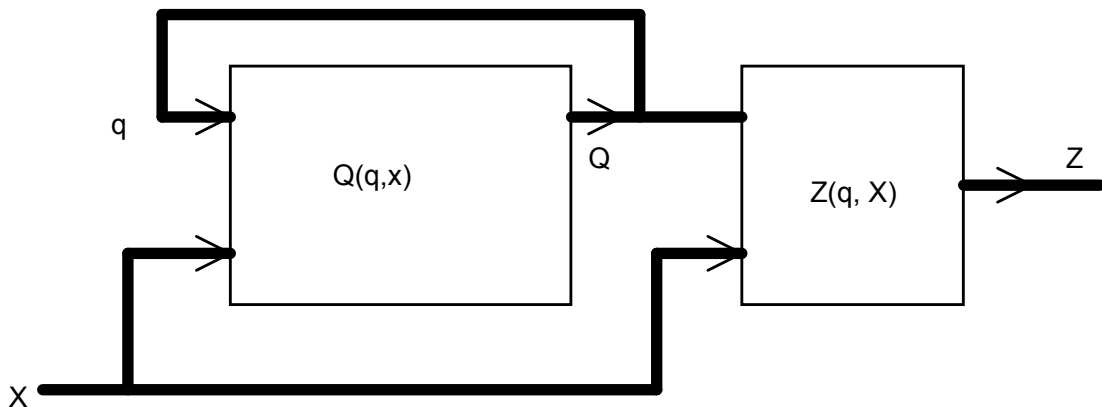


Aszinkron sorrendi hálózatok

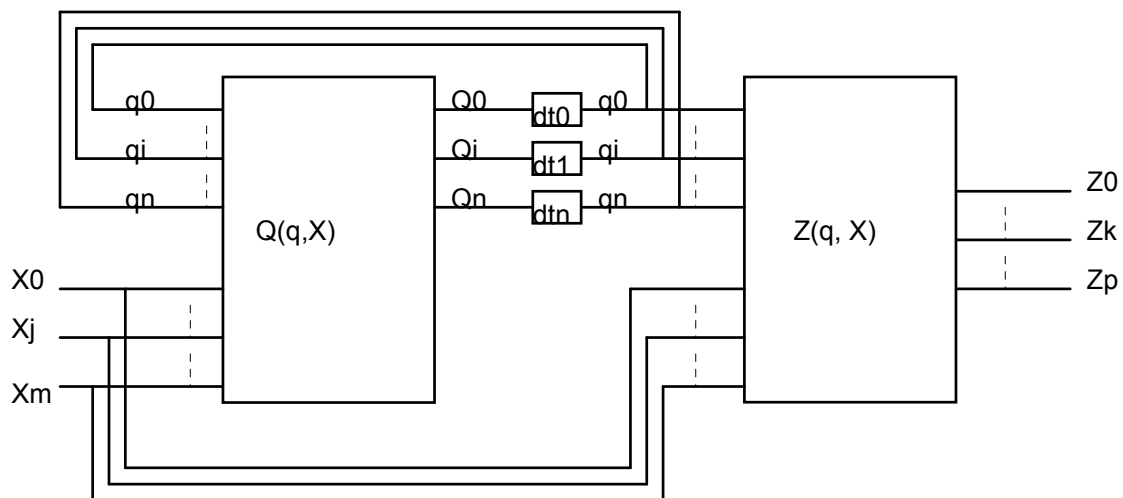
© Benesóczky Zoltán 2004

A jegyzetet a szerzői jog védi. Azt a BME hallgatói használhatják, nyomtathatják tanulás céljából. Minden egyéb felhasználáshoz a szerző beleegyezése szükséges.

**Az aszinkron hálózat struktúrája:
visszacsatolt kombinációs hálózat**



A működés egyszerűsített modellje



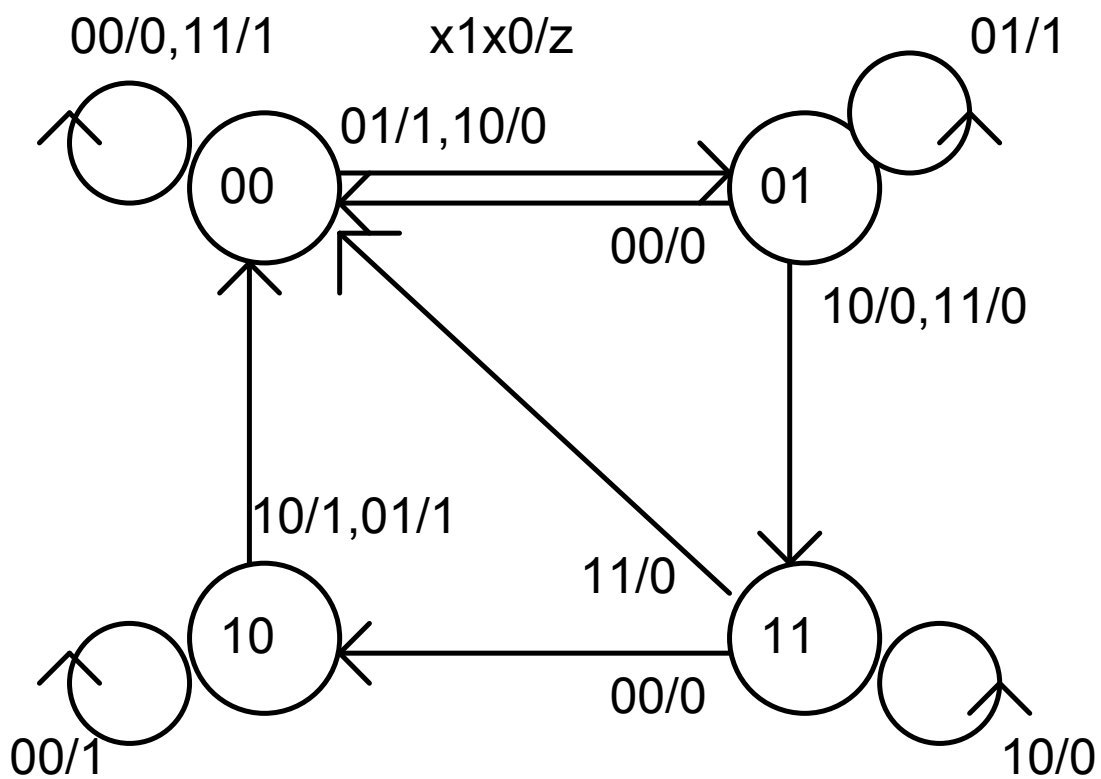
Egy állapot stabil valamely X bemeneti kombináció esetén, ha $Q(q, X) = q$

Egy állapot instabil valamely **X** bemeneti kombináció esetén, ha $Q(q, X) \neq q$

Példa. Hogyan viselkedik az alábbi aszinkron sorrendi hálózat az

y₁y₀=11 kódú állapotból indulva,

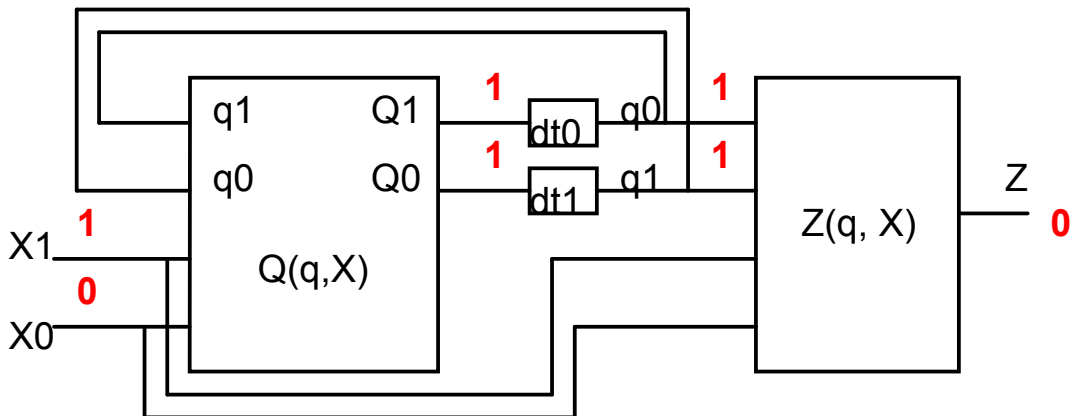
az **X₁X₀= 10, 00, 01** bemeneti sorozat hatására?



q ₁ q ₀ /x ₁ x ₀	00	01	11	10
00	00/0	01/1	00/1	01/0
01	00/0	01/1	11/0	11/0
11	10/0	--/-	00/0	11/0
10	10/1	00/1	--/-	00/1

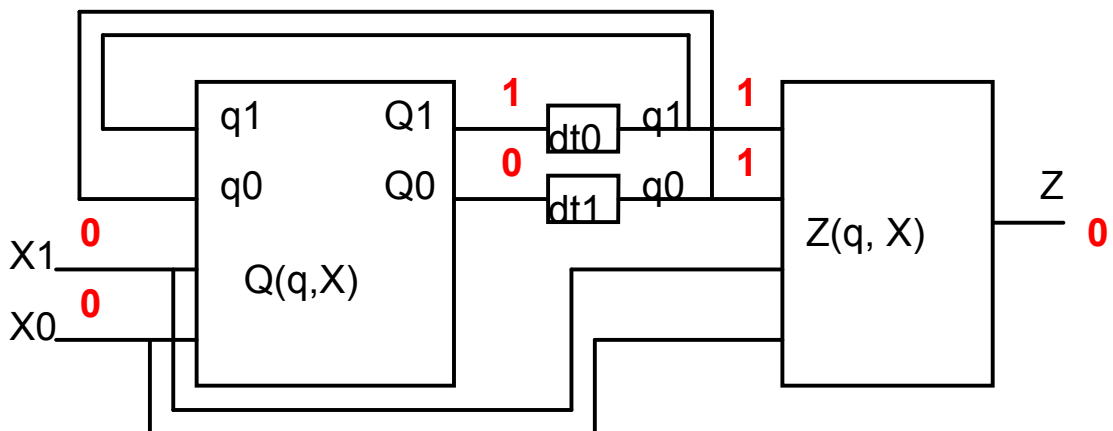
X1X0=10

q1q0/x1x0	00	01	11	10
00	00/0	01/1	00/1	01/0
01	00	01/1	11	11/0
11	10/0	--	00/0	11/0
10	10/1	00/1	--	00/1



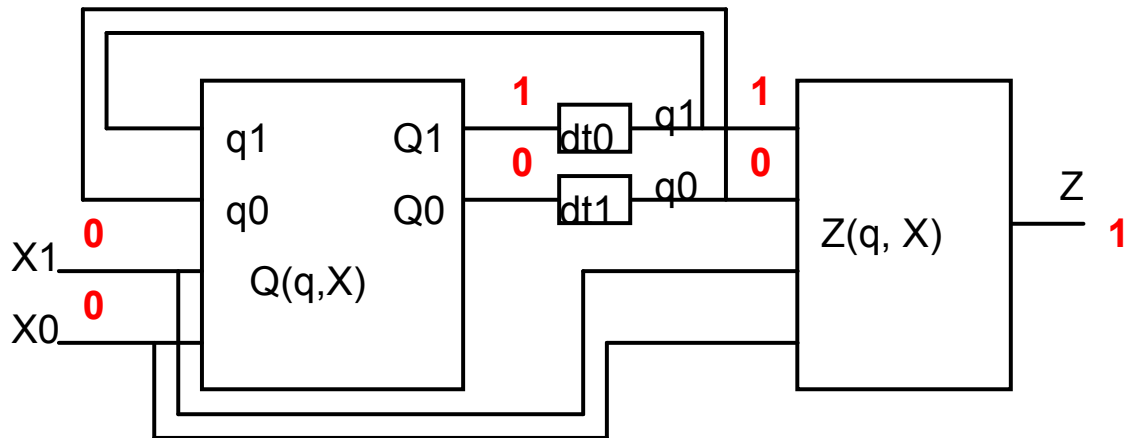
X1X0=00

q1q0/x1x0	00	01	11	10
00	00/0	01/1	00/1	01/0
01	00	01/1	11	11/0
11	10/0	--	00/0	11/0
10	10/1	00/1	--	00/1



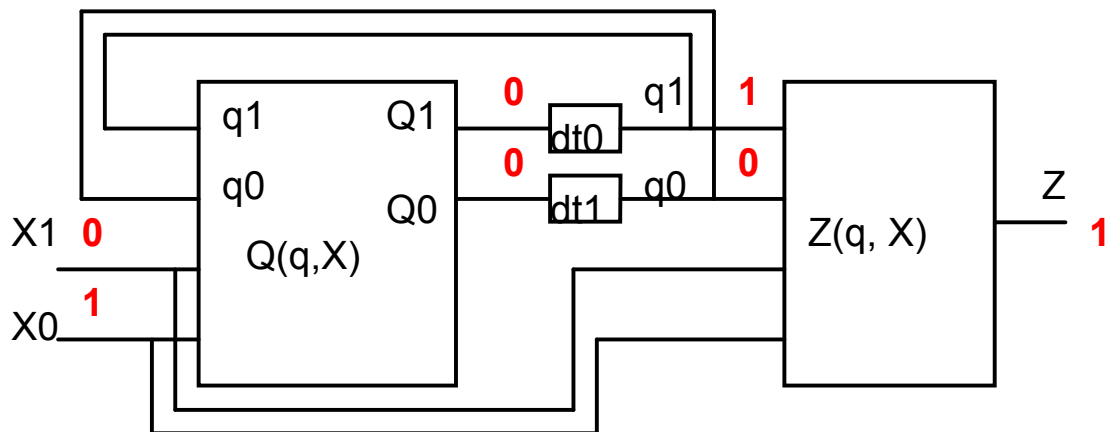
$X_1X_0=00$

q_1q_0/x_1x_0	00	01	11	10
00	00/0	01/1	00/1	01/0
01	00	01/1	11	11/0
11	10/0	--	00/0	11/0
10	10/1	00/1	--	00/1



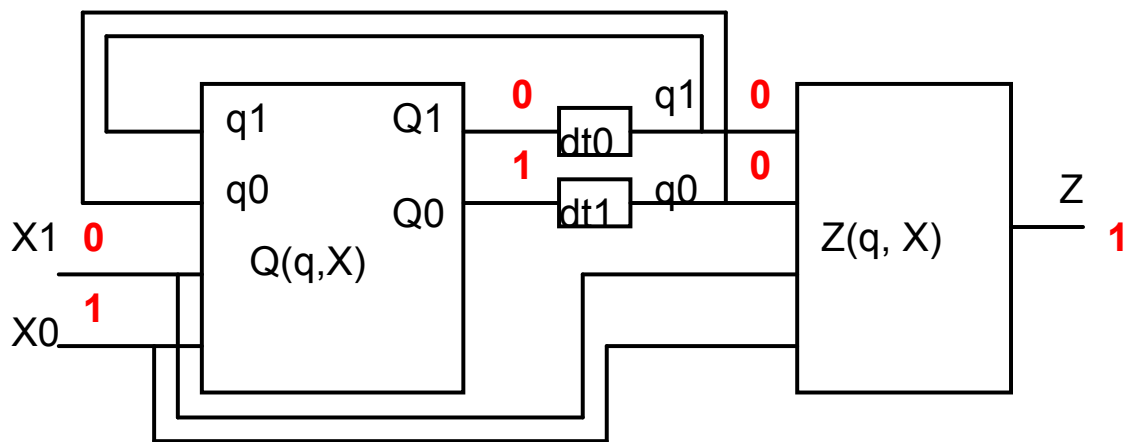
$X_1X_0=01$

q_1q_0/x_1x_0	00	01	11	10
00	00/0	01/1	00/1	01/0
01	00	01/1	11	11/0
11	10/0	--	00/0	11/0
10	10/1	00/1	--	00/1



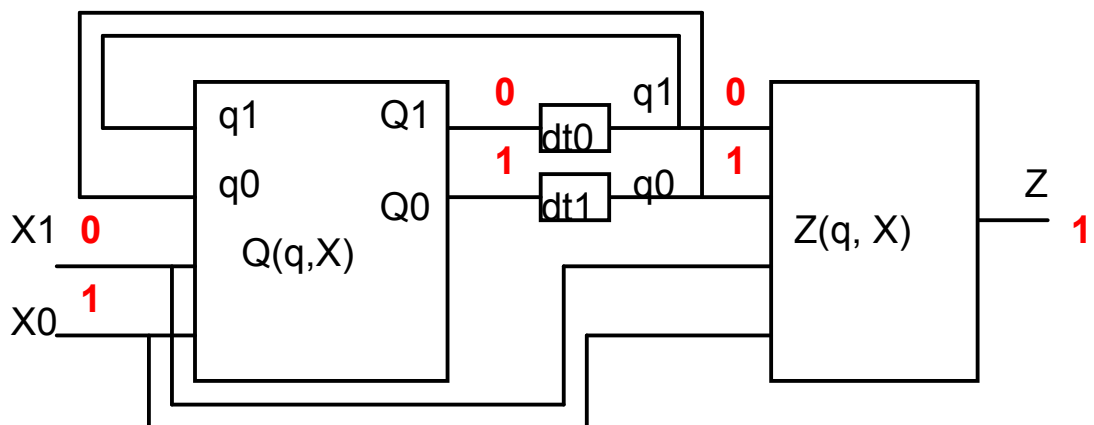
X1X0=01

q1q0/x1x0	00	01	11	10
00	00/0	01/1	00/1	01/0
01	00	01/1	11	11/0
11	10/0	--	00/0	11/0
10	10/1	00/1	--	00/1



X1X0=01

q1q0/x1x0	00	01	11	10
00	00/0	01/1	00/1	01/0
01	00	01/1	11	11/0
11	10/0	--	00/0	11/0
10	10/1	00/1	--	00/1



Összefoglalva:

X1X0	10	00	00	01	01	01
Állapot (q1q0)	11	11	10	10	00	01
Kimenet	0	0	1	1	1	1

Az aszinkron hálózat működtetésének feltételei

FMA (Fundamental Method Asynchronous) feltételek:

- **A bemeneten csak 1 Hamming távolságú változásokat engedünk meg.**
- **Bemeneti változás csak stabil állapotban megengedett.**

Mi történhetne, ha az alábbi hálózatra a $x1x0=11$ bemenet mellett stabil 00 kódú állapotban $x1x0=11$ után 00 -át adnánk? (Nem tartjuk be az FMA feltételt.)

q1q0/x1x0	00	01	11	10
00	00/0	01/1	00/1	10/0
01	00/0	01/1	11/0	11/0
11	10/0	--/-	00/0	11/0
10	10/1	00/1	--/-	11/1

$x1x0=11-01-00$ bemenet érzékelési sorrend esetén:

X1X0	11	01	01	00	00
Állapot	00	00	01	01	00
Kimenet	1	1	1	0	0

$11-10-00$ érzékelési sorrend esetén:

X1X0	11	10	01	01	00	00
Állapot	00	00	10	11	11	10
Kimenet	1	0	1	0	0	1

Több Hamming távolságú bemeneti kombináció változás esetén a bemenet érzékelési sorrendjétől függően hibás választ adhat a hálózat!

Itt feltételeztük, hogy egy-egy változás érzékelése között stabilizálódik a hálózat, s csak utána érzékeli a következő változást. A valóság rosszabb, a hálózat stabilizálódása előtt érzékelhet újabb változást!

További problémák az aszinkron sorrendi hálózatban.

q1q0/x1x0	00	01	11	10
00	00/0	01/1	00/1	01/0
01	00/0	01/1	00/0	11/0
11	10/0	00/1	00/0	11/0
10	10/1	01/1	10/0	00/1

Mi történhetne, ha az alábbi hálózatra a $x1x0=10$ bemenet mellett stabil 11 kódú állapotban $x1x0=10$ után 11-et adunk?

Ez 1 Hamming távolságú változás, betartjuk az FMA feltételt.

Az 11 kódú állapot után a 00 kódú következne, ehelyett a hálózat először 01-et vagy 10-át érzékel.

$q1q0=01$ érzékelése esetén:

X1X0	10	11	11	11
Állapot (qt)	11	11	01!	00
Kimenet	0	0	0	1

$q1q0=01$ érzékelése esetén:

X1X0	10	11	11
Állapot (qt)	11	11	10!
Kimenet	0	0	0

Hibás működés!

Ha az egymást követő állapotok kódjának Hamming távolsága 1-nél nagyobb, akkor **a szekunder változók érzékelési sorrendjétől függően hibásan működhet a hálózat.**

Versenyhelyzet van a szekunder változók között.

A versenyhelyzet kritikus, ha hibás működéshez vezet.

Az alábbi állapottáblán az $x_1x_0=00$ bemenetre stabil 10 állapotból indulva, 01 bemenetre a 2 Hamming távolságú 01 állapot következne, azonban ez nem vezet hibás működéshez, **a versenyhelyzet nem kritikus.** A hálózat előbb-utóbb a kívánt végállapotba kerül.

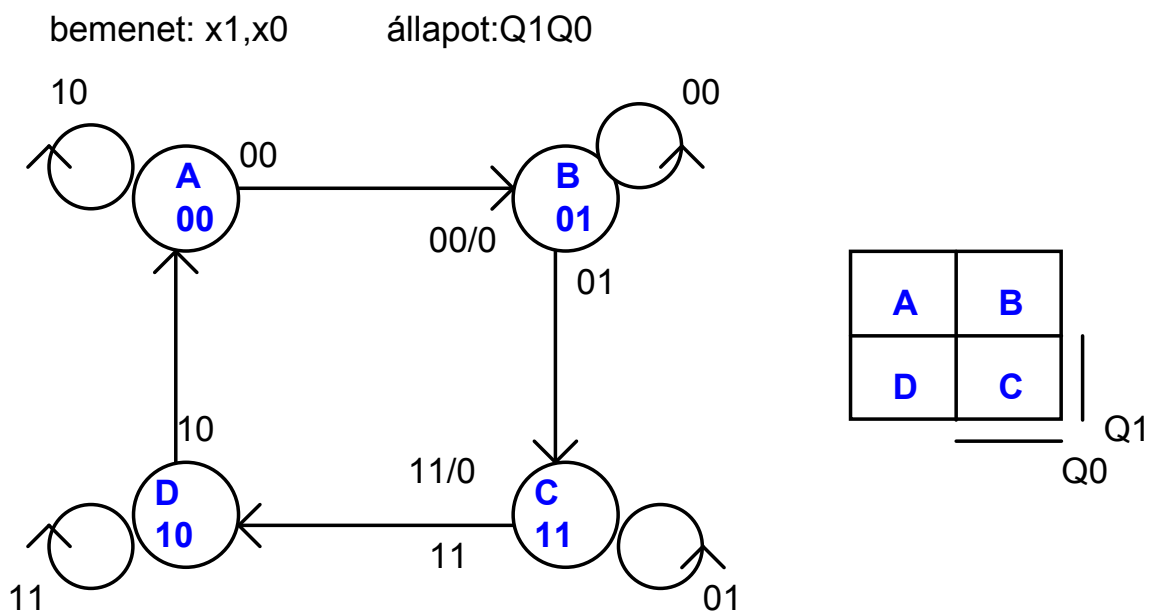
q_1q_0/x_1x_0	00	01	11	10
00	00/0	01/1	00/1	01/0
01	00/0	01/1	00/0	11/0
11	10/0	00/1	00/0	11/0
10	10/1	01/1	10/0	00/1

Az aszinkron hálózat nem tartalmazhat kritikus versenyhelyzetet! **A kritikus versenyhelyzetet megfelelő állapotkódolással lehet kivédeni.** (Lásd később.)

Az aszinkron hálózat állapotkódolásánál az elsődleges cél, a hálózat helyes működése. A helyes működés egyik feltétele, hogy a hálózat ne tartalmazzon kritikus versenyhelyzetet. Ezt többféleképpen elérhetjük:

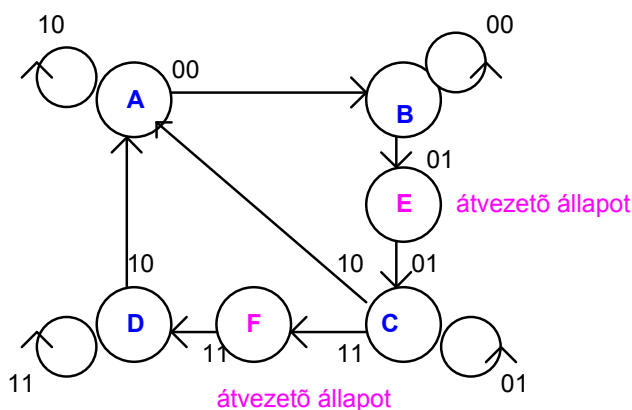
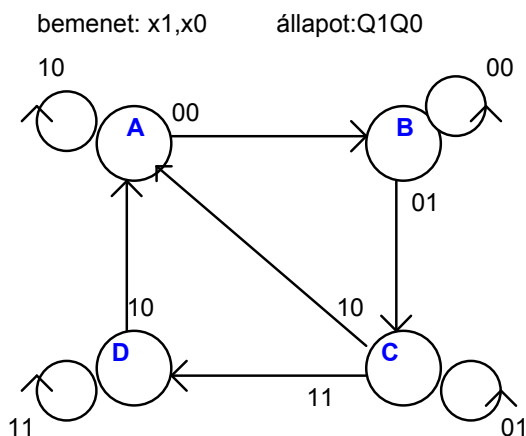
Versenyhelyzet mentes kódolás (szomszédos kódolás)

Ennél a módszernél az egymást követő állapotokat szomszédosan kódoljuk, vagyis az egymást követő állapotok kódja 1 Hamming távolságú. (Ez a kódolás nem összetévesztendő a szinkron hálózatok azonos nevű kódolási módszerével, melynek algoritmus a más!)



Átvezető állapotok felvétele

A szomszédos kódolás ill. a szomszédossági követelmények teljesítése vagy lehetséges közvetlenül, a minimális számú szekunder változóval, vagy sem. Ha nem teljesíthető, akkor az állapotgátfot kiegészítjük instabil állapotokkal úgy, hogy azok egyrészt nem változtatják meg a lényegi működést (hogy adott bemenetre hol stabilizálódik a hálózat), másrészt a kiegészítés után már szomszédosan kódolható lesz a hálózat.

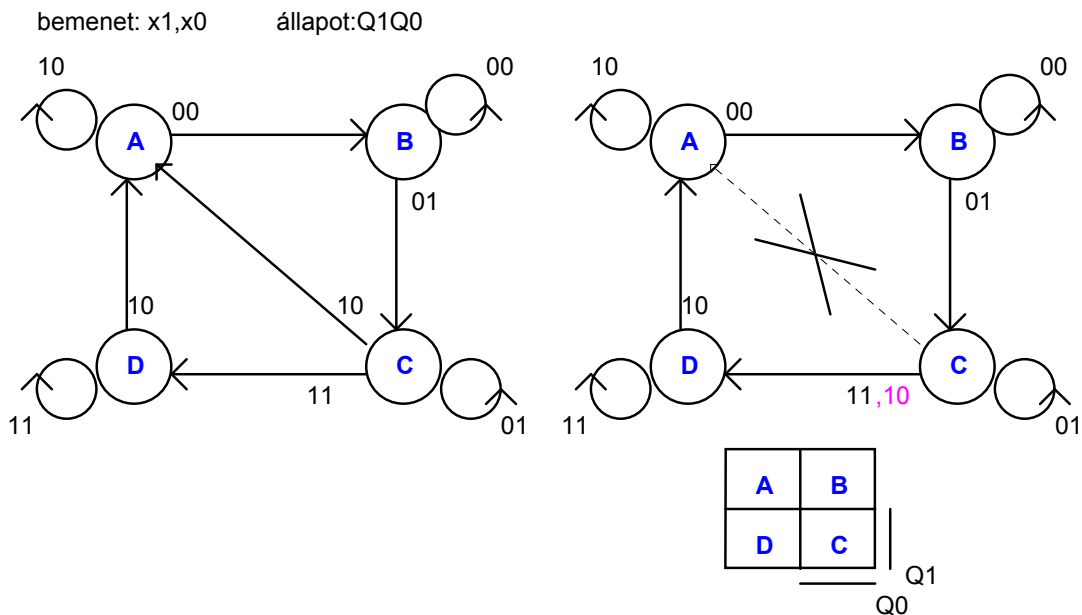


A	D
B	
E	
C	F

$\begin{array}{l} \text{Q2} \\ \text{Q1} \\ \text{Q0} \end{array}$

Instabil állapotok módosítása

Az előbbi módszer a minimálisnál több állapothoz vezethet. Ha az állapottábla tulajdonságai megengedik, akkor más lehetőség is adódhat a versenyhelyzet mentes kódolásra. Azokat az állapotátmenteket, amelyek közvetlenül vezetnek egyik állapotból egy másikba megpróbálhatjuk más állapotokon keresztül vezetni. Ha egy x_i bementre mondjuk egy Q_1 stabil állapotból a Q_2 stabilba kellene jutni, de van olyan Q_j állapot, melyre vagy nincs előírva, hogy hová menjen az x_i bementre, vagy az x_i -re szintén a Q_2 -be vezet, akkor a közvetlen Q_1 - Q_2 átmenet helyett a Q_1 - Q_j - Q_2 átmentet valósítjuk meg, ha az állapottáblát úgy módosítjuk, hogy a Q_1 -ből x_i -re Q_j -be vezetjük az hálózatot, Q_j -ből pedig Q_2 -be. Az ilyen a módosítások nem okoznak változást a lényegi működésben, viszont a szomszédossági követelmények számát csökkenthetik, így kis szerencsével a minimális számú szekunder változóval szomszédosan kódolhatóvá válhat a hálózat.

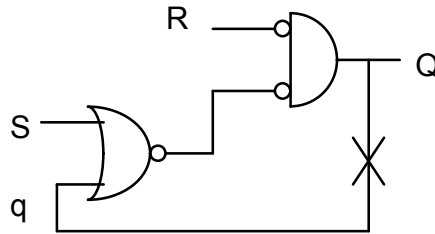


Versenyhelyzetet megengedő kódolás

Az állapottábla alapján megnézzük, hogy mely szomszédossági követelményeket nem szükséges betartani. Nyilván azokat, amelyek be nem tartása versenyhelyzetet ugyan okoz, de az nem kritikus. Ha ezzel sikerül lecsökkenteni a követelmények számát annyira, ami már teljesíthető a minimális számú szekunder változóval, akkor ez is megfelelő kódoláshoz vezet.

Aszinkron sorrendi hálózat analízise

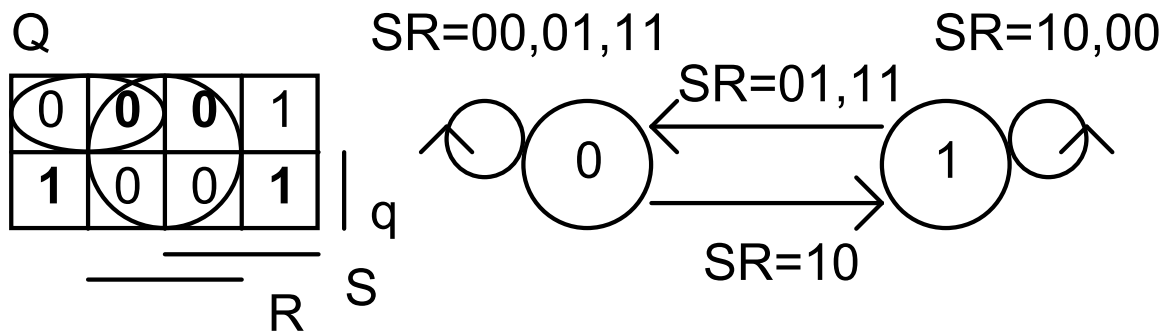
Analizáljuk az alábbi kapcsolási rajzzal megadott ASSH-ot!



Az analízishez először megszakítjuk a visszacsatolás(okat) majd felírjuk a szekunder változó(k) függvényét.

$$Q = \overline{R}(S + q)$$

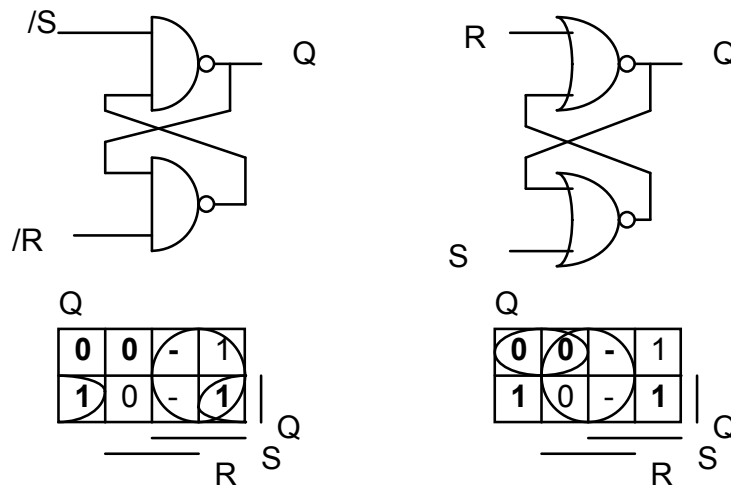
Ez alapján kitöltjük a kódolt állapotábrát. Majd felrajzoljuk az állapotgráfot.



A kapcsolat az **SR flip-flop**ot realizálja.

Aszinkron flip-flopok

A szinkron hálózatokhoz hasonlóan az aszinkron hálózatok is megvalósíthatók flip-flopok segítségével. Az SR flip-flop eredeti állapotáblája és szokásos megvalósításai:



Működése szövegesen:

SR=10 1-be ír (Q=1)

SR=01 0-ba ír (Q=0)

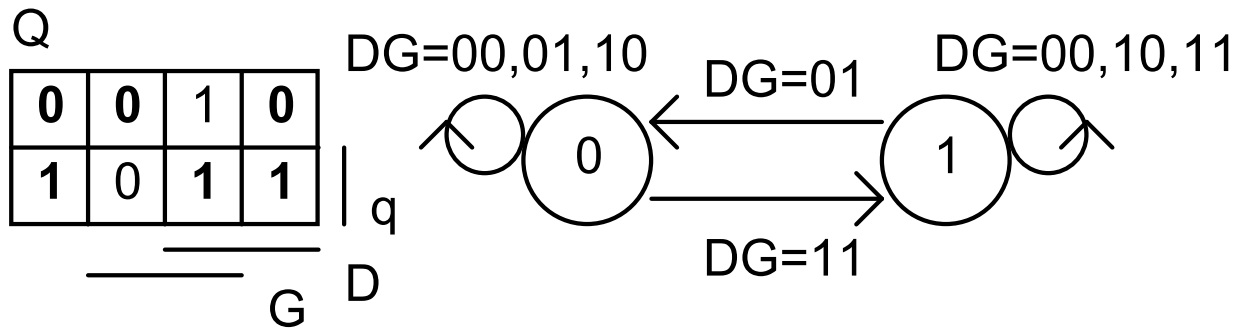
SR=00 megtartja állapotát (Q=q)

A szekunder változók igazság táblái alapján a szinkron hálózatokhoz hasonlóan a vezérlési tábla kitöltéséhez itt is megadható egy segéd táblázat:

qQ	SR	/S/R
00	0-	1-
01	10	01
10	01	10
11	-0	-1

DG flip-flop (latch)

A DG flip-flop Karnaugh táblája és állapotgráfja:



A DG flip-flop működése szövegesen:

$G=1$ átlátszó ($Q=D$)

$G=0$ megőrzi a $G=0$ előtti állapotát

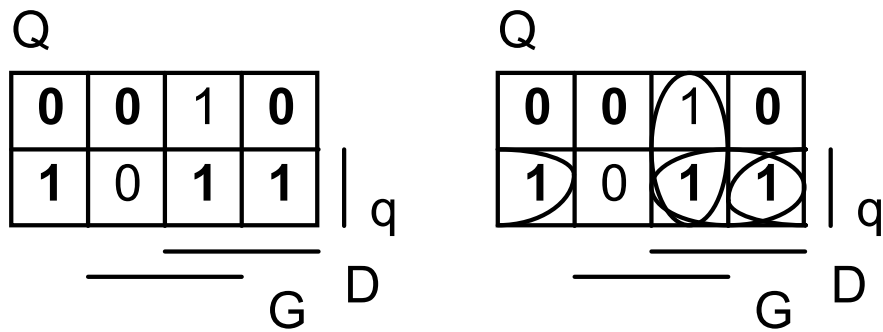
A segédtáblázat a vezérlési függvényekhez:

qQ	DG
00	0-, -0
01	11
10	01
11	1-, -0

Egyes esetekben többféle kitöltési lehetőség is van.

Példa: Készítsünk DG flip-flopot!

A megvalósítandó aszinkron állapotátábla. Ez itt egyben az egyetlen szekunder változó Karnaugh táblája is.



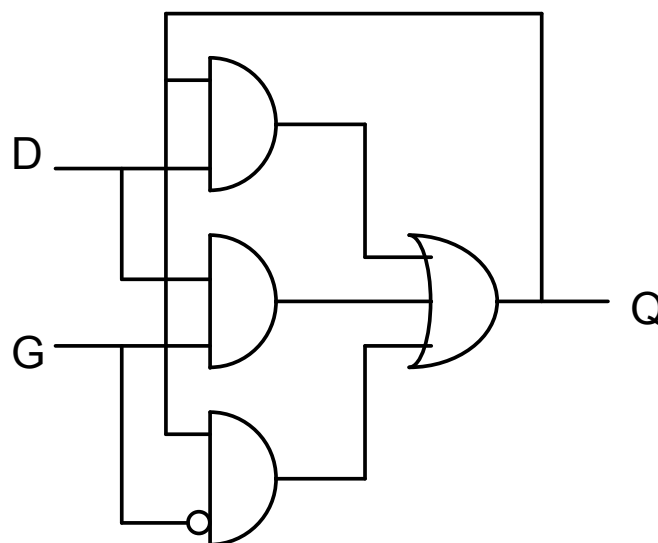
Ezt kell megvalósítani visszacsatolt kombinációs hálózattal.

Az Karnaugh táblát hazárdmentesen kell lefedni!
A hazárd hibás működéshez vezethet!

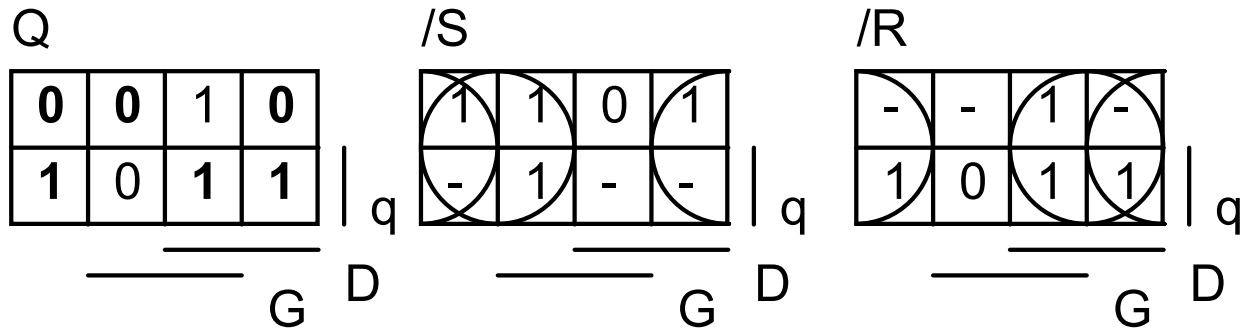
A hazárdmentes lefedés:

$$Q = qD + q/G + DG$$

A kapcsolási rajz:



Készítsünk negált bemenetű SR flip-flop segítségével DG flip-flopot.



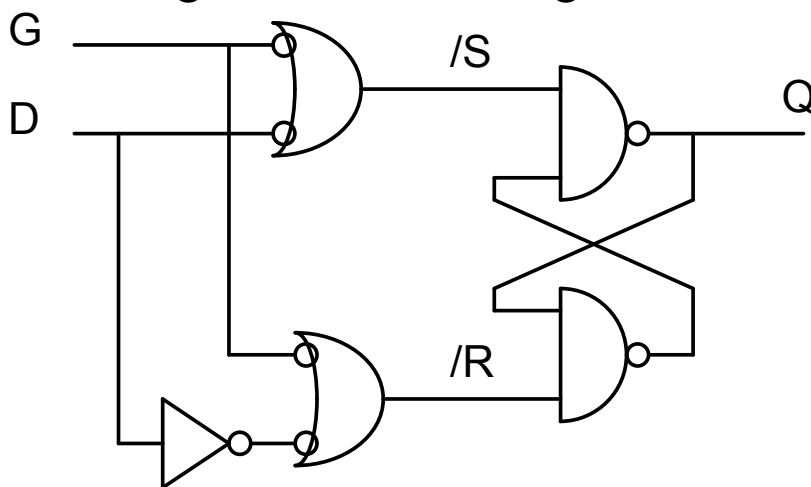
A vezérlési táblák az alábbi SR segédtábla segítségével tölthetők ki.

qQ	SR	/S/R
00	0-	1-
01	10	01
10	01	10
11	-0	-1

A hazardmentes lefedések:

$$/S = /D + /G \quad /R = D + /G$$

A homogén NAND-os megvalósítás:



Az alábbi egyenletekkel adott egy aszinkron sorrendi hálózat:

(X1, X2 a bemenetek, q1, q2 a két RS flip-flop kimenete.)

$$S1 = \overline{X2} \cdot q2 \cdot \overline{q1} \quad R1 = \overline{X2} \cdot q2 \cdot q1$$

$$S2 = X1 \cdot X2 \quad R2 = \overline{X2} \quad Z = q1$$

Tartalmaz-e a hálózat kritikus versenyhelyzetet?

