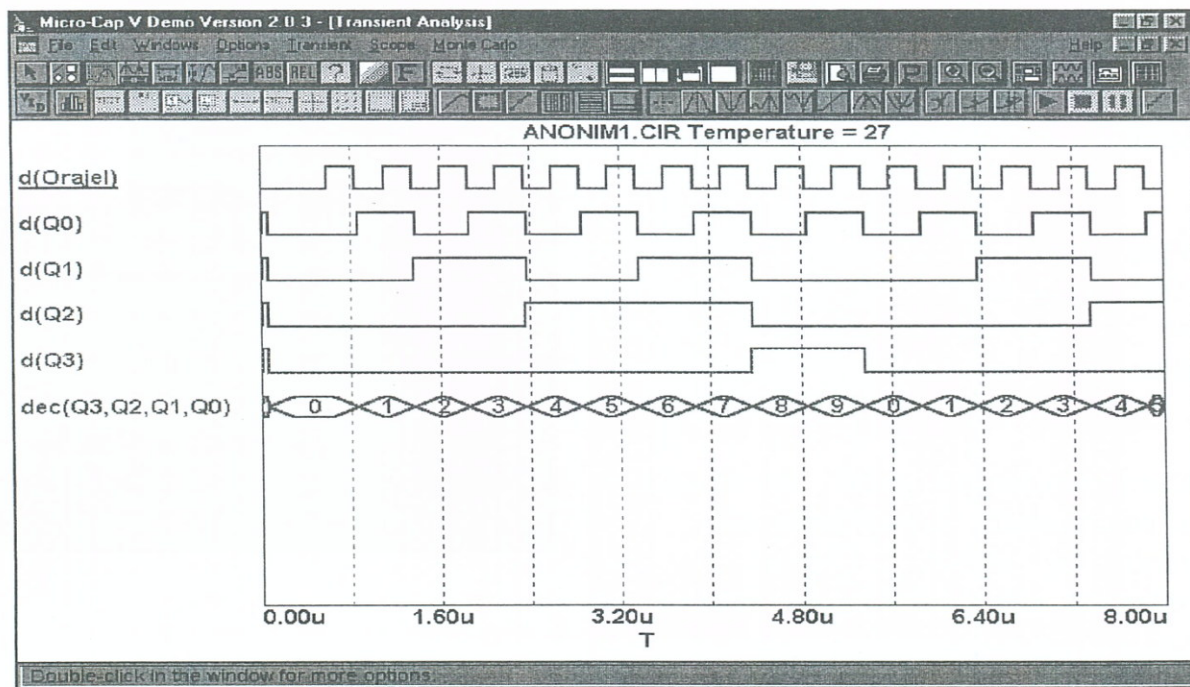
### Megoldás:

Jelen példánál a működés a kapcsolási rajz alapján nem fejthető vissza, mert a kombinációs hálózatrészeket egy PROM valósítja meg. A felépítés alapján csak néhány elvi következtetés mondható ki. Egyrészt a hálózat nem kap bemeneti jelet (eltekintve az órajeltől), ami a számlálókra jellemző, még hozzá azoknak is csak az egyszerűbb típusaira, melyek a számlálási irányt nem képesek változtatni. Ugyancsak a számlálókra jellemző az is, hogy nincs külön kimeneti kombinációs hálózat, a hálózat kimeneti jeleit közvetlenül a tárolóelemek szolgáltatják, azaz ez egy Moore-modell szerinti szinkron sorrendi hálózat. A tárolóelemek száma 4, tehát a hálózatnak maximum 16 állapota lehet, de a működés vizsgálata nélkül nem dönthető el, mennyi ebből a megengedett és mennyi a tiltott állapot.

**Összefoglalva:** a hálózat nagy valószínűséggel valamilyen számláló, tehát a vizsgálatát is ennek megfelelően kell elvégezni.

## 3. lépés

Válassza ki a menüből a tranziens analízist, **állítsa be** a szimulációs vizsgálat idejét (*Time Range*). A 2.lépésben leírtak alapján célszerű ezt nagyobbra választani, mint 16 órajel-periódus ideje. A párbeszédablakban az *Y Expression* dobozokba írja be a vizsgálandó jelek nevét (Órajel, Q0, Q1, Q2, Q3)! Mivel a hálózat nagy valószínűséggel valamilyen számláló, a kimeneti jelek bináris formában kirajzoltatása mellett ajánlatos hexa formában is ábrázoltatni őket.



#### 4. lépés

Vonja le a végső következtetést!

#### Megoldás:

Az időfüggvények alapján megállapítható, hogy a hálózat a 9-es állapot után visszalép a 0 állapotba, a megengedett állapotok száma 10, ezek kódolása BCD kód szerinti, tehát ez a hálózat egy BCD kódú számláló (egy dekádos, azaz egy BCD számjegyes).

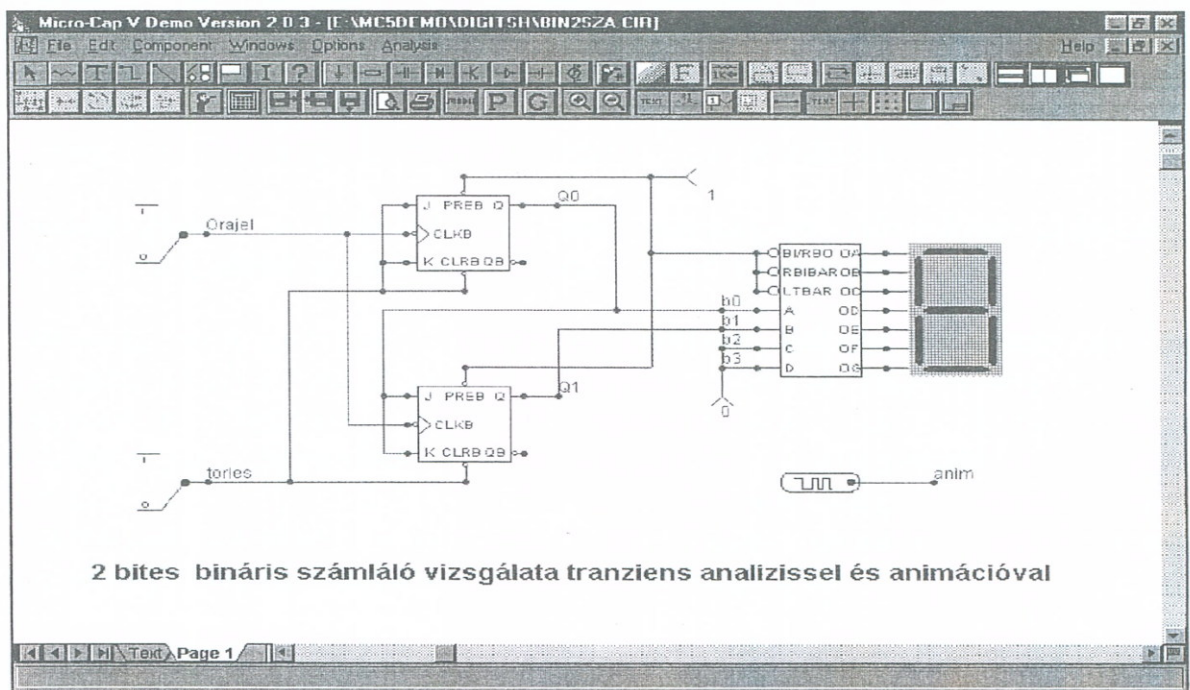
## 2. BEMUTATÓ GYAKORLATOK

A gyakorlat tárgya:	<i>Sorrendi hálózatok vizsgálata és tervezése</i>
A gyakorlat célja:	<i>A sorrendi hálózatok különféle vizsgálati és megvalósítási lehetőségeinek bemutatása. A legelterjedtebb flip-flop alaptípusok bemutatása, alkalmazásuk gyakorlása. A sorrendi hálózatok alapvető tervezési és megvalósítási tevékenységeinek gyakorlása. A logikai építőelemek nem zérus késleltetési ideje miatt fellépő sebességkorlátok bemutatása. Típusfeladatok megoldási lehetőségeinek bemutatása, a megoldások gyakorlása. A hibafeltárás és hibelhárítás gyakorlása</i>
Elméleti alapok:	<i>Dr. Szittyá Ottó: Bevezetés az elektronikába.</i>
Az állományok helye:	<i>C:\MC5DEMO\SZEKV\</i>

### 2.1 Ismert funkciójú kapcsolás működésének vizsgálata animációval

#### 1. lépés

Töltse be az M\_2BKA7.CIR állományt. Ez egy olyan mintakapcsolást tartalmaz, amely megkönnyíti a saját kapcsolásaink elkészítését, ugyanis rendelkezésünkre bocsájtja a bemeneti jelforrásokat és a kimeneti jelek megjelenítésére szolgáló alkatrészeket. Az M\_2BKA7.CIR például a bemeneti jelek előállításához két kapcsolót, a kimeneti (max 3 bites) bináris jelek megjelenítéséhez egy 7 szegmenses kijelzőt tartalmaz, a dekódoló áramkörével együtt.



- 2. lépés** Nevezze át a kapcsolást (pl. BIN2SZA.CIR névre) és mentse el!
- 3.lépés** Állítsa össze a 2 bites bináris számláló kapcsolási rajzát! Alkalmazza az MC5 saját, ideális JK flip-flopjait (ezeket a Component, Digital Primitives, Edge-Triggered Flip-Flops menüben találja meg).
- 4.lépés** Válassza ki a menüből a tranziens analízist, állítsa be a szimulációs paramétereket a megismert módon (lásd *Ismert funkciójú kapcsolás működésének vizsgálata animációval* c. fejezet). Kapcsolja be az animációs módot, válassza meg az animációs paramétereket a javasolt módon!
- 5. lépés** Indítsa el a szimulációt, a *törlés* kapcsolóval állítsa alapállapotba a számlálót, majd az *Órajel* kapcsoló működtetésével ellenőrizze a működést! Vizsgálja meg, hatásos-e az órajel a törlés jel aktív állapotában! Vizsgálja meg, az órajel felfutó vagy lefutó éle hatására változik-e a kimeneti jel!
- Vizsgálati eredmény:** A *törlés* jel még az órajel aktív állapotában is elsőbbséget élvez az órajeles működtetéssel szemben. A kimeneti jel az órajel lefutó éle után változik meg.
- 6. lépés** Vegye fel a számláló (kódolt) állapot-táblázatát:

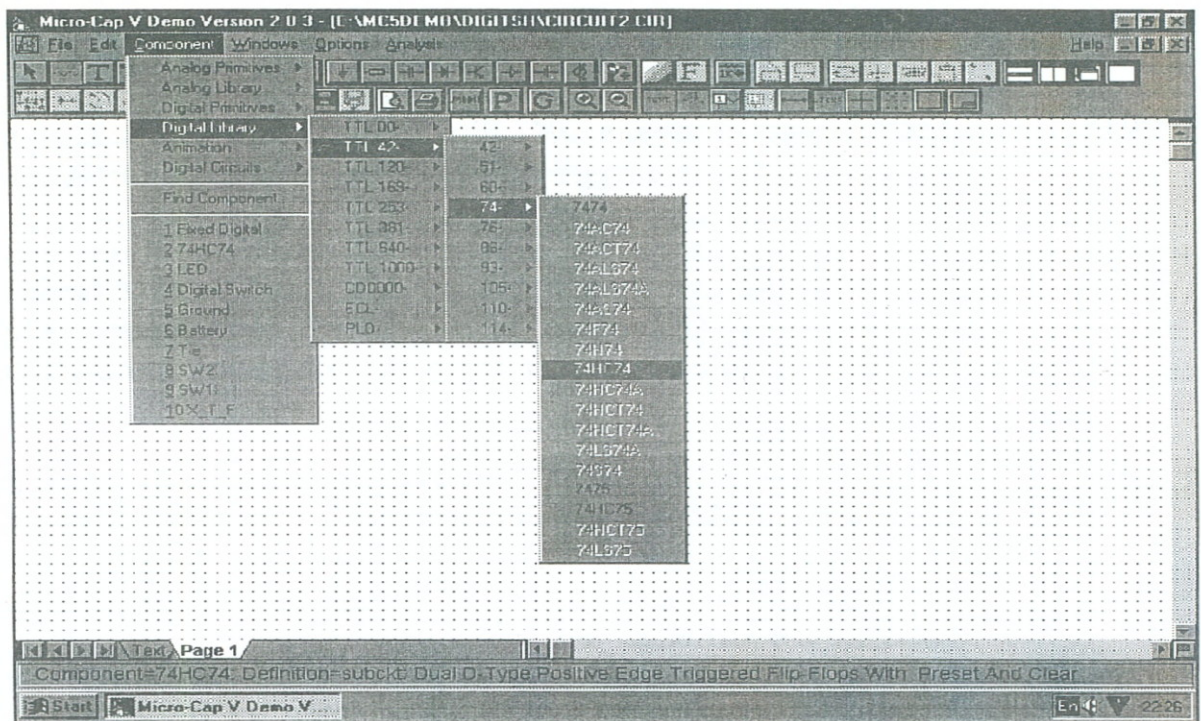
Vizsgálati eredmény:

n.ütem	n+1.ütem
$Q_1^n Q_0^n$	$Q_1^{n+1} Q_0^{n+1}$
00	01
01	10
10	11
11	00

## 2.2 Ismert funkciójú kapcsolás sebességi jellemzőinek meghatározása

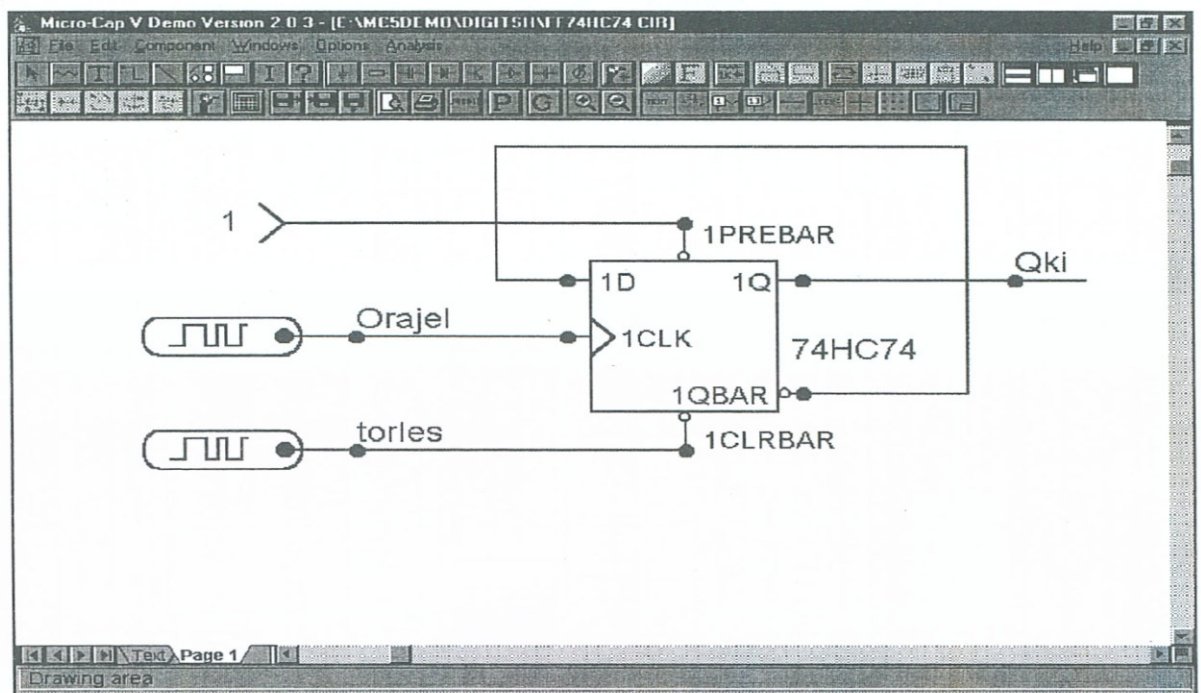
### 2.2.1 A 74HC76 típusú flip-flop jelterjedési késési idejének meghatározása

- 1. lépés** Nyisson egy új kapcsolást! Válassza ki a *Component, Digital Library, TTL 42-, 74* menüből a 74HC74 élvezérelt D flip-flopot:



## 2. lépés

Csatolja vissza a flip-flop Q kimeneti jelét a D bemenetre, azaz készítsen a flip-flopból egy 1 bites frekvenciaosztót! **Egészítse ki** a kapcsolást az órajelet és az alapállapot-beállító jelet szolgáltató stimulus generátorokkal! **Nevezze el** a jeleket!



## 3. lépés

Állítsa be a stimulus generátorok paramétereit a Text lapon! Az alapállapot-beállító stimulus generátor az indítás után csak egy rövid ideig tartó 0 szintű pulzust hozzon létre. Az órajelet szolgáltató stimulus generátor kimenetén periodikus jelet adjon.

```

.Micro-Cap V Demo Version 2.0.3 - [FF74HC74.CIR]
File Edit Component Windows Options Analyze Help

.DEFINE torol
+ONS 1
+10NS 0
+60NS 1

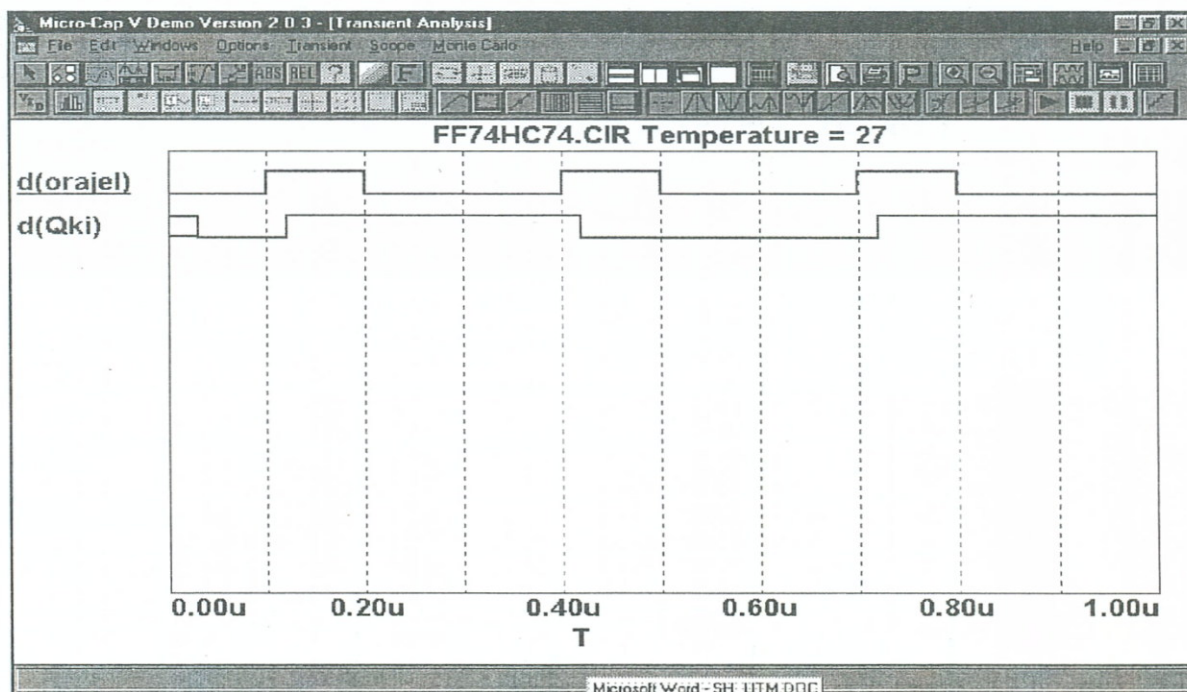
.DEFINE OJEL
+LABEL=ASTART
+0ns 0
+100ns 1
+200ns 0
+300ns GOTO ASTART 100 TIMES

Text/ Page 1 /
Text area

```

## 4. lépés

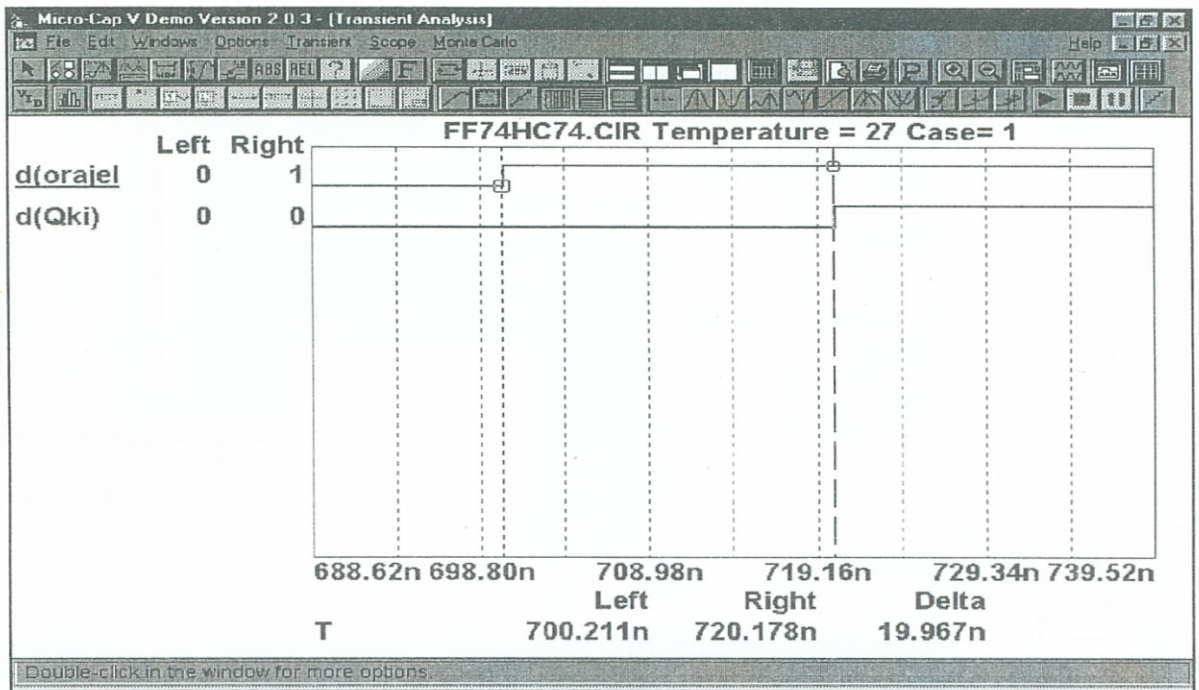
Válassza ki a menüből a tranziens analízist! **Állítsa** be a szimulációs paramétereket! Mivel elegendő a vizsgálatot csak néhány órajel-periódusnyi ideig végezni, tehát a szimulációs idő lehet például 1  $\mu$ s. **Adja meg**, melyik jeleket akarja kirajzoltatni (az órajel és a kimeneti jel a fontos)! **Indítsa el** a szimulációt!



## 5. lépés

**Nagyítsa ki** az ábrának azt a részletét, ahol az órajel H szintre váltása után megváltozik a Qki kimeneti jel! **Kapcsoljon át** kurzor módba, és határozza meg a flip-flop órajel-bemenete és Qki kimenet között mérhető jelterjedési késési időt! **Állítsa** a bal kurzort az órajel felfutó élére, a jobb kurzort pedig a Qki kimeneti jel ehhez legközelebb eső éléhez (jelen ábrán ez a felfutó él). **Olvassa le** az ábra alatt a Delta T értéket:





**Vizsgálati eredmény:** 19.967n azaz kerekítve 20 ns a mért jelterjedési késési idő.

## 2.3 Kapcsolás megtervezése, működésének ellenőrzése

### 2.3.1 Tervezés kapuk és flip-flopok felhasználásával, a megadott állapot-táblázat alapján

**Feladat:** Tervezze meg az alábbi állapot-táblázattal megadott szinkron sorrendi hálózatot kapuk és T flip-flopok felhasználásával

n.ütem	n+1.ütem		$Z_2Z_1$	
	X=0	X=1	X=0	X=1
a	b	e	00	00
b	c	a	00	00
c	d	a	00	00
d	a	a	01	01
e	f	a	00	00
f	g	e	00	00
g	h	e	00	00
h	e	e	10	10

**Állapotkód:**

Állapot	Kód
a	000
b	001
c	010
d	011
e	100
f	101
g	110
h	111

## 1. lépés

Készítse el a kódolt állapot-táblázatot!

Megoldás:

n. ütem $Q_3^n Q_2^n Q_1^n$	n+1. ütem		$Z_2 Z_1$	
	X=0	X=1	X=0	X=1
000	001	100	00	00
001	010	000	00	00
010	011	000	00	00
011	000	000	01	01
100	101	000	00	00
101	110	100	00	00
110	111	100	00	00
111	100	100	10	10

## 2. lépés

Írja fel a T flip-flop működési táblázatát!

Megoldás:

n. ütem $\rightarrow$ n+1. ütem	T
0 $\rightarrow$ 0	0
0 $\rightarrow$ 1	1
1 $\rightarrow$ 0	1

1 → 1	0
-------	---

**3. lépés**

A kódolt állapot-táblázat és a T flip-flop működési táblázata alapján **tervezze meg** a bemeneti kombinációs hálózatot!

**Megoldás:**

**Q3<sup>n+1</sup>**

Q3 <sup>n</sup> Q2 <sup>n</sup> \ Q1 <sup>n</sup> X	00	01	11	10
00	0	1	0	0
01	0	0	0	0
11	1	1	1	1
10	1	0	1	1

**T3**

Q3 <sup>n</sup> Q2 <sup>n</sup> \ Q1 <sup>n</sup> X	00	01	11	10
00	0	1	0	0
01	0	0	0	0
11	0	0	0	0
10	0	1	0	0

**T3 =  $\bar{Q}2 \cdot \bar{Q}1 \cdot X$**

**Q2<sup>n+1</sup>**

Q3 <sup>n</sup> Q2 <sup>n</sup> \ Q1 <sup>n</sup> X	00	01	11	10
00	0	0	0	1
01	1	0	0	0
11	1	0	0	0
10	0	0	0	1

**T2**

Q3 <sup>n</sup> Q2 <sup>n</sup> \ Q1 <sup>n</sup> X	00	01	11	10
00	0	0	0	1
01	0	1	1	1
11	0	1	1	1
10	0	0	0	1

**T2 =  $Q2 \cdot X + Q1 \cdot \bar{X}$**

**Q1<sup>n+1</sup>**

Q3 <sup>n</sup> Q2 <sup>n</sup> \ Q1 <sup>n</sup> X	00	01	11	10
00	1	0	0	0
01	1	0	0	0
11	1	0	0	0
10	1	0	0	0

**T1**

Q3 <sup>n</sup> Q2 <sup>n</sup> \ Q1 <sup>n</sup> X	00	01	11	10
00	0	0	1	1
01	0	0	1	1
11	0	0	1	1
10	0	0	1	1

**T1 = Q1**

**4. lépés**

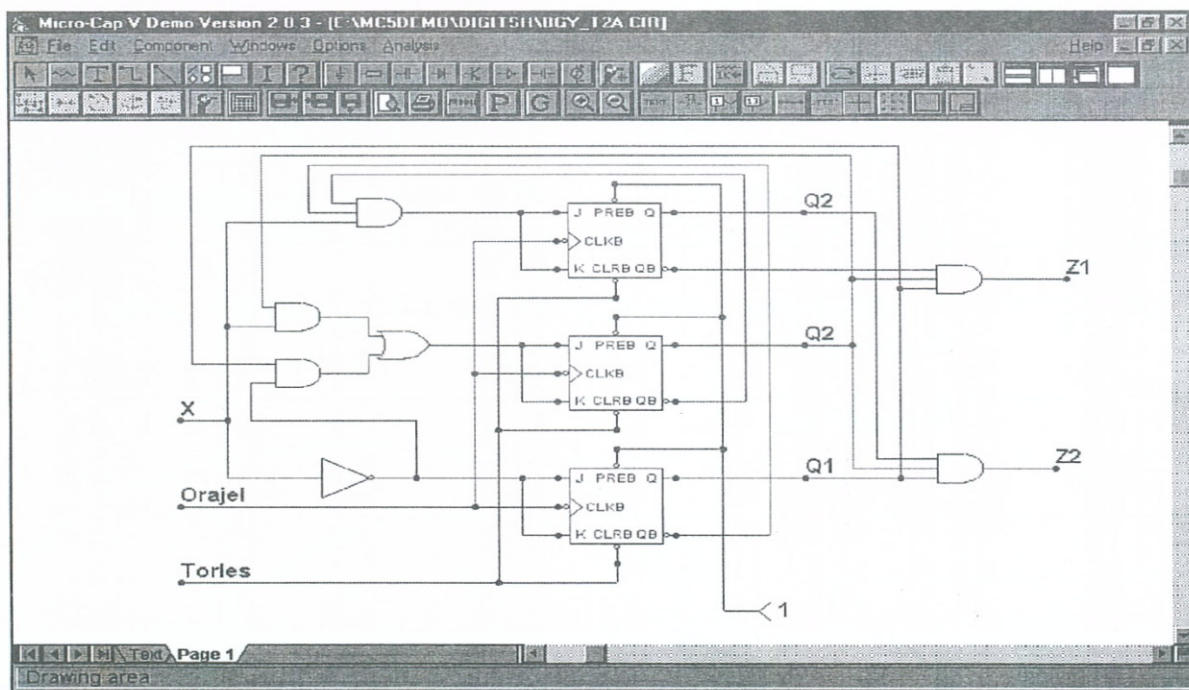
**Tervezze meg** a kimeneti kombinációs hálózatot is! **Írja fel** a kimeneti kombinációs hálózat logikai egyenleteit!

**Megoldás:**

$Z_2 = Q_3 \cdot Q_2 \cdot Q_1$  és  $Z_1 = \bar{Q}_3 \cdot \bar{Q}_2 \cdot \bar{Q}_1$

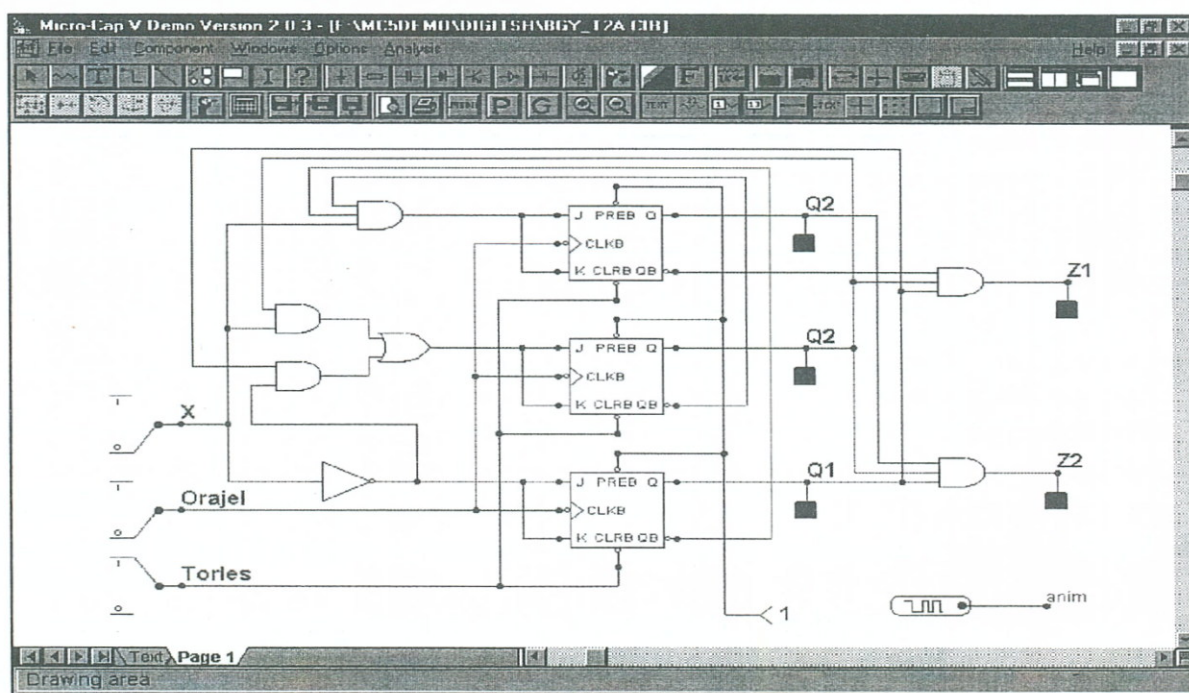
## 5. lépés

**Készítse el a kapcsolási rajzot! Válassza ki az alkatrészeket (mivel nincs előírás a felhasználandó alkatrészek típusára) a *Component*, *Digital Primitives*, *Standard Gates* és *Component*, *Digital Primitives*, *Standard Gates*, *Edge-Triggered Flip-Flops* menükből. A flip-flopok között nem található T flip-flop, de ez (mint ismeretes) igen könnyen előállítható a JK flip-flopból, a J és a K bemenet összekötésével.**



## 6. lépés

**Egészítse ki a kapcsolást az animációs vizsgálatot lehetővé tevő kapcsolókkal, LED-del, valamint az animációhoz szükséges stimulus generátorral!**



**7. lépés:** Ellenőrizze a hálózat működését az állapot-táblázat alapján!

**Vizsgálati eredmény:**

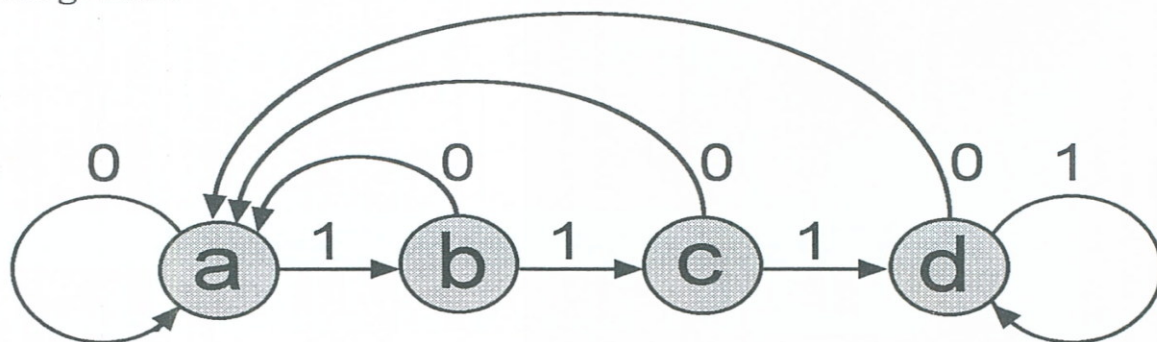
A megtervezett és összeállított hálózat az állapot-táblázattal megadott specifikációnak megfelelően működik.

**2.3.2 Tervezés kapuk és flip-flopok felhasználásával, a feladat szöveges megfogalmazása alapján**

**Feladat:** Tervezze meg azt a szinkron sorrendi hálózatot, amelyik figyeli az X bemenetére érkező jelet, és ha úgy találja, hogy a három legutolsó órajel-impulzus idején 1 értékű volt a bemeneti jel, 1 értékű kimeneti jelet ad mindaddig, amíg ez az állapot fennáll! A tervezéshez használjon 74HC sorozatú CMOS IC-eket!

**1. lépés** A szöveges specifikáció alapján **rajzolja meg** a hálózat állapot-diagramját!

**Megoldás:**



**2. lépés** **Rajzolja meg** az állapot-táblázatot, amely a továbbiakban a tervezés alapjául szolgál!

**Megoldás:**

n. ütem	n+1. ütem		Z	Z
	X=0	X=1	X=0	X=1
a	a	b	0	0
b	a	c	0	0
c	a	d	0	0
d	a	d	1	1

**3. lépés** **Válasszon** állapotkódot, **rajzolja meg** a kódolt állapot-táblázatot!

**Megoldás:** Belátható (és szisztematikus módszerrel ellenőrizhető), hogy az állapotok száma nem csökkenthető, ezért a 4 állapot kódolásához 2 db tárolóelemre van szükség.

**Állapotkód:**

Állapot	Kód
a	00
b	01
c	10
d	11

**A kódolt állapot-táblázat:**

n.ütem $Q_2^n Q_1^n$	n+1. ütem		Z	Z
	X=0 $Q_2^{n+1} Q_1^{n+1}$	X=1 $Q_2^{n+1} Q_1^{n+1}$	X=0	X=1
00	00	01	0	0
01	00	10	0	0
10	00	11	0	0
11	00	11	1	1

**4. lépés**

A kódolt állapot- táblázat alapján **rajzolja meg** az állapot-egyenletek egyszerűsítésére szolgáló Karnaugh-táblákat!

	X			
$Q_2^n Q_1^n$		0	1	$Q_2^{n+1}$
00		0	0	←
01		0	1	
11		0	1	
10		0	1	

	X			
$Q_2^n Q_1^n$		0	1	$Q_1^{n+1}$
00		0	1	←
01		0	0	
11		0	1	
10		0	1	

**5. lépés**

**Válassza meg** a felhasználandó tárolóelem típusát!

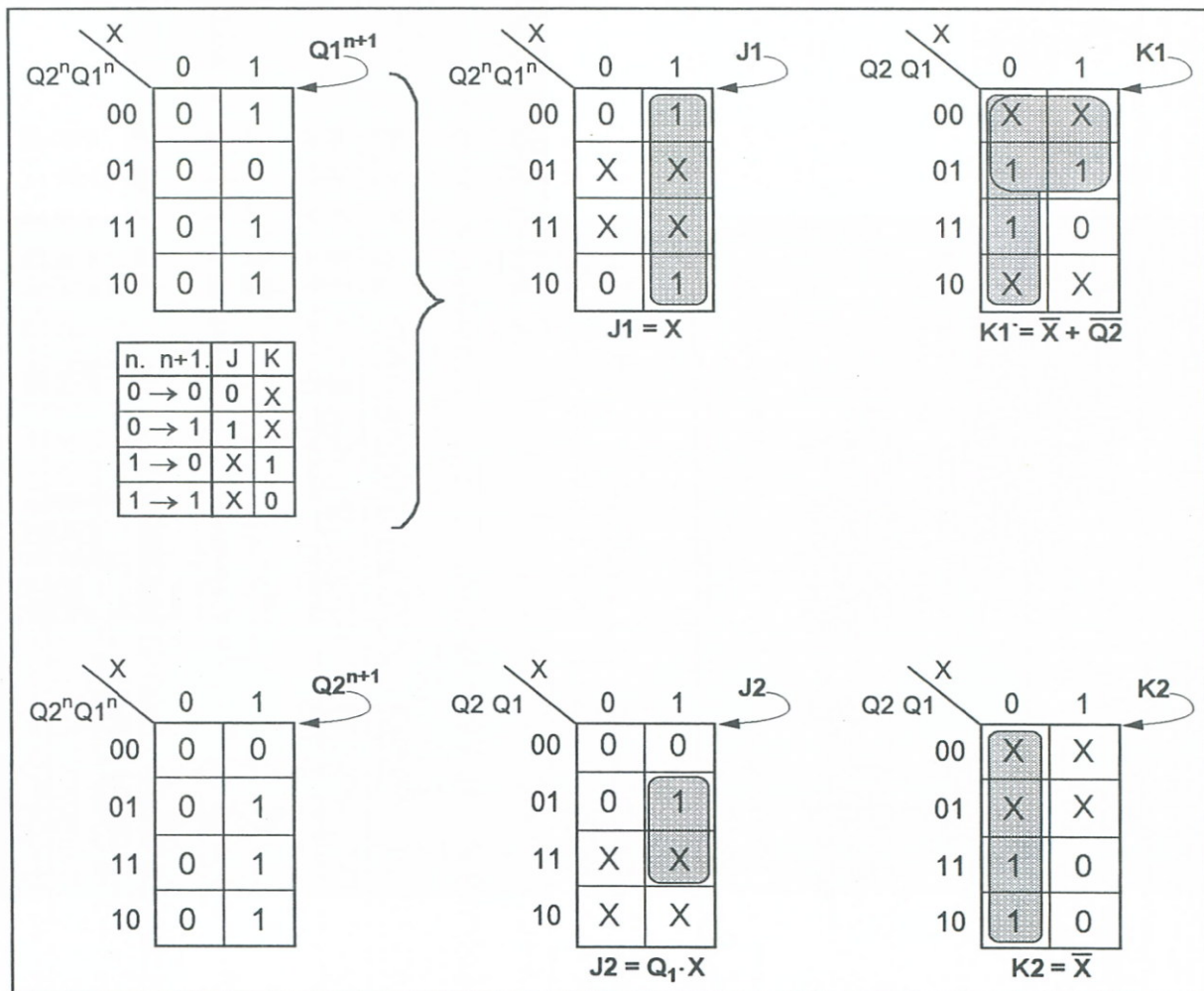
**Megoldás:**

Az alkatrész-könyvtárban a Student Verzió esetében csak a 74HC74 élvezérelt D flip-flop és a 74HC76 JK Master-Slave flip-flop található. Ezek közül a 74HC76 JK Master-Slave flip-flop választása célszerű.

**6. lépés**

A flip-flop működési táblázata és az állapot-egyenletek alapján **írja fel** a bemeneti egyenleteket! Az egyenletek egyszerűsített alakjának meghatározását a legkönnyebben grafikus módszerrel (Karnaugh-táblával) végezheti el.

**Megoldás:**



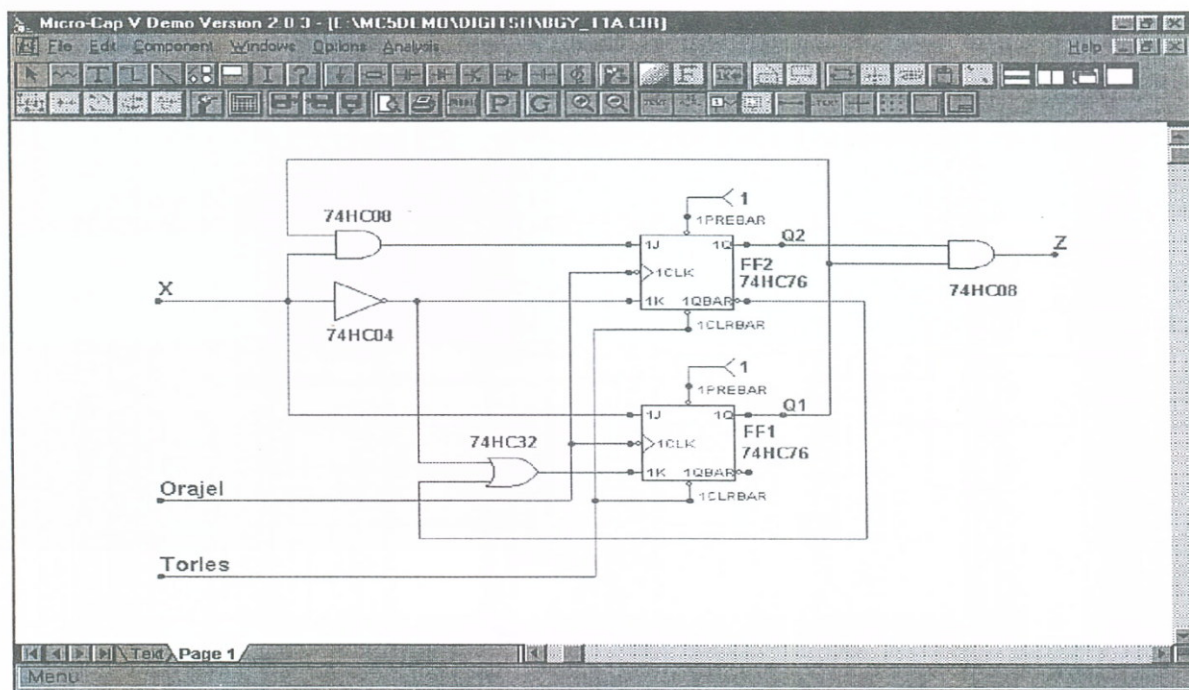
**7. lépés**

Az állapot-táblázat alapján **írja fel** a kimeneti kombinációs hálózat működését leíró egyenletet!

**Megoldás:**  $Z = Q_2 \cdot Q_1$

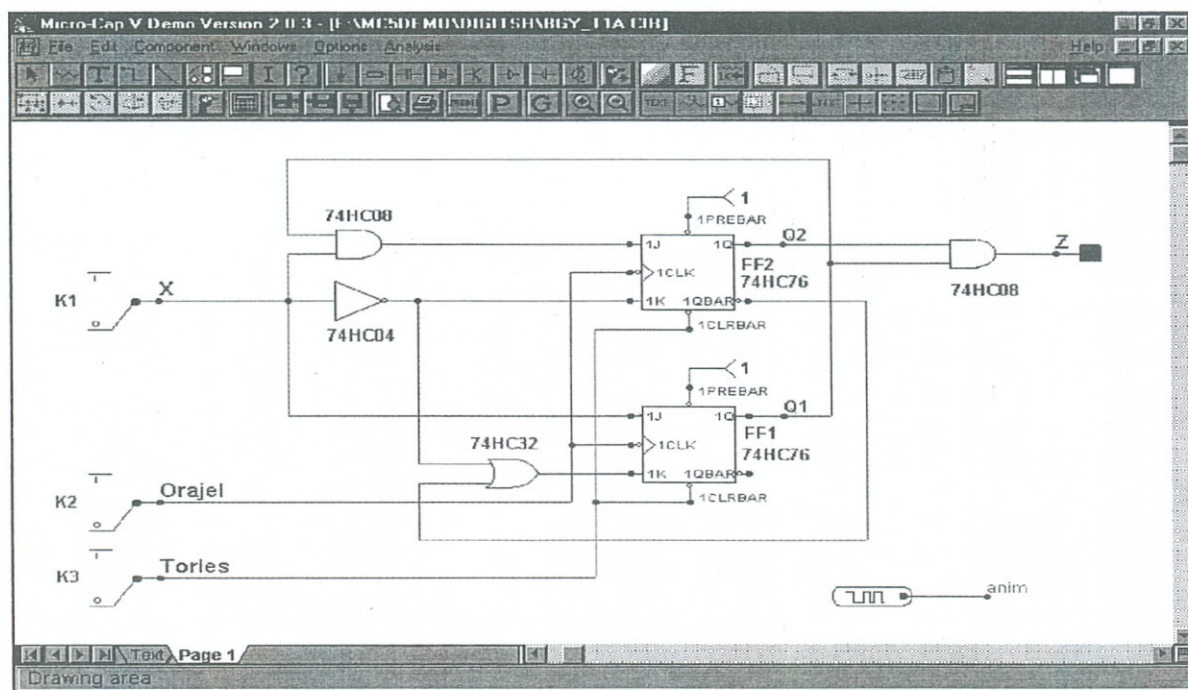
**8. lépés**

Rajzolja meg a kapcsolási rajzot!



## 9. lépés

Egészítse ki a kapcsolást az animációs vizsgálatot lehetővé tévő kapcsolókkal, LED-del, valamint az animációhoz szükséges stimulus generátorral!



## 10. lépés

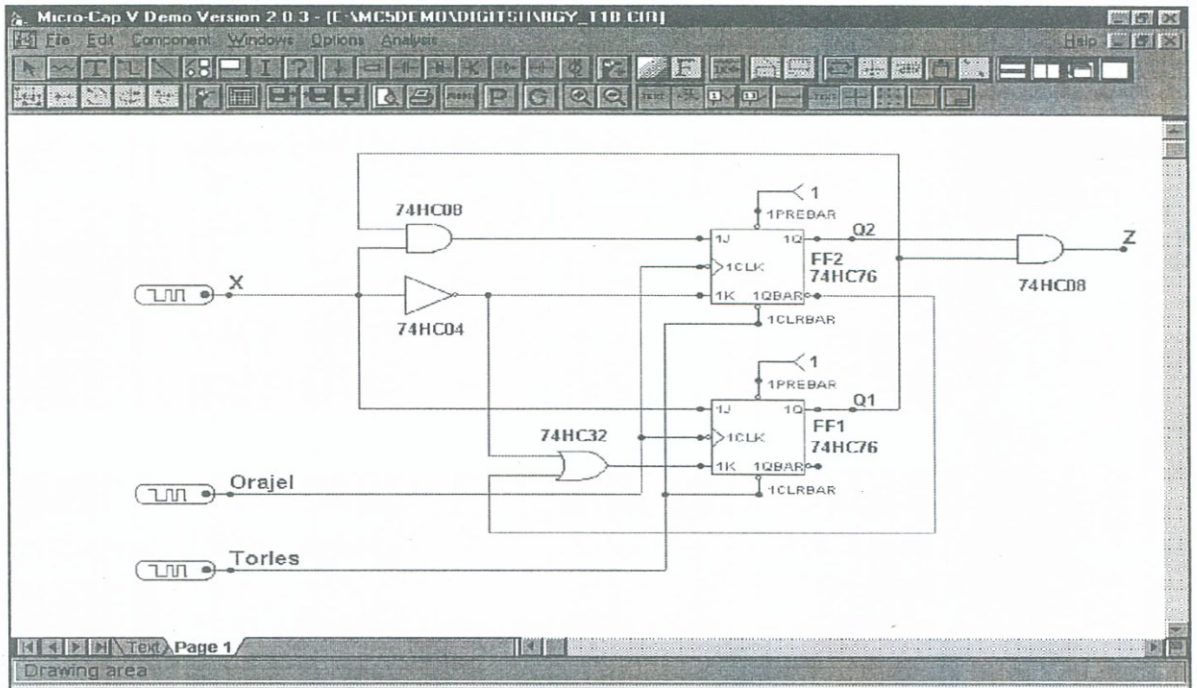
Válassza ki a menüből a tranziens analízist, állítsa be a szimulációs paramétereket a megismert módon. **Kapcsolja be** az animációs módot, válassza meg az animációs paramétereket a javasolt módon! **Ellenőrizze** az állapot-táblázat illetve az állapot-diagram alapján a kapcsolás működését!

## 11. lépés

**Ellenőrizze** a működést a bemeneti és kimeneti jelek időfüggvényei alapján! Ehhez a vizsgálathoz **alakítsa át** a kapcsolási rajzot, törölje ki az animációhoz szükséges elemeket, kapcsoljon a bemenetekre stimulus



generátorokat! Például: a *Törlés* jel legyen egy 500ns ideig tartó negatív logikájú impulzus, az Órajel periódusideje legyen 1us, kitöltési tényezője 50%, az X jel mindig az órajel 0 szintje alatt váltson értéket, az órajel 1 szintje esetén mérhető értékei pedig a 0010110111011110 bitsorozat szerint változzanak!



```

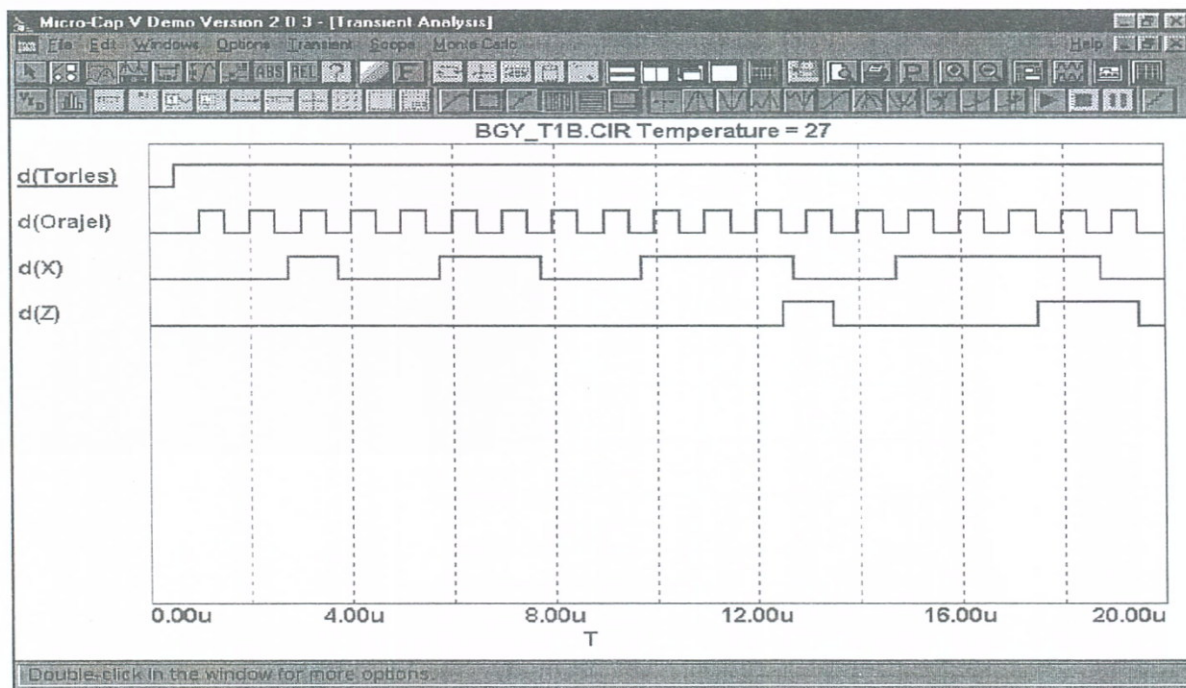
Micro-Cap V Demo Version 2.0.3 - [E:\MC5DEMO\DIGITS\INDGY_T10.CIT]
File Edit Component Windows Options Analysis Help
*****
.DEFINE ora
+0us 0
+LABEL=orastart
+1us 1
+1.5us 0
+2us GOTO orastart -1 TIMES
*****
.DEFINE alapall
+0us 0
+0.5us 1
*****
.DEFINE Xbe
+0us 0
+2.75us 1
+3.75us 0
+5.75us 1
+7.75us 0
+9.75us 1
+12.75us 0
+14.75us 1
+18.75us 0
*****
Text / Page 1 /
Text area

```

## 12. lépés

Válassza ki a menüből a tranziens analízist, állítsa a szimulációs idejét (Time Range) az X jelet szolgáltató stimulus generátor időzítése alapján 20us-ra! **Ne feledje el kikapcsolni az animációs módot!** A tranziens analízis párbeszéd-ablakban az *Y Expression* dobozokba írja be a

kirajzoltatni kívánt jelek nevét: (Törlés, Órajel, X, Z). Indítsa el a szimulációt!



**Vizsgálati eredmények:** Az ábrából láthatóan a kapcsolás a kívánalmaknak megfelelően működik, akkor és csak akkor ad 1 értékű kimeneti jelet, ha a három utolsó órajel-impulzus idején 1 értékű volt a bemeneti jel.

### 2.3.3 Tervezés programozható kombinációs hálózat és flip-flopok felhasználásával

**Feladat:** Tervezze meg az előző fejezetben kapukkal és flip-flopokkal megépített hálózatot programozható kombinációs hálózat és flip-flopok segítségével! Programozható kombinációs hálózatként válasszon PROM-ot! Használja fel az előző fejezet eredményeit a tervezéshez!

#### 1. lépés

**Induljon ki a kódolt állapot-táblázatból! Határozza meg a szükséges tárolóelemek számát és típusát!**

**Megoldás:** A tárolóelemek számát a kód már meghatározza. Az előző fejezetben leírtak itt is érvényesek, tehát 2 tárolóelem szükséges. PROM alkalmazása esetén a D típusú tárolóelem választása a legelőnyösebb, így a PROM listát egyszerűen a kódolt állapot-táblázatból lehet létrehozni.

#### 2. lépés

**Határozza meg a feladat megoldásához szükséges PROM méretét, azaz hány bemenetű és hány kimenetű PROM -ra van szükség! (Másként fogalmazva: mekkora tárolókapacitású PROM kell.)**

**Megoldás:** A szükséges bemenetek számát az határozza meg, hogy hálózatnak hány bemeneti jele van és hány tárolóelem van a hálózatban. E két érték összege adja a PROM szükséges bemeneteinek számát.

A szükséges kimeneti jelek számát a tárolóelemek számának és a kimeneti jelek számának összege határozza meg. Jelen feladatnál a PROM bemeneteinek száma:  $1+2=3$ , míg a kimenetek száma  $2+1=3$ .

### 3. lépés

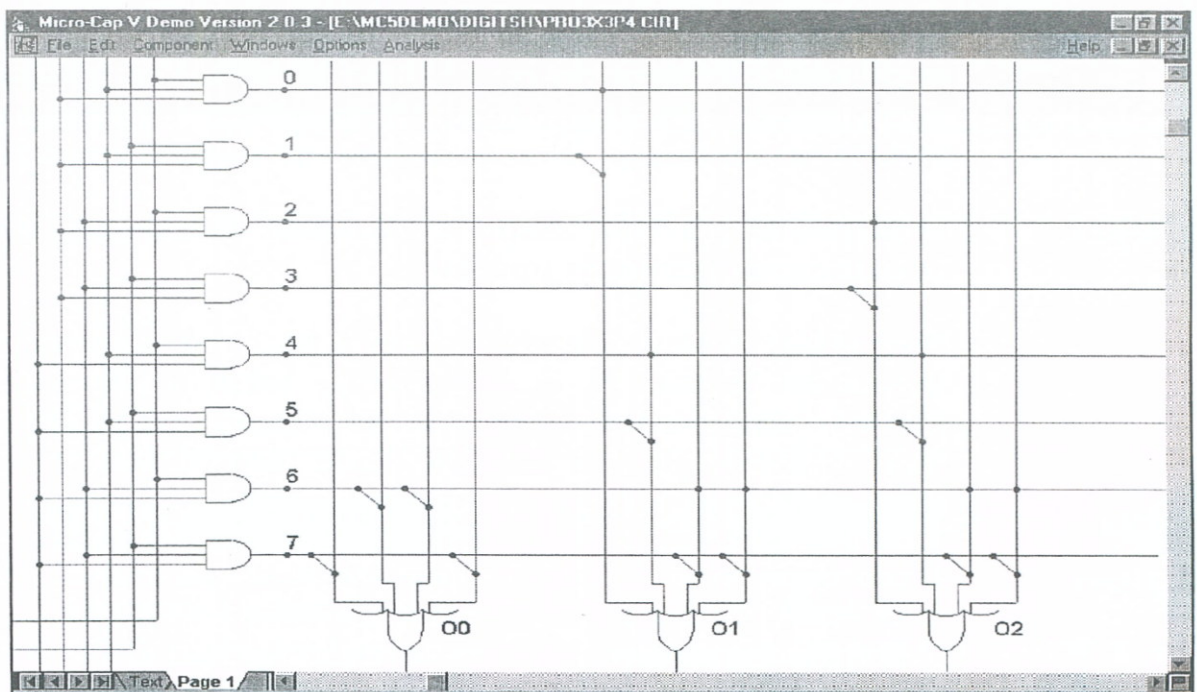
**Készítse el a PROM listát!**

**Megoldás:**

Cím	Tartalom
$Q_2Q_1X$	$D_2D_1Z$
000	000
001	010
010	000
011	100
100	000
101	110
110	001
111	111

### 4. lépés

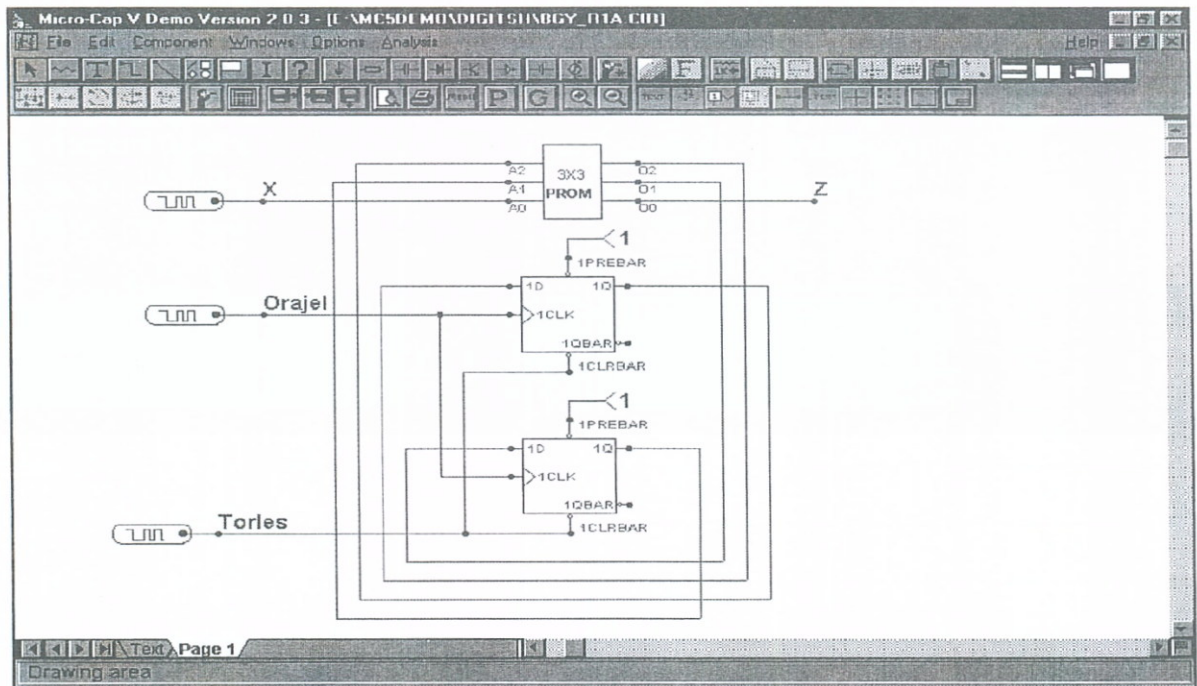
Programozza be a PROM-ot! Válassza ki a *File, Megnyitás* menüpontot. Keressen az aktuális könyvtárban található állományok közül egy beprogramozatlan, üres 3x3-as PROM -ot tartalmazó állományt (PRO3X3P1, PRO3X3P2, PRO3X3P3 vagy PRO3X3P4, Pl. legyen ez most a PRO3X3P4 állomány). A korábban már megismert módon végezze el a PROM beprogramozását!



### 5. lépés

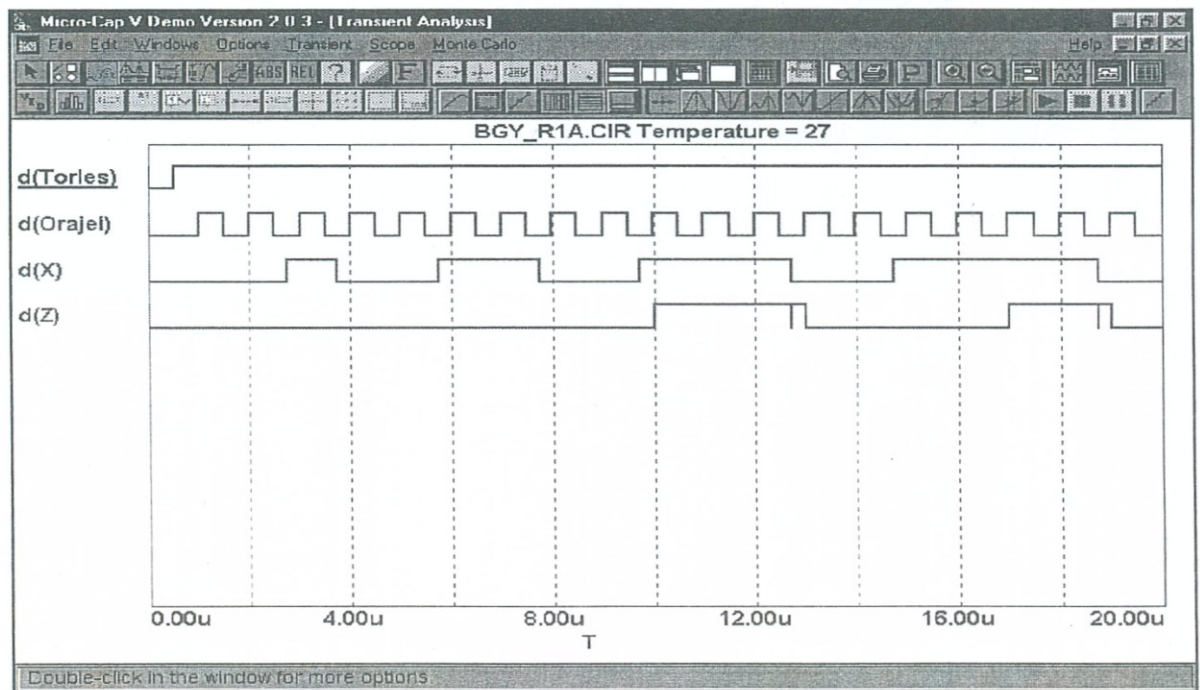
**Rajzolja meg a kapcsolási rajzot!** (A PROM-ot a *Component, Digital Circuits, PROM* menüből kell választani.) Használja ugyanazokat a

stimulus generátorokat, mint amelyeket a kapukból és flip-flopokból álló megoldás vizsgálatánál alkalmazott!



## 6. lépés

Válassza ki a menüből a tranziens analízist, állítsa a szimulációs idejét (Time Range) az X jelet szolgáltató stimulus generátor időzítése alapján 20 $\mu$ s-ra! A tranziens analízis párbeszéd-ablakban az *Y Expression* dobozokba írja be a kirajzoltatni kívánt jelek nevét: (Törlés, Órajel, X, Z). **Indítsa el a szimulációt!**



## 7. lépés

Értékelje a kapott eredményeket!

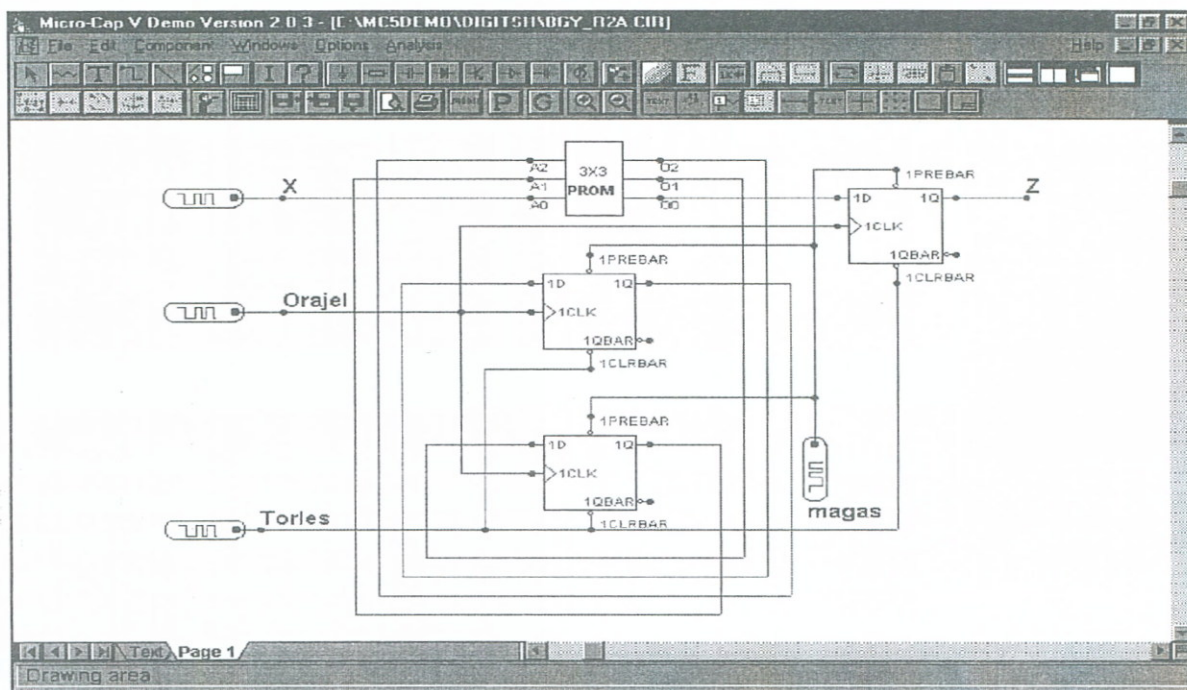
**Vizsgálati eredmény:** Az ábra alapján megállapítható, hogy a kimeneti hálózatrész statikus hazardot tartalmaz, mert az X bemeneti jel

változásának hatására akkor is változik a kimeneti jel, amikor logikai működés alapján nem lenne szabad változnia. A hazard okozója a PROM, mert a negált címjelek belső késleltetés nagyobb:

## 8. lépés

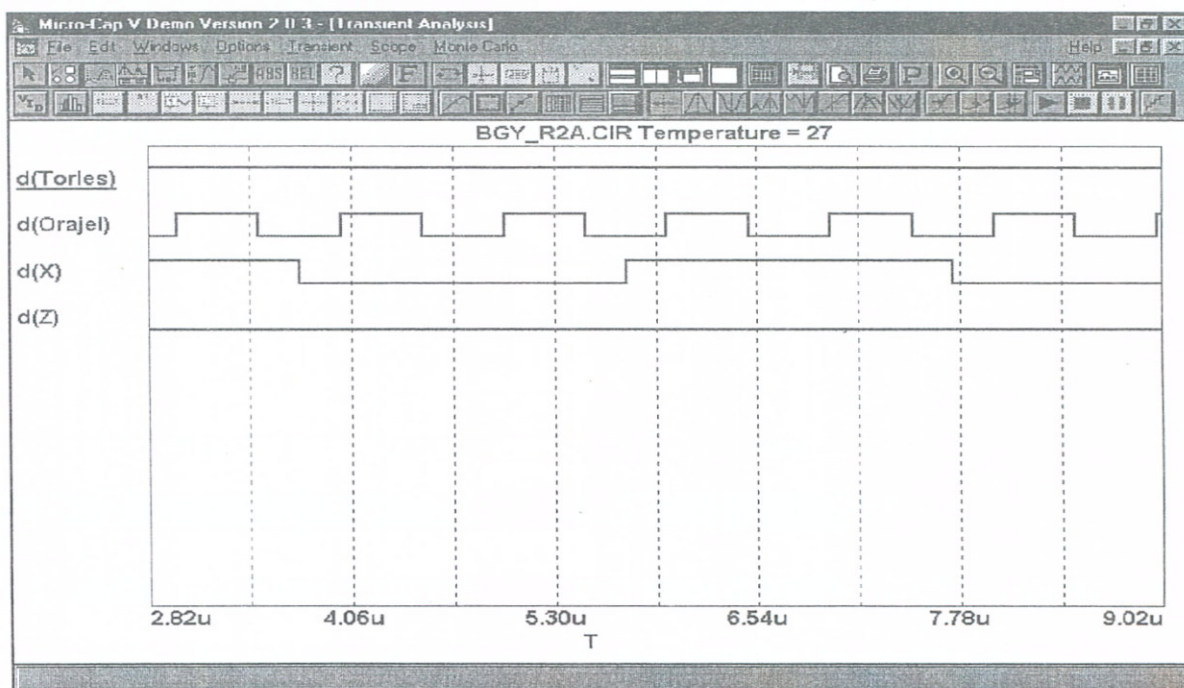
Módosítsa a kapcsolást, szüntesse meg a hazardot!

**Megoldás:** A legegyszerűbb változtatás, ha a kimeneti jelet is egy flip-flop állítja elő (így nem kell módosítani a PROM makró kapcsolását és csak 1 alkatrészsel és 1 csomóponttal bővül a kapcsolás, ami a Student Verzió korlátai miatt fontos):



## 9. lépés

**Indítsa el újra szimulációt, vizsgálja meg a működést, hatásos volt-e a módosítás!**



**Vizsgálati eredmény:** A hálózat most már hibamentesen működik.

## 2.4 Hibakeresés

### 2.4.1 Hibakeresés a megadott állapottáblázat alapján

**Feladat:** Keresse meg, miért nem teljesíti az állapot-táblázat szerinti specifikációt az alábbi hálózat (a kapcsolást a HIBAKER1.CIR állomány tartalmazza)! Tegyen javaslatot a hiba javítására! Végezze el a javítást, ellenőrizze a hálózat helyes működését!

Az állapot-táblázat:

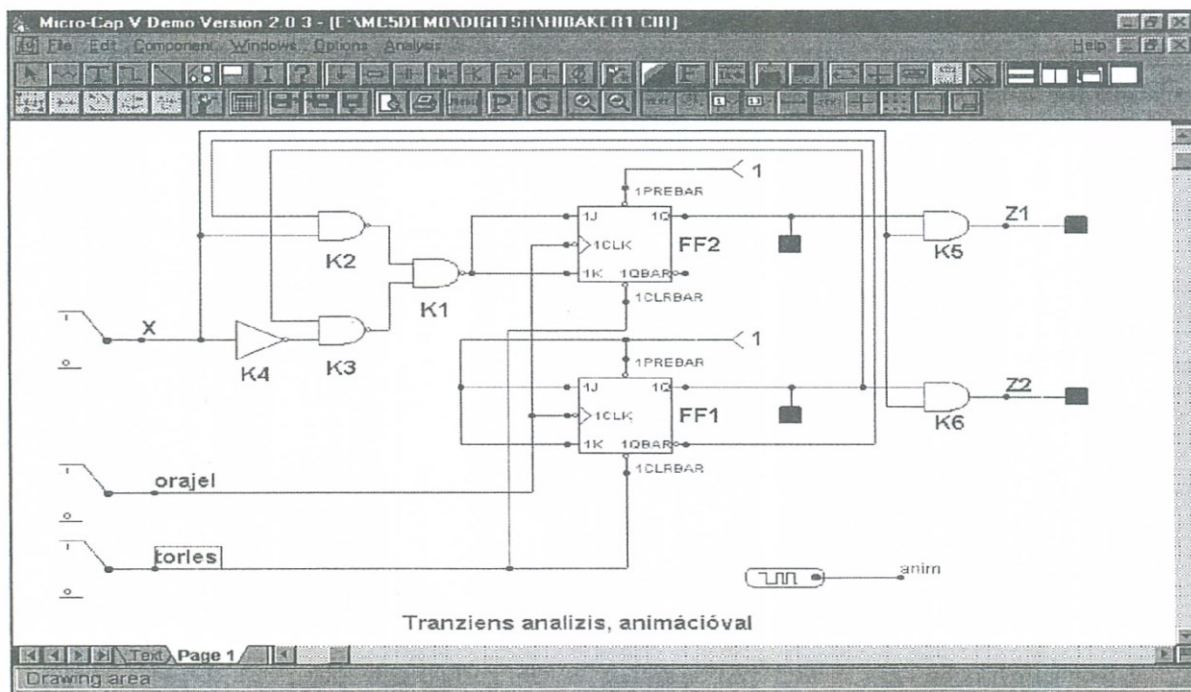
n.ütem	n+1. ütem	n+1. ütem	Z <sub>2</sub> Z <sub>1</sub>	Z <sub>2</sub> Z <sub>1</sub>
	X=0	X=1	X=0	X=1
a	d	b	00	00
b	a	c	00	01
c	b	d	00	10
d	c	a	00	11

Az állapotkód:

a	00
b	01
c	10
d	11

### 1. lépés

Töltse be a HIBAKER1.CIR állományt!



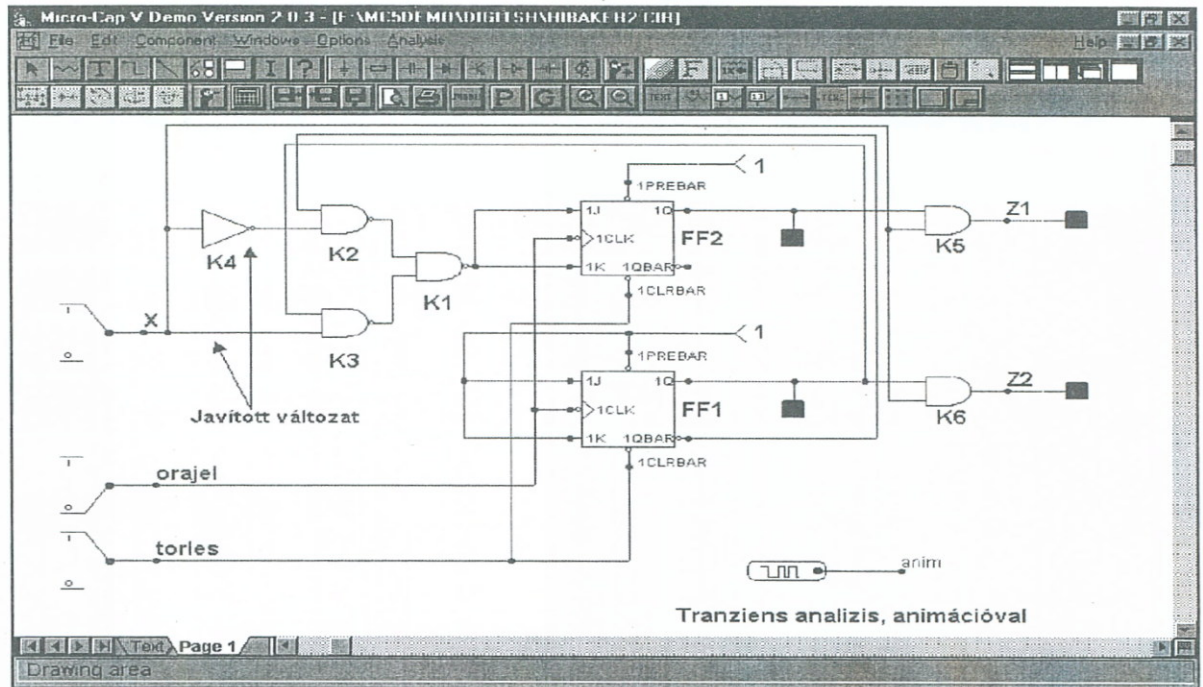
### 2. lépés

Végezze el a szimulációs vizsgálatot!

**3. lépés**      Értékelje a kapott eredményeket!

**Vizsgálati eredmények:** A hibás működés oka az, hogy a kapcsolásban a K3 kapu kapja a -X jelet a K2 kapu helyett. A hiba úgy javítható, ha a K4 inverter kimeneti jelét a K2 kapura vezetjük, míg a K3 kapu közvetlenül az X bemeneti jelet kapja.

**4. lépés**      Végezze el a kapcsolásban a javításokat!



**5. lépés**      Vizsgálja meg újra a kapcsolást, értékelje a kapott eredményeket!

**Vizsgálati eredmények:**

A javítás sikerrel járt, a hálózat most már az állapot-táblázatnak megfelelően működik.