

# Számítógép-architektúrák

## DMA kezelése – 8237 DMAC

(vázlat)

### DMA kezelés elmélete

A DMA témában jó összefoglaló található A2-ben (157-162. oldal).

Az anyag elérhető a **DMA-elmélet-es-gyakorlat.pdf** fájlban.

### 8237 DMA Controller általános megismerése

A 8237 DMA vezérlőt csak példaképpen, korlátozottan ismertetjük. Referenciának elegendő: **8237.pdf** (Ismertetéskor ezt követjük. Sajnos K1-ben nem ez, hanem a 8257 szerepel, ami nem képes memóriából memóriába történő átvitelre. Egyéb jellemzőiben hasonló, így érdeklődőknek ajánljuk a 76-91. oldal elolvasását.)

A 8237 DMA vezérlő perifériaként programozható, 16 I/O címet foglal el. (Az  $A_{3-0}$  címbiteket kötjük bele.)

Legfeljebb 4 DMA-s periféria kiszolgálásra alkalmas ( $DREQ_{3-0}$ ), de kaszkádosítható (csak azzal mi nem foglalkozunk).

Illesztése és felprogramozása a **8237.pdf** segédlet felhasználásával, az alábbiakban bemutatott módon történik.

### 8237 DMAC illesztése és felprogramozása

#### Mintafeladat:

*Adott egy olyan input periféria, amely két I/O portot foglal el ( $A_0$  címbitet kell belekötöni).  $A_0=1$  esetén a parancs regiszterét érjük el, ide két egymást követő out utasítással (alsó bájtt, felső bájtt) kell beleírni, hogy mennyi adatot kérünk tőle. Ha az adatok készen állnak, ezt a REQ lábán  $0 \rightarrow 1$  átmenettel jelzi. Ezután a megadott számú adat blokkosan (egymás után várakozás nélkül) kiolvasható az adat regiszteréből ( $A_0=0$ ). A beolvasásra használjunk 8237 DMA vezérlőt! Illesszünk még az MCS-85 rendszerbe 8kB EPROM-ot a 0000h címre és 8kB RAM-ot az E000h címre! Tervezzük meg a hardvert és írjuk meg a programot! (Kérjük a perifériától 500 bájtt adatot és tároljuk az E000h címtől!)*

#### A hardver tervezése:

Hardver tervezésének kritikus pontjai:

- Mivel a 8237 az  $A_{15-8}$  címbiteket a  $DB_{7-0}$  lábain adja ki, egy latch-et kell használnunk, amibe az adatbeírást (STB) a 8237 ADSTB jelével engedélyezzük, a kimenetét pedig a 8237 AEN jelével engedélyezzük!
- Nagyon fontos, hogy a perifériák címdekódereit a 8237 AEN jelével tiltani kell, nehogy a DMA átvitel közben megjelenő cím esetlegesen egyező bitmintája miatt valamelyik periféria engedélyezésre kerüljön!
- A 0-s csatorna használata esetén a periféria REQ kimenetét a 8237  $DREQ_0$  bemenetére kötjük, a 8237 0 aktív  $DACK_0$  kimenetével pedig egyrészt egy (0 aktív logikában VAGY kapuként működő) ÉS kapun keresztül engedélyezzük a periféria kiválasztását, másrészt egy másik ÉS kapu segítségével gondoskodunk perifériába kötött legelső címbit 0 értékéről is!

A többi értelemszerűen megy... A **DMA-elmélet-es-gyakorlat.pdf** fájl utolsó oldalán található ábrába rajzoljuk még be a 8kB EPROM-ot is!

## A szoftver tervezése:

Az ábra alatt található instrukciók alapján készítsük el periféria inicializálását és a DMA vezérlő felprogramozását végző programot! Az átvitel végét a főprogramból állapotolvasással figyeljük!

A 8237 felprogramozásának kritikus pontjai:

- A 00h címre illesztett 8237 felprogramozása során a **8237.pdf** segédlet végén található szoftver parancs címeket kell használnunk. (Például a parancs regiszter a 08h címen írható, a státusz regiszter ugyanezen a címen olvasható.)
- A felprogramozást egy Master Clear (általános törlés) paranccsal kezdjük, ami a 0Dh címre való írással váltható ki. (A kivitt érték érdektelen.)
- A felprogramozás idejére az összes csatornát tiltjuk a 0Fh címre írt 1111b segítségével.
- A parancs regiszterbe (Command Register a 08h címen) 00000000b értéket írunk: DACK(L), DREQ(H), normál írás, fix prioritás, normál időzítés, DMA engedélyezése, x=közömbös, memória-memória átvitel tiltva.
- Ezután felprogramozzuk a 0-s csatornát (CH 0) a következők szerint:
  - A mód regiszterbe a 0Bh címre 10000100b értéket írunk: blokkos átvitel, címnövelés, nincs AI, memória írás, 0-ás csatorna
  - A 0-s csatorna címregisztere a 00h címen található, ide beírjuk az E000h memória kezdőcímet alsó bájt, felső bájt sorrendben (inicializáláskor a bájt mutató 0-ra állt be, ezért nem kellett nekünk külön törölni).
  - A 0-s csatorna számláló regisztere a 01h címen található, ide beírunk a kért 500-nál 1-el kisebb (!) számot alsó bájt, felső bájt sorrendben. (A 8237 DMAC-t úgy tervezték, hogy nem 0-nál, hanem FFFFh-nál (-1) van TC (Terminal Count)!)
- Egyedileg engedélyezzük a 0-s csatornát a csatornánkénti maszkállítás módszerével: a 0ah címre 000b írásával: engedélyezés, 0-s csatorna.

A 8237 felprogramozása után utasítjuk a perifériát, hogy készítse el számunkra az adatokat és ha elkészült, indítsa el az átvitelt.

Ezután folytonosan olvassuk a 8237 státusz regiszterét a 08h címen mindaddig, amíg a beolvasott érték 01h-val maszkolva (ANI 01h) 0-t ad eredményül. (JZ vissza az állapotolvasásra.) A bitenkénti logikai ÉS művelet 0-tól eltérő eredménye jelzi, hogy a kért számú adat átvitele megtörtént.

## Gyakorló feladatok:

*1. A mintafeladat hardverét programozzuk fel úgy, hogy a DMA vezérlő az EPROM tartalmát másolja át a RAM-ba! (8237 felprogramozása, majd szoftver DMA kérés kiadása)*

*2. A mintafeladatot egészítsük ki azzal, hogy állapotolvasás helyett a 8237 0 aktív EOP kimenetén kiadott impulzus végét (felfutó él) egy 8259A ITC érzékelje, és jelezze az átvitel végét a 8085 processzornak. Programozzuk is fel a megszakításvezérlőt!*

Ajánlott olvasmányok:

K1: Grantner János, Horváth István, László Zoltán: Mikroprocesszor alkalmazási segédlet, Műegyetemi Kiadó, 1997. (76-91. oldal)

A2: Benesóczky Zoltán: Digitális tervezés funkcionális elemekkel és mikroprocesszorokkal, Műegyetemi Kiadó, Budapest, 1998. (157-162. oldal)