

Írás idődiagramja

4.59. ábra.

4.3. A MIKROPROCESSZOR ÉS KÖRNYEZETE

Az LSI áramkörök közül a legfontosabbat, a mikroprocesszorokat egy másik jegyzet fogja részletesen tárgyalni. Itt csak néhány, a mikroprocesszorokkal kapcsolatban gyakran előforduló kifejezést, fogalmat vezetünk be.

A mikroprocesszoros rendszerek, berendezések működésének megértéséhez, tervezéséhez, megvalósításához az eddigi, a digitális technikában megszokott - áramkörök összehuzalczásán alapuló - szemlélettől és gyakorlattól el kell szakadni.

Minden mikroprocesszor - függetlenül attól, hogy milyen jellem-

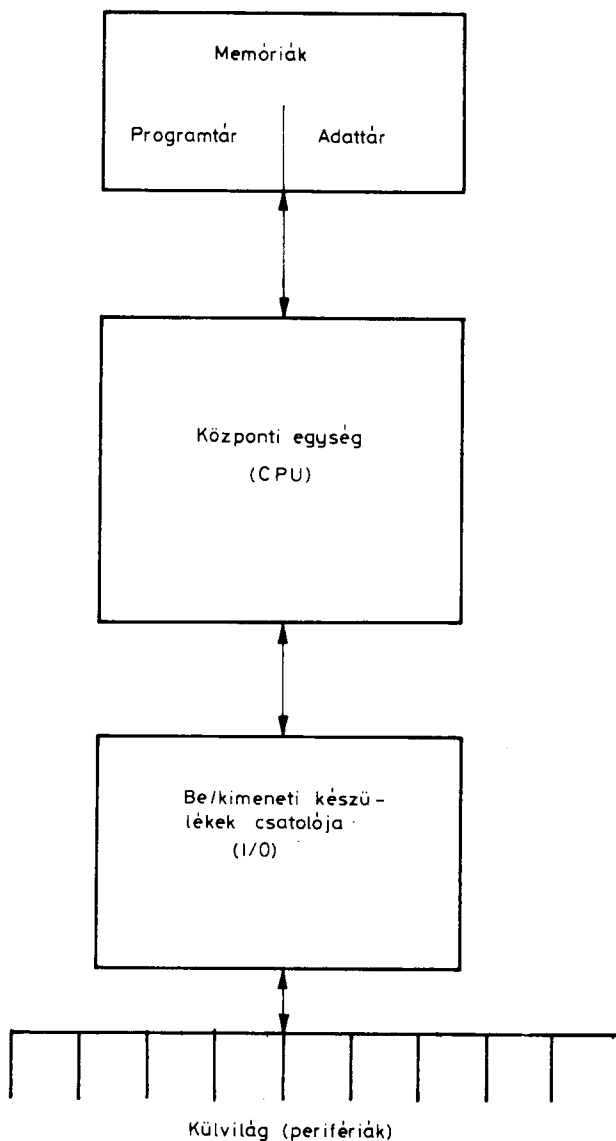
zőkkel rendelkezik: hány bites, stb. - a működéséhez szükséges környezetével együtt (óragerátor, busz meghajtók, memóriák, stb.) - függetlenül attól, hogy a környezetnek mekkora része van összeintegrálva a központi egységgel - egy mikroszámítógépnek tekinthető.

Ebből következik, hogy működését a benne tárolt működtető program, a "software" fogja meghatározni. A program utasítások sorozatát jelenti: a berendezés működése pedig ezen utasítások végrehajtását. Az utasításokat csak akkor tudja a berendezés végrehajtani, ha a mikroprocesszoros berendezés felépítése, a "hardware" ezt lehetővé teszi.

Mikroprocesszor típustól függetlenül egy mikroszámítógépnek a következő főbb egységekkel kell rendelkeznie ahhoz, hogy utasításokat végre tudjon hajtani:

- központi egység, amely a többi egységgel való kommunikációt vezérli, irányítja (CPU),
- programtár és adattár, az egyikben a működtető program (ROM), a másikban a program működése során keletkező vagy a külvilágból bevitt adatok tárolódnak (RAM),
- be/kiviteli egységek, amelyek a külvilággal pl. perifériákkal: billentyűzet, képernyő, relék, kijelzők, stb. tartják a kapcsolatot,
- kommunikációs utvonalak, amelyek az egységek közötti információ-áramlást biztosítják (sin vagy busz-rendszer).

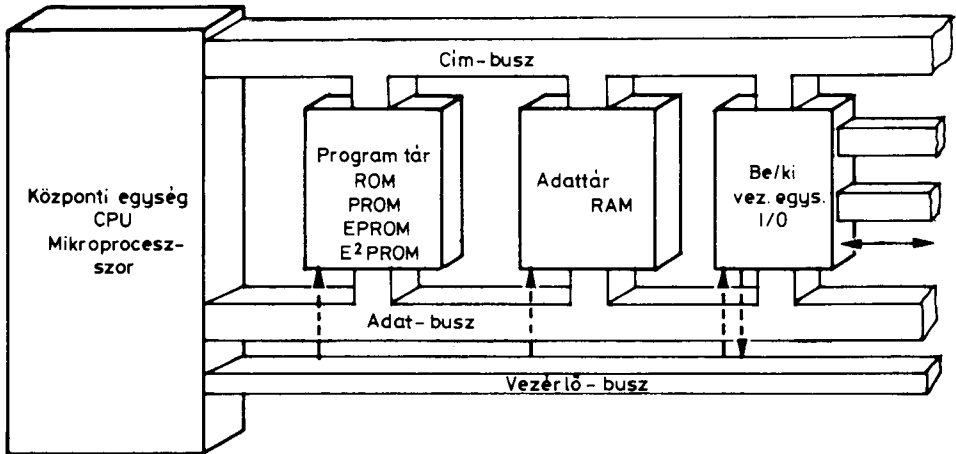
A 4.60. ábrán bemutatjuk egy számítógép elvi felépítését, a 4.61. ábrán pedig egy mikroszámítógép tipikus felépítését. Az ábrán látható három kommunikációs utvonal funkciója a következő: a központi egység az ún. cimbuszon adja ki azt az információt, hogy melyik egységgel kíván kommunikálni, vagyis, hogy melyik címről kíván olvasni, ill. hogy hova, melyik címre akar információt kiküldeni. A vezérlőbusz vezetékein adja ki a központi egység az általa végezni kívánt műveletre vonatkozó információját - például azt, hogy a megcímzett egységtől adatot vár, vagyis olvasni szeretne. A legtöbb processzor-típus azt is megkülönbözteti egy másik vezérlőjellel, hogy memóriától vagy I/O egységtől várja az adatot. (Néhány processzornál nincs külön vezérlőbusz, hanem magán az adatbuszon küld felváltva vezérlőszót és adatot.)



4.60. ábra.

Az adatbusz vezetékei szolgálnak arra, hogy az adat - olvasás-
kor - a megcímzett egységtől eljusson a központi egységhez,
íráskor pedig a központi egységtől a megcímzett egységhez.
Sin-, vagy busz-rendszer alatt olyan egymástól elszigetelt ve-
zetések kötegét értjük, amelyet párhuzamos információ-átvitel

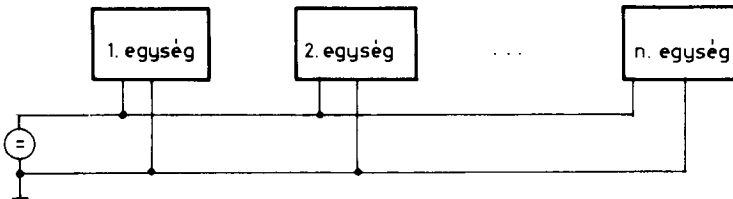
céljára hoztak létre, több egység rácsatlakozását és egymással való kommunikációját teszi lehetővé.



4.61. ábra.

A kommunikáció módjától függően a buszok között megkülönböztetünk

- egyirányú átvitelt lehetővé tevő (pl. tápfeszültség-busz, amely a rendszer működéséhez szükséges tápfeszültséget juttatja el a részegységekhez (4.62. ábra),

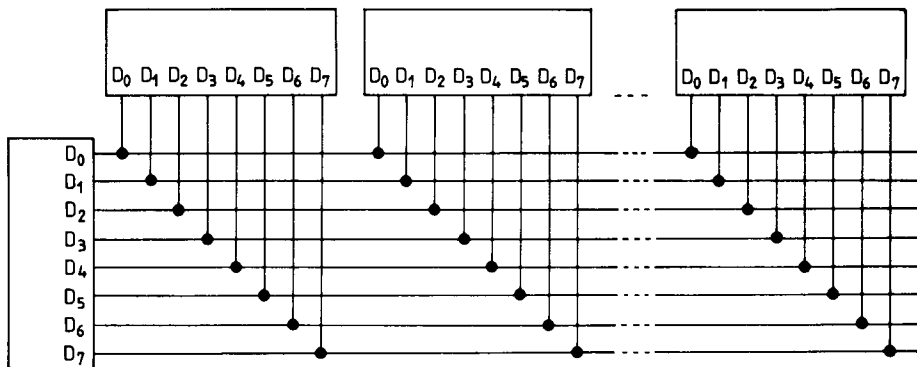


4.62. ábra.

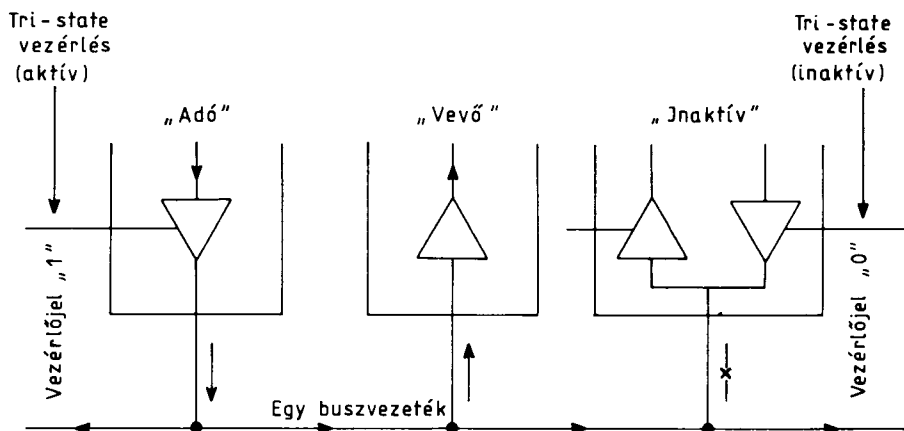
- kétirányú adatátvitelt lehetővé tevő (pl. az adatbusz, amely a processzor szóhosszuságától függő darabszámú vezetékből álló vezeték-csoport - 4.63. ábra).

A kétirányú (bi-directional) buszra párhuzamosan rácsatlakozó egységek közül egyszerre csak egyetlen egy lehet "adó", a többi egység közül egy vagy esetleg több is lehet "vevő", de a kommunikációban most nem résztvevő "adó"-knak inaktív állapotban kell lennie. Ezt a feltételt könnyűszerrel a "tri-state" (három állapotu) kimenetekkel rendelkező egységek tudják teljesíteni.

síteni, bonyolultabb vezérléssel az open-kollektoros kimenetek is busszá szervezhetőek. A 4.64. ábrán egy kétirányú busz egyetlen vezetékére csatlakozó "tri-state" áramköröket mutatjuk be.



4.63. ábra.

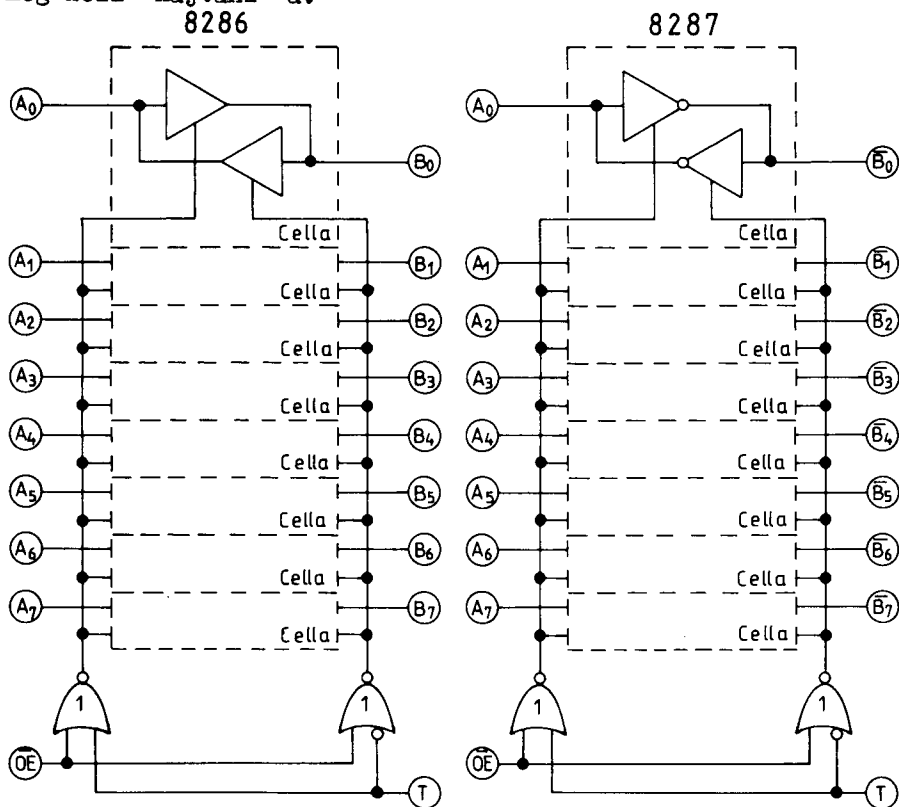


4.64. ábra.

Forgalomban vannak 4 vagy 8 bites kialakítású, egy vagy kétirányú adatátvitelt megvalósító ún. buszmeghajtó integrált áramkörök, mind bipoláris, mind MOS technológiával (4.65. ábra).

A buszmeghajtó áramkörökre azért van szükség, mert pl. a legtöbb processzor kimenőjeleire a FANOUT = 1, vagyis a legfeljebb 1 TTL bemenettel terhelhetők. Tehát az "adó"-egységnél meghajtó, erősítő eszközökre van szükség, hogy a buszvonalakra csatlakozó egységek a bemeneteken kapott logikai jeleket

egyértelműen fel tudják ismerni, azaz az "adó"-nak a buszt meg kell "hajtani"-a.



Logikai rajzok

8286				8287			
A ₀	1	20	V _{CC}	A ₀	1	20	V _{CC}
A ₁	2	19	B ₀	A ₁	2	19	$\overline{B_0}$
A ₂	3	18	B ₁	A ₂	3	18	$\overline{B_1}$
A ₃	4	17	B ₂	A ₃	4	17	$\overline{B_2}$
A ₄	5	16	B ₃	A ₄	5	16	$\overline{B_3}$
A ₅	6	15	B ₄	A ₅	6	15	$\overline{B_4}$
A ₆	7	14	B ₅	A ₆	7	14	$\overline{B_5}$
A ₇	8	13	B ₆	A ₇	8	13	$\overline{B_6}$
OE	9	12	B ₇	OE	9	12	$\overline{B_7}$
GND	10	11	T	GND	10	11	T

Tok lábkiosztások

A ₀ ÷ A ₇ : „Helyi”-busz adata	
B ₀ ÷ B ₇ : Rendszer-busz adata	
\overline{OE}	: Kimenet engedélyezés
T	: Vezérlés (tri-state)

A kivezetések megnevezései

4.65. ábra.

Ehhez a témához szorosan hozzátartozik az ún. "CHIP SELECT" ("áramkör kiválasztás") jel fogalma. A buszra csatlakozó egységeknek egyértelműen tudniuk kell, hogy a buszon éppen létrejövő kommunikációban részt kell-e venniük vagy sem. Arról már esett szó, hogy azt mindig a processzor dönti el, hogy melyik egységgel kíván kommunikálni, s ezt az információt a cibuszon közli az egységekkel. A cibuszon megjelenő bitmintha viszont egy konkrét címet jelent, amely cím mögött a három fő egység (programtár, adattár, I/O egység) bármelyikének egy konkrét tárolóeszköze (rekesz, regiszter) állhat. Tehát mindegyik egységnek fel kell ismernie, vagy valamiképpen tudomást kell szereznie arról, hogy a kiküldött cím az ő "tartományába" esik-e. Ezt az "áramkör kiválasztva" jelet hívjuk Chip Select jelnek (rövidítve CS). Ezek az egyes egységeket aktív vagy inaktív állapotba vezérlő jelek a cibusz vonalainak olyan dekódolásával állíthatók elő, amely az egyes egységekhez tartozó címtartományok szétválasztását adják.

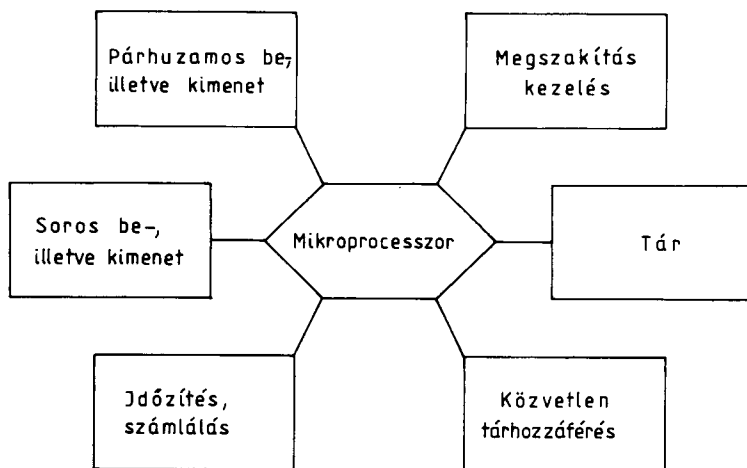
A 4.61. ábrán a buszok jelölése mutatja, hogy nincsenek lezárva, folytathatók. Éppen ebben rejlik a mikroprocesszoros rendszerek egyik legnagyobb előnye, hogy közel azonos funkcionális felépítés dacára szinte korlátlanul bővíthetők a rendszerek a további egységeknek a buszokra való csatlakoztatásával. Ily módon kis- és nagybonyolultságú feladat, sok periférikus eszköz működtetése mind megoldható viszonylag egyszerű hardware műveletekkel.

4.4. KIEGÉSZÍTŐ ELEMEEK

A mikroprocesszorok alkalmazásának kezdeti szakaszában nem volt semmiféle kiegészítő elem. Az első generációs mikroprocesszorok alkalmazásának természetes velejárója volt az a nagyszámú kis és közepes integráltságú elem, amely "körülvette" a mikroprocesszort, lehetővé téve kapcsolatát a "külvilág"-gal, a megoldandó feladattal - hozzáillesztve az érzékelők, kapcsolók, stb., azaz az információ-források és a kijelzők, kiíró- vagy végrehajtó elemek jeleit a mikroprocesszorhoz. A jelek be- és kivitelét már a mikroprocesszor végezte,

amely a kezdeti egyszerű alkalmazások során olyan feladatokat is ellátott, mint pl. egy alfanumerikus billentyűzet figyelése, az egyes billentyűk állapotának egymás után való, folyamatos végigpásztázásával (scanning), majd egy billentyű lenyomásakor - a billentyűmátrixban elfoglalt pozíciója alapján - a hozzárendelt kód szerinti alprogram indítása. A mikroprocesszorok ilyen egyszerű feladatokkal való "terhelése" a későbbi, bonyolultabb feladatok jelentkezésénél már nem volt problémamentes. Így az első időszak rendszereinek hardware-felépítése csak nagyon kismértékben használta ki az LSI technológiából származó előnyöket. Ezen kívántak változtatni, amikor a gyártók egyszerű kiegészítő funkciókra is egységeket hoztak létre, amelyekhez egy adott mikroprocesszorral való közvetlen (minden további elem nélkül való) kapcsolat létrehozására a mikroprocesszor busz-jelei (cim-, adat- és vezérlőjelek) közvetlenül rácsatlakoztathatók (családely).

Ilyen módon szinte valamennyi mikroprocesszor tipushoz létrehoztak kiegészítő elemeket, amelyekkel a rendszerelemek közötti és a feladattal való kapcsolat könnyebben létrehozható. A 4.66. ábrán hat ilyen alap feladatkört mutatunk be, amelyekhez kapcsolódó elemeket ebben a "második generációs" korszakban dolgoztak ki.

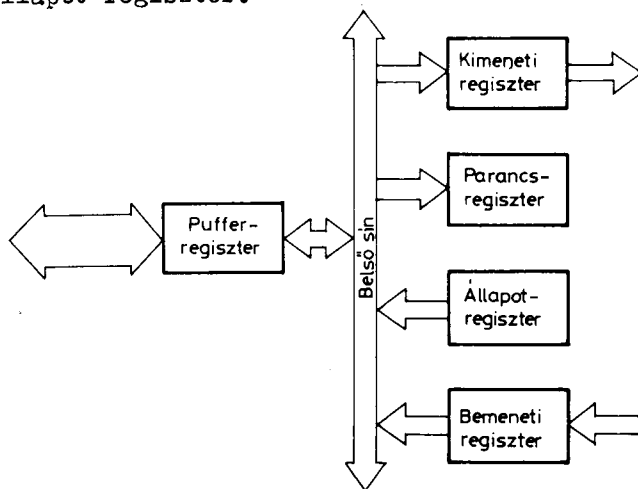


4.66. ábra.

Ezek a megszakítás (interrupt) kezelés, közvetlen tárhozzáférés (DMA: Direct Memory Access); tár- és tárkiválasztó elemek, időzítők és számlálók, valamint soros és párhuzamos be- és kimeneti (I/O) elemek. Ezzel párhuzamosan - az analóg IC-kegyártási eredményeit is figyelembe véve - a közvetlen, a folyamatokhoz való kapcsolat elemei (pl. analóg-digitális és digitális-analóg konverterek, multiplexerek, stb.) és a nagytávolságu adatátvitel elemei: adó-vevők, kódoló-dekódolók (CODEC), modulátor-demodulátorok (MODEM) is többé-kevésbé processzor típustól független, és standardizált formában jelentek meg.

A következő "harmadik generáció" egyre összetettebb periféria-vezérlői már programozhatók. A processzor saját cím-, adat- és vezérlőjelei segítségével információcserét folytat a kiválasztott vezérlő valamelyik regiszterével (sokszor egy buffer-regiszter közbeiktatásával). Ezek a regiszterek funkciójuk szerint a következő 4 csoportba oszthatók (4.67. ábra):

- bemeneti regiszter,
- kimeneti regiszter,
- parancs regiszter,
- állapot regiszter.



4.67. ábra.

A be- és kimeneti regiszterek a külső adatforrás, ill. nyelő és a mikroprocesszor valamelyik külső regisztere közötti adat-

utat biztosítják, úgy, hogy mindkét oldalon független, aszinkron adatcserét lehessen létrehozni.

A vezérlő parancsregisztere(i) a működés módját és paramétereit szabja meg a processzortól kapott utasításoknak megfelelően. Az átvitt információ tehát azonos módon juttatható a vezérlőbe, akár adat, akár parancs típusu is az.

Ilyen módon programozható pl. egy soros átvitelt vezérlő elem (8251) esetén az átvitel típusa (szinkron vagy aszinkron) az átvitt karakter hossza (bitszám), az átvitel sebessége (Baud-rate), a hibajelzést szolgáló paritásképzés módja (páros, páratlan, nincs), stb.

A vezérlő mindenkori állapotát az állapot- vagy más szóval státuszregiszter mutatja. A processzor ennek kiolvasásával szerezhethet tudomást pl. az átvitt információ meghibásodásáról vagy ezen belül is a meghibásodás módjáról (pl. keretezési hiba, amikor aszinkron átvitel esetén a záró stopbit helyén nem a megfelelő bit érkezik, tehát az információ "elcsuszott").

Hasonló elven működik a többi programozható perifériavezérlő elem is. Pl. a megszakításvezérlő (8259) esetén az egyes megszakításkérésekhez rendelt prioritásszint programozhatóan változtatható.

A közvetlen tárhozzáférés (DMA vezérlő: 8257) esetén szükséges átviteli paraméterek - kezdő cím és blokkhossz - mellett a mód is megadható. Ez lehet cikluslopásos (amely az egyes, a processzor számára rendelkezésre álló tárciklusok közé ékelt ciklusokban végzi az átvitelt és ezért "átlátszónak", transparentnek is szokták nevezni), vagy egy vagy több blokk folyamatos átvitelét biztosító blokk- (burst: sorozat) átvitel.

Egy vezérlő több azonos funkció kiszolgálását is végezhet és vannak többféle feladatot ellátó komplex vezérlők is. (Pl. az Intersil 6100, a DEC PDP8/E miniszámítógép központi egységet emuláló CMOS mikroprocesszor 6102 kiegészítő eleme DMA vezérlő mellett tárbővítőt, intervallum időzítőt és megszakításvezérlőt is tartalmaz.)

A következő "negyedik generáció" kiegészítő elemei már nem csupán programozhatók, hanem összetett perifériák vezérlői, műveletcsoport vagy feladatorientált önálló processzorai. Ezeket összefoglalóan szokták TPU (Task Processing Unit: feladat-

feldolgozó egység) egységnek is nevezni. Illusztrációként bemutatunk néhányat közülük egy-egy tipikus feladat megoldására.

Új típusu, nagy (millió byte) címtartományu mikroproceszorok és a korszerű operációs rendszerek alkalmazása a tárkezelés - számítógépek esetében már megszokott - formáit igényelte. Az MMU (Memory Management Unit: tárkezelő egység) a tár változtatható felosztását (segmentation), dinamikusan változtatható áthelyezését (relocation) és kijelölt szegmensek védelmét (protection) végzi.

A BAU (Bus Arbiter Unit: sinvezérlőegység) a sínrendszer osztott idejű felhasználásából adódó konfliktushelyzeteket oldja meg (Arbiter: döntőbíró), megfelelő prioritás-áramkörök segítségével, előállítva a címdekódolás és a visszajelzés aszinkron átvitel (hand-shake) vezérlőjeleit is.

A kiegészítő elemek amellet, hogy az elembázis leggyorsabban fejlődő területét képezik, a mindenkori technológia alapján megvalósított legkomplexebb (legnagyobb lapkaméret, legtöbb elemi áramkör egy lapkán - a nem ismétlődő struktúráju elemeket figyelembe véve) LSI és VLSI áramköröket is jelentik.