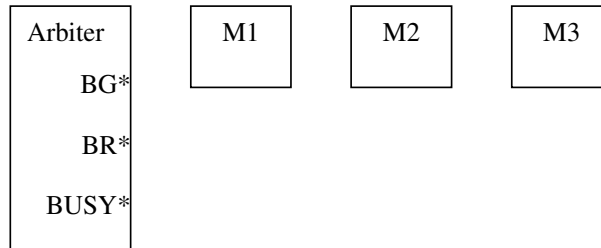


## Arbitráció

### 1. feladat

a) **Rajzolja** be a mellékelt ábrába egy közös kérés-felfűzött válasz elven működő arbitrációs rendszer jeleit.



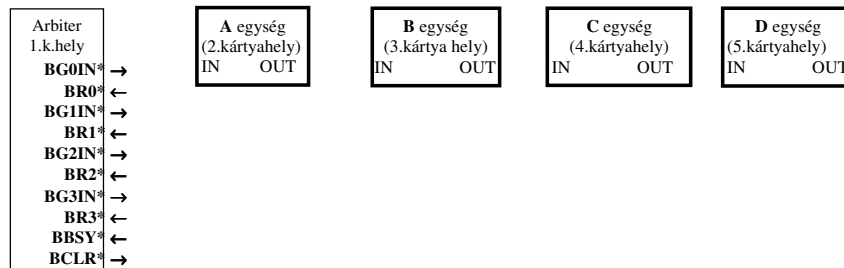
b) **Milyen** buszmegszerzési stratégiát valósít meg egy ilyen rendszer?

### 2. feladat

VME rendszernél az **A** és **D** egység a **BR0\***-án, az **B** és **C** egység a **BR1\***-en kér buszvezérlési jogot. Az arbiter a **BR3-BR0** szintek között Round-Robin elven működik. Pillanatnyi prioritás, **csökkenő** prioritási sorrendben, **BR3-BR2-BR1-BR0**.

**A** és **C** egység valamennyi, míg a **B** és **D** csak a **RWD** buszelengedési stratégiát képes megvalósítani.

a) **Rajzolja be** fenti feltételeknek megfelelően az arbitrációs rendszerhez tartozó jeleket az ábrába.



b) Milyen sorrendben kapják meg a buszvezérlési jogot?

c) Mi a sorrend ha az első egység adatátvitel alatt a 6.kártya-helyen lévő **E** egység **BR3\*-on** is kér buszvezérlési jogot?

### 3. feladat

Multibus II rendszernél az **A, B, C** master a mellékelt arbitrációs azonosító kódot adja a buszra.

**A: 110111**

**B: 101110**

**C: 111011**

a) **Mutassa be**, hogyan dől el, melyik master kapja meg elsőnek a buszvezérlési jogot.

b) **Milyen buszmegszerzési stratégiát** valósít meg a **MB II** mechanizmusa?

c) Az **A,B,C** bejelentkezése után három órajellel később a mellékelt arbitrációs kódú **D** és **E** egységnél is buszvezérlési igény keletkezik. **Hogyan alakul** a teljes sorrend?

**D: 100000**

**E: 000111**

d) **Mi lenne** az **E** egység normál kódja, amelynél maradéktalanul teljesülne az eredeti sémamegszerzési stratégia?

**4. feladat**

**Multibus II** rendszerben az **A, B, C** master az alábbi arbitrációs azonosító kódot adja a buszra:

**A: 100111**

**B: 101111**

**C: 100110**

a) **Mutassa be**, hogyan dől el, melyik master kapja meg elsőnek a buszvezérlési jogot.

b) Az **ABC** bejelentkezése után három órajellel később a mellékelt arbitrációs kódú **D** és **E** egységnél is buszvezérlési igény keletkezik. Adja meg, hogy az **ABCDE** egységek milyen sorrendben kapnak buszvezérlési jogot.

**D: 100001**

**E: 001111**

**5. feladat**

a) Hány **arbitrációs modul** (arbiter) és hány **master modul** lehet egy VME rendszerben?

b) Hány **arbitrációs modul** (arbiter) és hány **master modul** lehet egy **MB II** rendszerben?

c) **Hogyan biztosítja a MBII rendszer az egyenlő esélyű buszmegszerzési stratégiát?**

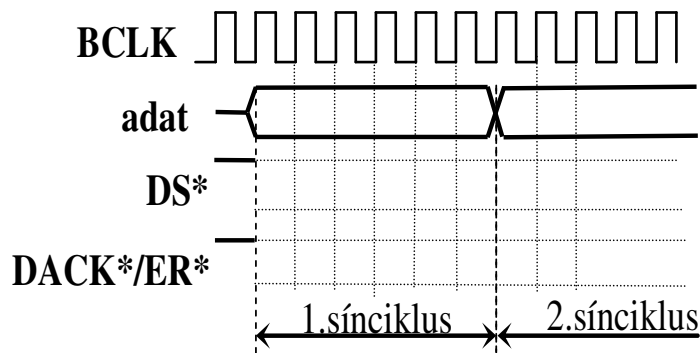
**Adatátviteli protokollok, időzítések****6. feladat**

**Teljesen reteszelt** protokollt alkalmazó **szemiszinkron** sínrendszer jelei láthatók a mellékelt ábrán (az órajel **10MHz**)

a) **Mekkora lehetne** egy írási sínciklus **minimális** ideje, egy ilyen rendszerben?

b) **Mi történik**, ha egy nem létező címre írna a rendszer?

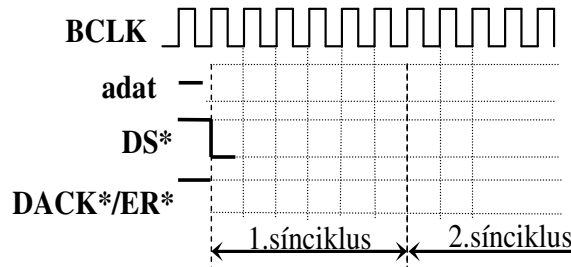
c) **Rajzolja be** a hiányzó jeleket és a jelátmenetek közötti, ok-okozati viszonyt mutató nyilakat **írás** esetén.



**7. feladat**

Teljesen **retesztelt** protokollt alkalmazó **szemiszinkron** sínrendszer jelei láthatók a mellékelt ábrán (az órajel **100MHz**).

- a) **Mekkora** lehet egy olvasási sínciklusban a **minimális elemi** időzítés, egy ilyen rendszerben?
- b) **Mi történik**, ilyen protokollnál, nem létező címről történő olvasás esetén (és hogyan fejeződik be a sínciklus)?
- c) **Rajzolja** be a hiányzó **jeleket** és a jelátmenetek közötti, **ok-okozati** viszonyt mutató **nyilakat** olvasás esetén.



**Megszakításkezelés**

**8. feladat**

Hány megszakítás kezelő modul és hány megszakításkérő modul lehet egy VME rendszerben?

**9. feladat**

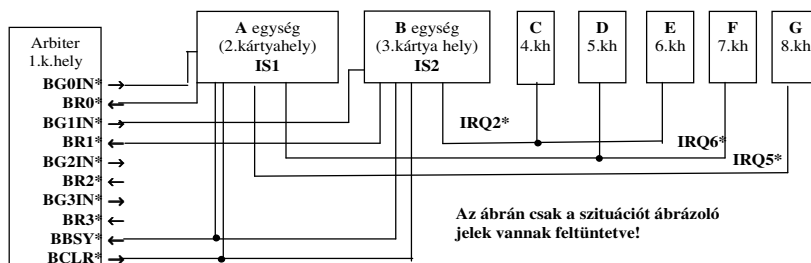
Magyarázza el mit jelent, hogy a **MULTIBUS II** virtuális megszakítási rendszert alkalmaz!

Hány kérő és kezelő lehet ebben a rendszerben?

**10. feladat**

VME rendszernél a 2.kártyahelyen (2.kh) **A** egység egy **IS1** jelű a 3kh.-en **B** egység egy **IS2** jelű megszakítás kezelőt tartalmazó master. Az **A** a **BR0\*** a **B** a **BR1\*** jelen kapcsolódik az arbiterhez. Az arbiter **Round-Robin** módon működik, a pillanatnyi prioritás csökkenő sorrendben **BR3-BR2-BR1-BR0**. **IS1** fogadja az **IRQ7\***, **IRQ6\***, **IRQ5\*** jeleket, míg **IS2** fogadja az összes többi megszakításkérő jelet. **IS1** és **IS2** **fix** prioritással működik a magasabb számú vonalon érkező kérés magasabb prioritást jelent a hozzá tartozó kiszolgálónál.

**C** egység (4.kh) és **E** egység (6.kh) az **IRQ2\***vezetékre, a **D** ( 5.kh) és **F** ( 7kh) egység az **IRQ6\*** vezetékre G egység (8.kh) **IRQ5\*** vezetékre kapcsolódik. **C,D,E,F,G** egységeknél egyszerre keletkezik megszakítás kérési esemény.



- a) **Adja meg**, milyen sorrendben kerülnek kiszolgálásra.
- b) **Mi** a sorrend, ha csak az **IS1** kezelő van a rendszerben **valamennyi** IRQ vonal számára?

## Folyamatok együttműködése

### 11. feladat

A P1 és P2 folyamatok a következő utasításokat hajtják végre párhuzamosan:

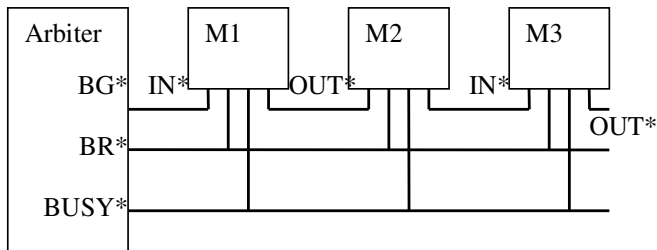
P1	P2
{ i = counter; counter = i+1; }	{ i = counter; counter = i- 1; }

A *counter* változó közös memóriaterületen helyezkedik el (mindkét folyamat elérí), és kezdeti értéke 10.

- a) **Adja meg**, hogy a lefutási sorrendtől függően mik lehetnek a *counter* változó lehetséges értékei, ha az egyes utasításokat oszthatatlan műveletek valósítják meg.
- b) **Mik lehetnek** a változó lehetséges értékei, ha a kapcsos zárójelek közé írt utasításokat egyetlen, oszthatatlan művelet valósítja meg?

## Megoldások

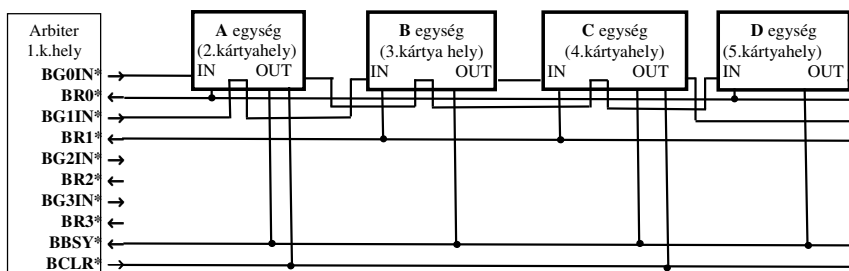
1. a)



**b) Stratégia:**

Prioritásos, mert a soros felfűzés egyértelműen meghatározza a kiszolgálási sorrendet. Egy egység csak akkor kaphat sínvezérlési jogot, ha az előtte lévők nem kértek.

2. a)



b) sorrend: BACD

c) sorrend: BAECD

3. a)

	ARB5*	ARB0*
A	1 1 0 1 1 1	1 1 1 1 1 1 1 1 1 1 1 1
B	1 0 1 1 1 0	1 0 1 1 1 1 1 0 1 1 1 0
C	1 1 1 0 1 1	1 1 1 1 1 1 1 1 1 1 1 1
	1 0 0 0 1 0	1 0 1 1 1 1 1 0 1 1 1 0 → B

b) Buszmegszerzési stratégia: Egyenlő esélyű  
 Indoklás: Normál prioritással senki nem léphet be a buszmegszerzési ciklusba mielőtt az utolsó egység kiszolgálása elkezdődik( így egy egység nem kerülhet kétszer kiszolgálásra)

c) sorrend: BEACD

d) E-kódja: 100111

4. a)

	ARB5*	ARB0*
A	1 0 0 1 1 1	1 0 0 1 1 1 1 1 0 0 1 1 1
B	1 0 1 1 1 1	1 0 1 1 1 1 1 1 1 0 1 1 1 1
C	1 0 0 1 1 0	1 0 0 1 1 0 1 0 0 1 1 0 1 0 0 1 1 0
	1 0 0 1 1 0	1 0 0 1 1 0 1 0 0 1 1 0 → C

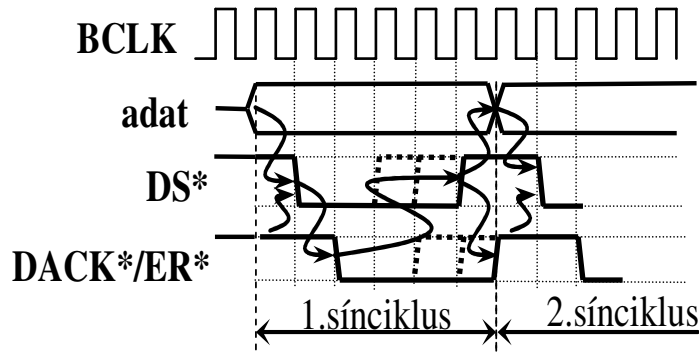
b) sorrend: CEABD

5. a) Egy arbiter, mert centralizált arbitrációs rendszert ír elő a szabvány. A mesterek száma elvileg nem korlátozott. Négy szintű rendszer, négy kérő vonal, mindegyikre több kérő csatlakozhat /négy engedélyezés szintenként láncolható/ szintenkénti láncolással tetszőlegesen bővíthető a mesterek száma.

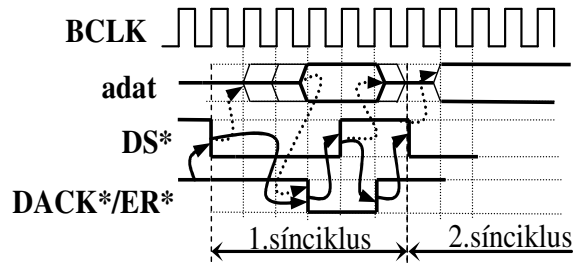
5. b) Maximum 20 arbiter lehet, mert elosztott arbitrációs rendszer, annyi arbitrációs részegység lehet ahány master. A mesterek száma 20, mert ennyi arbitrációs azonosító kódot definiáltak a párhuzamos versenyztetésű HW mechanizmushoz.

5. c) A különleges prioritású kérés esetét kivéve, egy master csak akkor léphet be az arbitrációs ciklusba, ha nincs függőben lévő más kérés (BRQ\* vonal).

- 6. a)  $T_{WCmin} = 400ns$
- 6. b) Leáll („lefagy”) az adatátviteli ciklus, mert nem kap válaszjelet.
- 6. c)



- 7. a)  $T_{min} = 10ns$
- 7. b) Válaszjel hiányában leáll a sínciklus. Egy time-out modul adott idő múlva abortálja a sínciklust.
- 7. c)



8. Hét kezelő lehet, mert a rendszer 7 IRQ vonalat használ és minden kezelőhöz legalább egy kérő vonalnak kell tartoznia. 256 kérő lehet, mert a kiszolgálásra kerülő megszakításkérő egység egy 8 bites vektorral azonosítja a hozzá rendelt kezelő programot, így 256 különböztethető meg.

9. Virtuális: Nincsenek a sínen a megszakításkéréshez és a kiszolgáláshoz hozzárendelt kérő-nyugtázó jelek. A megszakításkérést az üzenetátadási címtartományban egy megszakítás típusú váratlan üzenetként küld a kérő a kiszolgálónak. Megszakítás kezelő: 255+1 Kérő 255+1 lehet, mert az üzenet fejlécében 8 bit jelöli a célt (kiszolgáló) és 8 bit azonosítja a kérőt.

- 10. a) sorrend: CDEFG
- 10. b) sorrend: DFGCE
- 11. a)

(a) $i = counter;$	(c) $i = counter;$
(b) $counter = i+1;$	(d) $counter = i-1;$

Lehetséges sorrendek:	counter
a b c d	10
a c b d	9
a c d b	11
c d a b	10
c a b d	9
c a d b	11

- 11. b) counter = 10