

Számítógép-architektúrák

Perifériakezelési módszerek

(vázlat)

Feltétel nélküli

A perifériának (és persze a processzornak) mindig rendelkezésre kell állnia. Az átvitel **in/out** utasításokkal történik. Példák: kapcsolósor állapotának beolvasása, LED-ek kigyújtása latch-en keresztül.

Előny: nagyon egyszerű megvalósítás.

Hátrány: nem mindig alkalmazható.

Feltételes, ez megvalósítható:

a) állapotolvasással

- A periféria egy tároló beállításával (vonal szintjével, le/felfutó éllel, esetleg ezek együttesével) jelzi az átviteli igényét.
- A processzornak folyamatosan olvasnia kell a periféria állapotát, egyébként az igény elveszhet!
- Igény esetén a processzor végrehajtja az átvitelt.

Előny: egyszerű hardver.

Hátrány: Az állapotolvasás foglalja a processzort. Ha a processzornak más feladata is van, az állapotolvasás áttekinthetlenné teszi a programot.

b) megszakítással

- A periféria a fenti módon jelzi az átviteli igényét.
- A periféria jelzését közvetlenül vagy megszakításvezérlőn keresztül a processzor megfelelő bemenetére illesztettük.
- A periféria igénye esetén a processzor által éppen végrehajtott program futása megszakad, az igény kiszolgálásra kerül, majd a megszakított program futása folytatódik.

Előny: a processzornak nem kell a periféria állapotát figyelnie.

Hátrány: valamivel bonyolultabb hardver, sok adat átvitele esetén a regiszterek értékének mentése és visszatöltése jelentősen ronthatja a teljesítményt.

c) DMA vezérlő segítségével

- A periféria a fenti módon jelzi az átviteli igényét.
- A periféria jelzését a DMA vezérlő dolgozza fel.
- A periféria igénye esetén a DMA vezérlő elveszi a rendszersínt a processzortól és az átvitelt maga hajtja végre.
- Az átvitel megtörténtét a DMA vezérlőnek valamilyen módon jeleznie kell a processzor felé (például státusz beállításával vagy megszakítással)!
- A DMA vezérlő visszaadja a rendszersínt a processzornak

Előny: Hatékonyabb adatátvitel – még a feltétel nélkülihez képest is, mivel nincs program-végrehajtás (például: in/out utasítások felhozása, mutatók és számlálók kezelése, stb.), az adatbuszon keresztül a memória és a periféria között csak az adatok áramlanak.

Hátrány (az előnyök ára): bonyolultabb hardver (DMA vezérlő szükséges), a DMA vezérlőt fel kell programozni, és tipikusan megszakításkezelésre is szükség van.

Megjegyzés:

A feltétel nélküli, az állapotolvasásos és a megszakításos esetekben az adatokat a CPU viszi át, míg DMA vezérlő alkalmazása esetén a DMA vezérlő hajtja végre az adatátvitelt.

Feladatok:

1. Példa feltétel nélküli periféria kezelésre

Illesszünk 8085' processzorhoz 8db kapcsolót (ellenállások felhasználásával) az 1Ch perifériacímre és 8 db LED-et (8 bites latch felhasználásával) az 1Dh perifériacímre. Írjunk programrészletet, amely beolvassa a kapcsolók állását és megjeleníti a LED-eken! (A program tárolásához szükséges ROM illesztésével most nem foglalkozunk.)

2. Példa állapotolvasásos periféria kezelésre (vezérléssel kiegészítve)

Illesszünk 8085' processzorhoz 8kB ROM-ot a 0000h címre, 8kB RAM-ot az E000h címre, és egy 2 I/O címet elfoglaló perifériát a 06h/07h perifériacímre úgy, hogy a 07h címet íráskor vezérlésre, olvasáskor állapotolvasásra; a 06h címet pedig adatátvitelre lehessen használni! Vezérléshez a bitkiosztás: D₀: egy bájt kivitelének igénye, D₁: folytonos kivitel igénye, D₂: folytonos kivitel vége, D₃: egy bájt bevitelének igénye, D₄: folytonos bevitel igénye, D₅: folytonos bevitel vége. Státuszolvasásnál a bitkiosztás: D₀: egy bájt kivihető az eszközbe, D₁: egy bájt beolvasható az eszközből. Készítsünk programot, amely előbb folytonos átvitellel 100 bájtot olvas be az eszközből és eltárolja valamilyen alkalmas memóriaterületen, majd a 100 bájtot folytonos átvitellel fordított sorrendben kiírja a perifériába!

A hardverre egy minta megoldás: **Periferiaillesztes.pdf** (1. oldal)

3. Részben ismétlődő jellegű feladat.

Illesszünk 8085' processzorhoz 3x 2kB ROM-ot a 0000h címtől, 1kB NVRAM-ot az 1800h címtől (nem teljes címdekódolás megengedett!) és 7x 8kB RAM-ot a 2000h címtől. Az NVRAM (0 aktív) NE bemenetét egy adott perifériacímre írt D₀ adatbittel lehessen vezérelni!

Minta megoldás: **Periferiaillesztes.pdf** (2. oldal)

Ajánlott olvasmány az elméleti részhez:

Benesóczky Zoltán: Digitális tervezés funkcionális elemekkel és mikroprocesszorokkal, Műegyetemi Kiadó, Budapest, 1998. (162-164. o.)