

2. fejezet

A mikrokontroller

(A 8031/51 család)

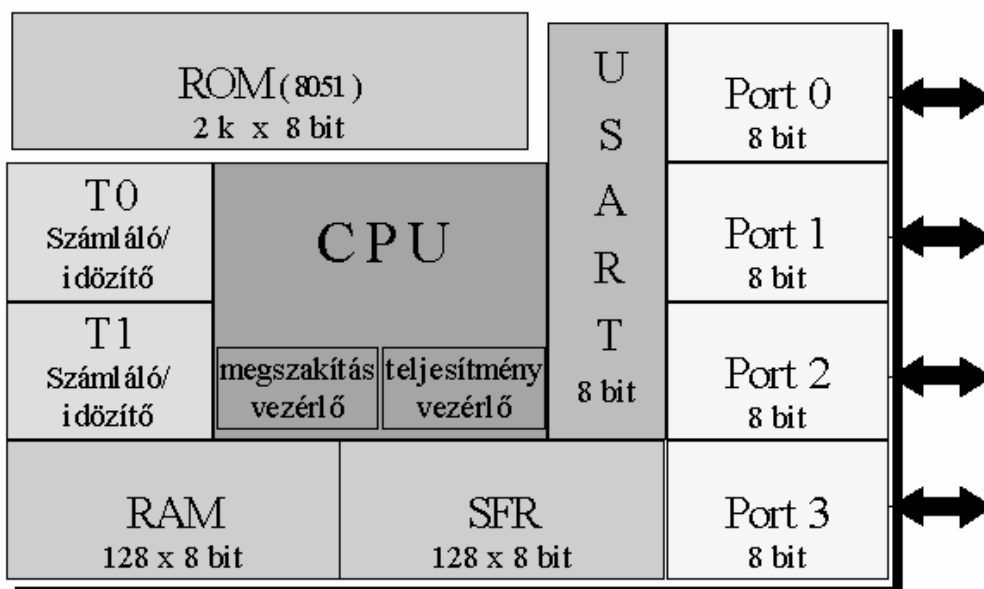
A 2. fejezet tartalomjegyzéke:

2.	A mikrokontroller.....	23
2.1.	A 8051 mikrokontroller család.....	23
2.1.1.	A CPU és regiszterei	24
2.1.2.	Belső memória	24
⇒	<i>A Bankok és használatuk</i>	25
⇒	<i>A bit-címezhető memória, és használata</i>	26
⇒	<i>Általános felhasználású memóriaterület</i>	26
⇒	<i>Speciális Funkció Regiszterek (SFR)</i>	27
2.1.3.	A Portok	27
2.1.4.	Programozható időzítő / számláló egységek	28
⇒	<i>Időzítő / számlálók vezérlése</i>	29
⇒	<i>Időzítő / számlálók 0-ás és 1-es üzemmódjai</i>	30
⇒	<i>Időzítő / számlálók 2-es üzemmódja</i>	31
⇒	<i>A T0 időzítő/számláló 3-as üzemmódja</i>	31
2.1.5.	Soros vonali illesztő	32
⇒	<i>Adás</i>	34
⇒	<i>Vétel</i>	35
2.1.6.	Megszakítások.....	35
⇒	<i>Megszakítások kiszolgálása</i>	36
2.1.7.	Teljesítmény felvétel vezérlése	37
2.2.	Külső memóriák illesztése	37
⇒	<i>Adat-, és programmemória kezelése</i>	37
⇒	<i>Külső perifériák csatlakoztatása</i>	38

2. A mikrokontroller

A mikrokontroller egyetlen közös áramkörbe (chip-be) integrált csökkentett utasításkészletű **mikroprocesszor** és több **periféria áramkör**. A több cég által is gyártott 8031/51 típusú mikrokontrollerben a következő perifériák találhatók: (14. ábra)

- két 16 bites **időzítő/számláló** (T0, T1),
- négy 8 bites - bitenként is kezelhető - **Port** (P0, P1, P2, P3),
- 128 bájt **belső memória** (RAM),
- 128 bájt **SFR memória** (RAM),
- 4 Kbajt **programmemória** (ROM csak a 8051 típusban),
- RS 232 szabvány szerinti **soros vonalillesztő**,
- **megszakítás** vezérlő egység,



14. ábra

2.1. A 8051 mikrokontroller család

A 8051 mikrokontroller családnak különböző típusú mikrokontrollerei vannak. A család alap-típusa a 8031/51 kontroller. A továbbfejlesztett változatok mindegyikében változatlanul benne van az alaptípus minden eleme, és további perifériákat is tartalmaz, vagyis működésben **felülről kompatibilisak**. A család minden tagjának **azonos az utasítás-készlete**. A bővített perifériák miatt több SFR regisztert, illetve megszakítást használható. A kontroller család elemeit több cég is (Siemens, Philips, Dallas, Atmell stb.) gyártja és forgalmazza.

Néhány elterjedt típus és periféria bővítései :

- **8032/52** 3. időzítő/számláló (**T2**),
- **80C552** 3. időzítő/számláló (**T2**),
 Watchdog számláló (**T3**),
 8 csatornás (10 bites) **A/D** átalakító,
 2. soros vonalillesztő (**I²C** busz),
 két **PWM** modulációs kimenet.
- **80515/535** 3. időzítő/számláló (**T2**),
 8 csatornás (8 bites) **A/D** átalakító,
 programozható referencia-feszültség,
 2. soros vonalillesztő (RS232),

2.1.1. A CPU és regiszterei

A mikrokontroller központi műveletvégző és vezérlő egysége (CPU) - a tulajdonképpeni mikroprocesszor rész - hajtja végre

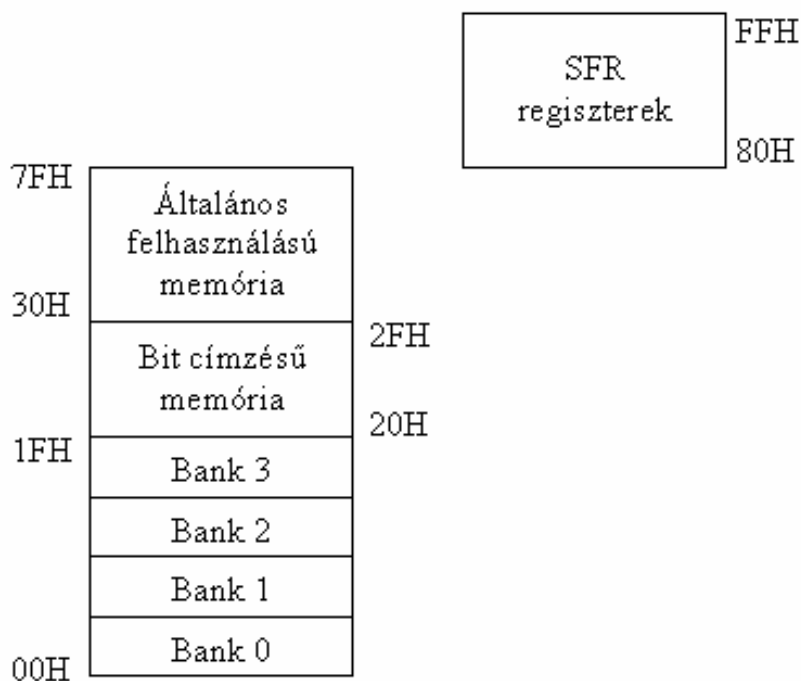
- az **utasításbeolvasás** -t, és "értelmezés"-t,
- az **aritmetikai**-, és **logikai** műveleteket,
- a belső **működés** - utasítástól függő - **vezérlését**,
- a **külső vezérlőjelek** generálását.

A CPU működését közvetlenül támogató SFR regiszterek:

- az **akkumulátor** (ACC) a műveletek operandusz-, és célregisztere,
- a **státusz-regiszter** (PSW), a műveletek eredményétől függő feltétel bitek tárolója,
- a **stack-pointer** (SP), tárolja a stack memóriába utoljára letárolt információ címét,
- a **program számláló** (PC), tárolja a következő utasítás-bájt címét.
- a **B regiszter** , a szorzás, ill. osztásnál az egyik operandusz-, és célregiszter.

2.1.2. Belső memória

A 8031/51 típusban 128 bájt - míg a család többi tagjában 256 bájt - méretű operatív belső memória és 128 bájtos SFR memória van. (15. ábra)



15. ábra

A **belső operatív** memória - címsorrendben - a következő részekre tagozódik:

- az alsó 32 bájt (00H - 1FH) négy 8 bájtos **Regiszter Bank** - ra oszlik (BANK0, BANK1, BANK2, BANK3),
- a következő 16 bájt (20H - 2FH) **bit-címezhető** memóriaterület,
- a további címeken (30H - 7FH az 51-es típusnál, illetve 30H - FFH a többi kontroll-érnél) **általános felhasználású** memória.

Az **SFR** jelölésű memóriaterület ugyancsak a 80H - FFH címen helyezkedik el, de címzési módja különbözteti meg az ugyanezen címtartományban lévő általános felhasználású memóriától

⇒ A Bankok és használatuk

A Bankok közül mindig csak **egy** - az aktívnak választott - érhető el az **utasításban foglalt** címzés-el. A kiválasztott bank egyes regisztereit az **R0, R1, ... R7** szimbólumokkal jelöljük. Az **R0**, és **R1** használható **indirekt** címzéshez, míg a többi tárolásra, vagy **ciklusszervezéshez** alkalmazható. A teljes **BANK** terület **bármelyik** bájtja **direkt** címzéssel tetszőlegesen írható, vagy olvasható.

A Bank kiválasztása a **PSW** regiszter **RS1, RS0** jelű bitekbe írt értékkel, történik (a két bites bináris szám adja a Bank számát, pl. RS1 = 1, RS0 = 0 a BANK2 -t teszi aktívvá).

A **PSW** regiszter bitkiosztása a következő:

CY	AC	F0	RS1	RS0	OV	-	P
-----------	-----------	-----------	------------	------------	-----------	----------	----------

Az egyes **flag-ek**, **jelző**-bitek:

- **CY** (carry) az **átvitel**-, **áthozat** -bit, amely az aritmetikai műveleteknél jelzi a **túlcsordulást**,
- **AC** az alsó **négy bit túlcsordulását** jelző bit, a **BCD** átalakításnál van szerepe,
- **F0** felhasználó által használható jelző-bit,
- **RS0-RS1** határozzák meg az **aktuális** Bank -ot,
- **OV** az **aritmetikai** túlcsordulást jelzi,
- **P** az **Accumulátorban** lévő **1 páros számát** jelzi (0, ha páratlan számú 1 van).

A CY, OV, és P bitek értékétől függően program-ugrásokat lehet végrehajtani. A vonatkozó feltételes vezérlésátadó utasításokat a programozásnál fogjuk ismertetni.

⇒ **A bit-címezhető memória, és használata**

A bit-címezhető bájtok egyes bitjei külön-külön is írhatók, olvashatók, illetve értékükkel logikai műveletek végezhetők. Az egyes bitek kétféle módon címezhetők, még-pedig

- **bájt cím. bitszám** formában pl. **20.5** a **20H** című bájt **5**-ös bitje (a bit-szám 0 - 7 szám valamelyike, 0 az első, 7 a nyolcadik bitet jelenti),
- **abszolút bitcímzéssel** **00H - 7FH** címtartományban. A 16 bájtos memóriaterületen lévő 128 (16 * 8) bit számozása a legelső bájt legkisebb helyértékű bitjénél kezdődik (20.0 - 00H), majd a következő bitek így tovább, s sor a legutolsó bájt legnagyobb helyértékű bitjével fejeződik be (2F.7 - 7FH).

A bites logikai műveletvégzés lehetősége folytán a **mikrokontrollerek** nagyon előnyösen használhatók **automatizálási** (vezérlési) feladatokat megvalósító berendezések (PLC – k) építésére.

⇒ **Általános felhasználású memóriaterület**

A belső **operatív memória** 30H címtől kezdődő memóriaterületét minden megkötöttség nélkül lehet **változók** értékének tárolásra felhasználni.

A memória terület **00H - 7FH** címen elhelyezkedő **bájtjai** (a 8031/51 típusnál csak ez létezik) mind **direkt**-, mind pedig **indirekt** címmel elérhetők. A **80H - FFH** című bájtok csak indirekt címmel érhetők el, mivel ugyan ezen a címenek vannak az **SFR** terület bájtjai, amelyek kezeléséhez viszont a **direkt** címezést kell használni.

⇒ **Speciális Funkció Regiszterek (SFR)**

Az **SFR** jelölésű - a 80H - FFH című - memóriaterületen helyezkednek el a

- **CPU** regiszterek,
- programozható perifériák **vezérlő**-, és **státusz** regiszterei,
- a **Port**-ok kimeneti tárolói.

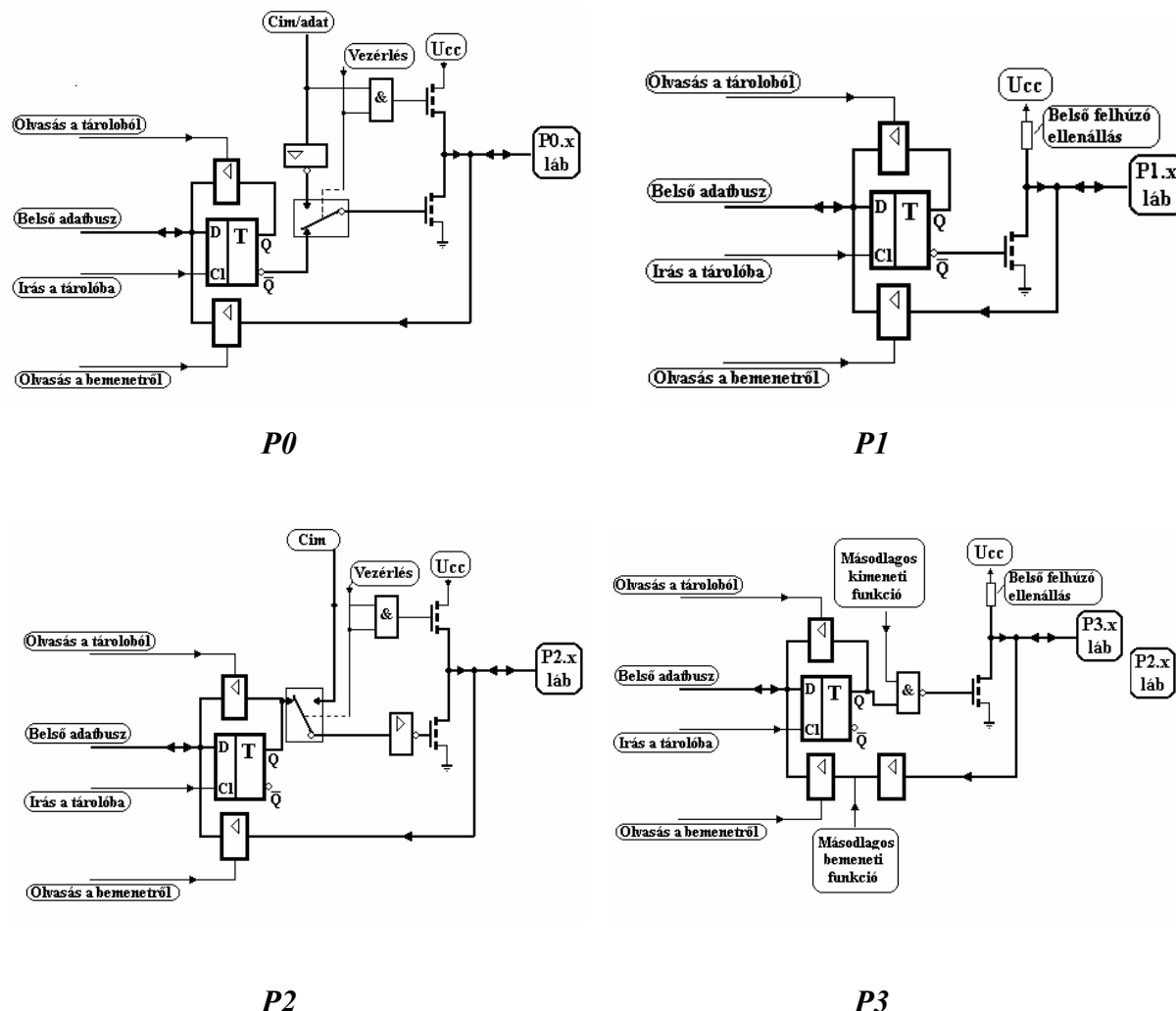
Az SFR regiszterek között vannak olyanok, amelyek **csak bájtosan**, és olyanok is amelyek **bitjei egyenként is** írhatók, olvashatók. Az utóbbi csoportba azok a regiszterek tartoznak, amelyeknek a címe 8-al osztható (pl. 80H,88H....F0H,F8H).

2.1.3. A Portok

A mikrokontrollerek és a környezet között az adatcsere a Portokon keresztül történik. A portok mind bájtosan, mind pedig bitesen kezelhetők. Mindegyik port-bit alkalmazható be-, vagy kimenetként is. A mikrokontroller családnál a portok többsége nem csupán egyszerű I/O portként használható, hanem további, úgynevezett másodlagos funkcióra is alkalmazható. A négy Port egy - egy bitjének kapcsolási rajza a 4.3. ábrán látható.

A 8031/51 kontrollernél az egyes portok **másodlagos** funkciói a következők:

- **P0** multiplexelt **CIM - ADAT** busz (AD0.... AD7),
- **P1** **nincs**
- **P2** **cím felső 8 bitje** (A8 A15),
- **P3** bitenként **különböző**:
 - soros** vonal (RxD - P3.0, TxD - P3.1)
 - külső **megszakítás bemenetek** (INT0 - P3.2, INT1 - P3.3),
 - számláló bemenetek** (T0 - P3.4, T1 - P3.5),
 - külső **memória vezérlése** (WR - P3.6, RD - P3.7).



4.3.ábra

Minden port-bit **írásakor** az adat a **D-tárolóba** kerül, amelyben tárolt érték állítja be - az illesztő áramkörön keresztül - a **kimenet állapotát**. Egyszerű **olvasáskor** a **láb**on lévő adat kerül a **belső buszra**. A **read-modify** típusú utasításokkal a tároló tartalmát lehet **visszaolvasni**

Amikor a port-bitet **bemenetként** kívánjuk felhasználni, vagy a **másodlagos** funkciót akarjuk engedélyezni, akkor a D - tárolóba **HIGH** szintet (1 - et) kell írni. A **RESET** jel hatására a mikrokontroller mindegyik portjánál a tárolókba magától **1** íródik.

2.1.4. Programozható időzítő / számláló egységek

A 8031/51 típusú mikrokontrollerben két **programozható időzítő/számláló** (T0, T1) periféria van. A két periféria felépítése **egyforma**, és három üzemmódban azonosan is használható. **El-térés** a kettő között az, hogy az "**osztott**" üzemmódra (3-as üzemmód) csak a **T0** - ás időzí-

tő/számláló programozható, míg a *soros adatátvitelt* ütemező Baud-rate generátor-ként csak a *T1* - es használható.

A *T0* egység a TH0, és a TL0 jelű, míg a *T1* pedig a TH1, és a TL1 jelű SFR regiszterekben *számlálja* a *kiválasztott* bemenet *impulzusait*. Az impulzusforrás lehet az *órajel 12*-e ($f_{OSZC}/12$), vagy egy *külső impulzussorozat*, amelyet a T0 (P3.4), vagy a T1 (P3.5) lábakra csatlakoztatunk. A számlálók *üzemmódját* a *TMOD* (Timer-Modus) nevű SFR regiszter tartalma határozza meg. A számlálók *vezérlő* bitjei, és a *túlsordulást* tároló bitek a *TCON* (Timer-Control) - bitenként is írható, olvasható - SFR regiszterben vannak.

A számláló akkor használható *digitális késleltetőnek* (időzítőnek), ha a számlált jelsorozat frekvenciája állandó. Ez a feltétel a belső – frekvencia-stabilizált - órajel kiválasztásánál teljesül, és ekkor mondjuk, hogy az egység *időzítő*. A kívülről érkező jelsorozat kiválasztásakor *esemény-számlálásról* beszélünk.

Az időzítés alapegysége a mindenkor órajel periódusidejének *12* szerese, mivel *gépi ciklusonként* növekszik (incrementálódik) a számtartalom. A számlálásnál a *külső* jel élváltásának meghatározásához *két gépi ciklus* szükséges, és ezért a legnagyobb számlálható frekvencia az órajel-frekvencia *24*-e.

⇒ *Időzítő / számlálók vezérlése*

A *T0* számláló periféria üzemmódját a TMOD regiszter *alsó négy* bitje, míg a *felső négy* bit a *T1* egységét állítja be. (Az azonos szimbólumok indexe utal a számláló számára).

- A *C/T0(1)* bit *választja* ki az *impulzus-forrást*. A bit logikai 1 érteke a leosztott belső órajelet, míg a 0 érték a külső lábra érkező jelet kapcsolja a számlálóra.
- *M10(1)* és az *M00(1)* bitekből álló bináris szám határozza meg az számláló-regiszterek működésének módját, az *üzemmód*-ját.
- A számláló *indításának*, *leállításának* módját a *GATE0(1)* kapuzó bit szabja meg. 0 értékénél *csak szoftverből*, míg egy értékénél *szoftverből, és külső jellel* (hardver módon) *is* vezérelhető a számláló.

A számlálás indítható, illetve leállítható:

- *csak szoftver* vezérlési módban a *TCON* regiszterben lévő *TR0(1)* szoftver indító bit logikai 1 szinttel *indít*, 0 - val pedig *leállít*,

- *szoftver, és hardver* vezérlésű módban, a $TR0(1)$ bit és az $INT0(1)$ lábra kívülről adott vezérlő jel együttes **1** szintje **indítja** a számlálást, míg bármelyik **0** értéke **leállítja** azt.

A **TMOD** és a **TCON** regiszterek bitjei a következők:

TMOD	T1				T0			
	G	C/T	M1	M0	G	C/T	M1	M0

TCON	TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0
-------------	------------	------------	------------	------------	------------	------------	------------	------------

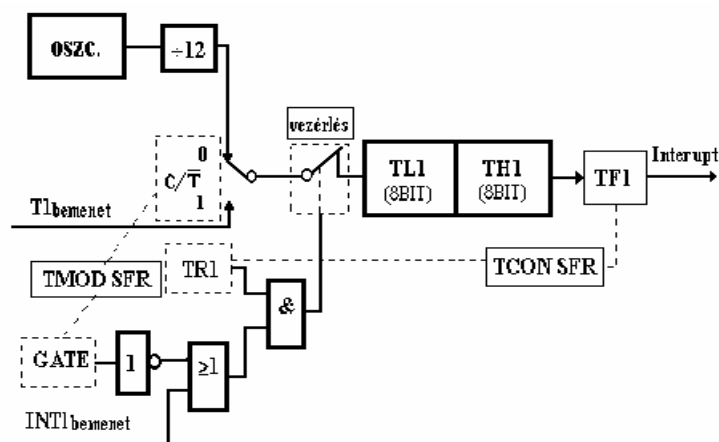
A számlálók leállítása után is megmarad a regiszterekben lévő számtartalom, és egy újra indítás után erről az értékről folytatódik a számlálás.

A számláló regiszterek tartalma programból - tetszőleges értékre - bármikor átírható.

⇒ **Időzítő / számlálók 0-ás és 1-es üzemmódjai**

Mindkét üzemmódban szabadon futó előre számlálóként működik a periféria. A különbség csupán az, hogy a 0 - ás módban a magasabb helyértékű regiszter (a **TH0(1)**) csak 5 bites, míg az 1 - es módban 8 bites. Ennek megfelelően a **0 - ás** üzemmódba állított számláló **13 bites**, míg az **1 - es** módusú pedig **16 bites**.

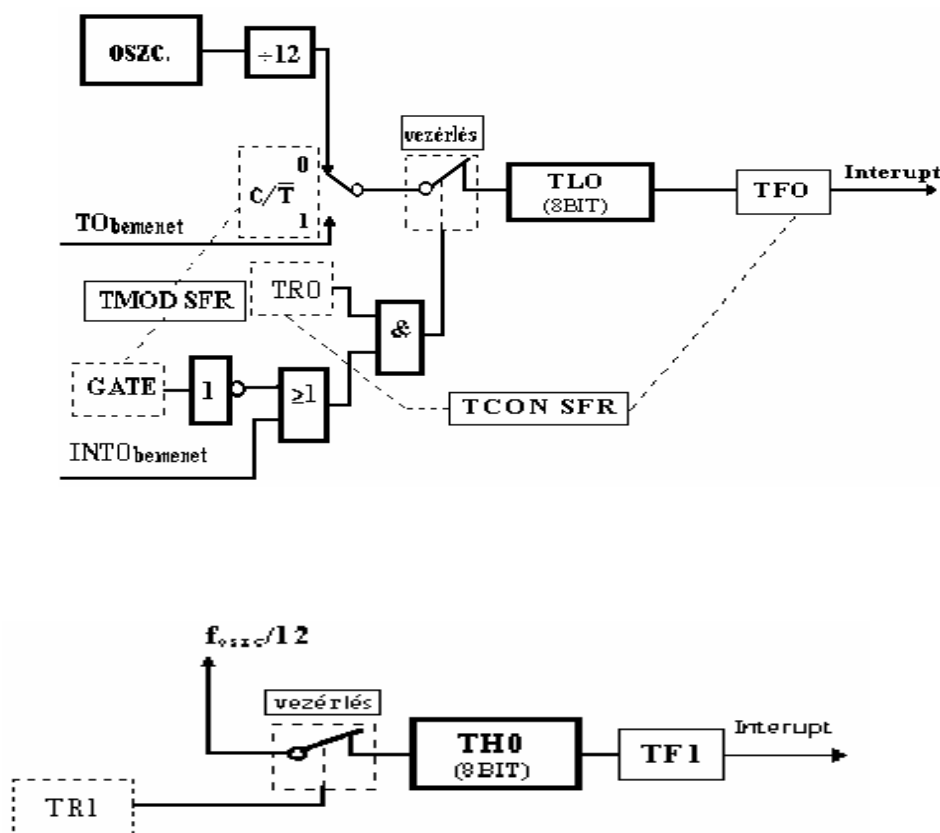
A két regiszterből álló számláló kiegészül egy túlsordulás bittel (**TF0 (1)**), amely a TCON regiszterben van. A túlsordulást követően a számlálás kezdődik előlről, és a túlsordulás bit is törlődik. A túlsordulás megszakítás-t (interrupt) is indíthat, ha az engedélyezett. Az áramköri felépítést szemlélteti a 16. ábra.



16. ábra

- A **TL0** regiszterhez csatlakoznak a **T0 vezérlő áramkörei**, és ezért e **regiszterben 8 bitesen** a leosztott **órajel (időzítés)**, vagy a **T0 lábra** érkező **impulzusok számlálása** történhet (**eseményszámlálás**).

Az áramköri felépítést szemlélteti a 18. ábra.



18. ábra

Ezt az üzemmódot akkor kell használnunk, ha szükséges a soros kommunikáció - tehát a T1 foglalt - és a feladat megoldásához szükséges két független számláló.

2.1.5. Soros vonali illesztő

A mikrokontroller beépített perifériája a soros adatátviteli illesztőegység.

A soros átvitelnél a kontroller a **TxD** jelű lábon adja a kiviendő, míg az **RxD** jelű lábon fogadja a beérkező adatokat.

Az átviteli egység **Full - duplex**, ami azt jelenti, hogy egyidejűleg mind adás, mind pedig vétel is folyhat.

A periféria üzemmódjainak vezérlő bitjei, illetve a státusz bitek az SCON jelű SFR regiszterben vannak.

Négyféle üzemmódban működhet az átviteli egység. A leggyakrabban az RS232C szabvány-nak megfelelő 3-as üzemmódot használják.

A soros - aszinkron kommunikációt a **T1 időzítő/számláló** ütemezi **2-es** üzemmódban. Az átvitel Baudban (bit/sec) mért sebességét az órajel frekvenciája és a számláló modulusa határozza meg, amely számítható is, vagy a katalógusokból kikereshető.

A kommunikáció a 19. ábra szerinti bitsorrendben **bájtanként** történik.

- **Adásszünetben** az átviteli vonalakon magas szint van.
- Egy bájt adása az egy bit-ideig tartó alacsony szintű un. **START** bittel **kezdődik**.
- Ezután kerülnek egymás után a vonalra - a legkisebb helyértékűvel kezdve - a bájt bitjei, az **ADAT - bitek**.(D0 – D7)
- Az adatok után következhet egy **paritás** - ellenőrző bit, amely programozástól függően el is maradhat.(P)
- A bájt adását egy, vagy két - magas szintű - **STOP** - bit zárja le.



19. ábra

Az adatátvitel **önszinkronozó**, mivel egy bájt átvitelét egy magasról-alacsony szintre váltó **él indítja** mindig. Adásszünetben a vonalak magas szinten vannak.

A kommunikáció csak akkor lesz hibamentes, ha az adó és a vevő **átviteli sebessége**, és **formátuma** azonos.

A 20. ábrán látható az **RS232** szabvány szerint működő soros vonali illesztő egység logikai vázlata.

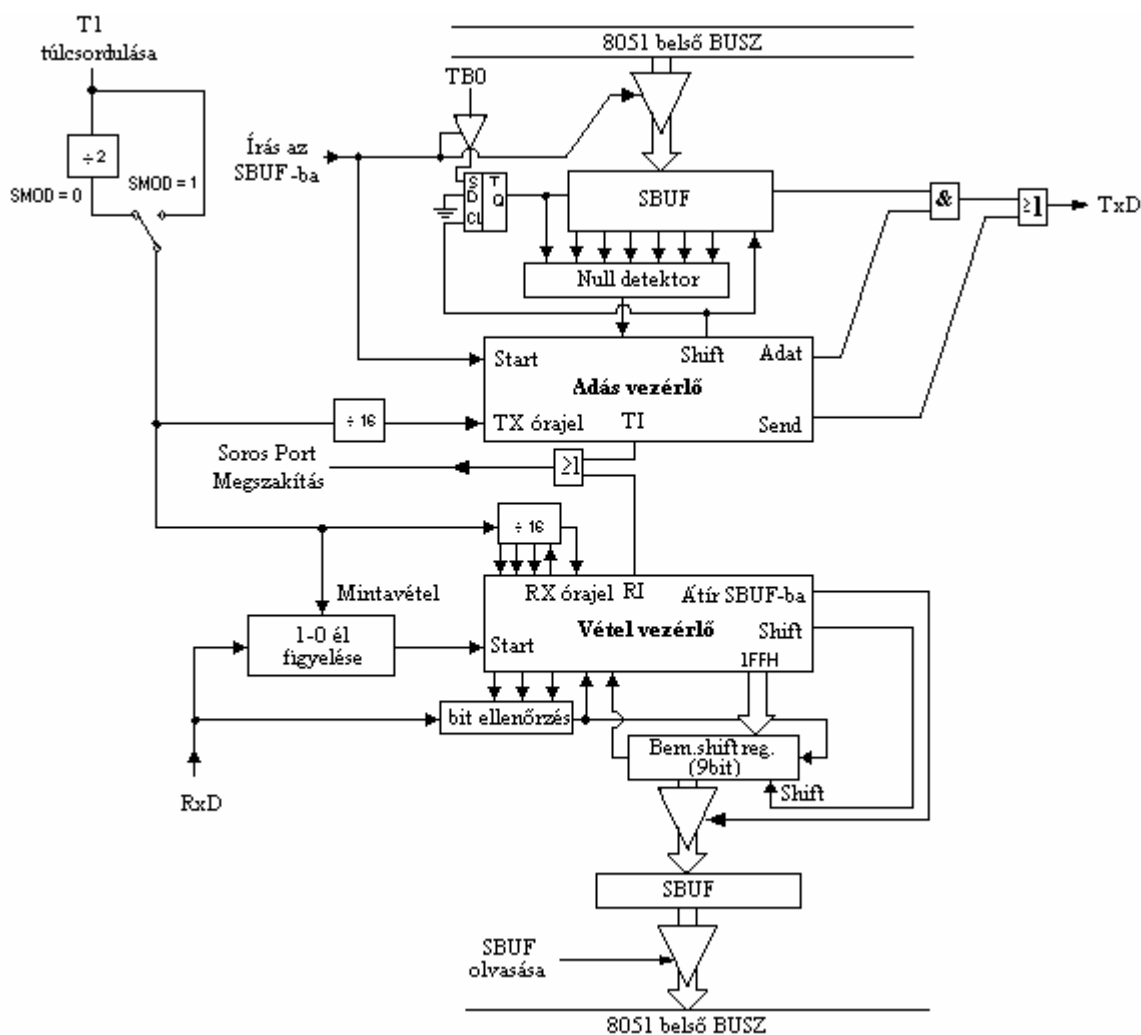
A programozható illesztő a **TxD** (Transmit =adó), és az **RxD** (Rcieve =vevő) vonalakon keresztül kommunikál a csatlakozó egységgel. E mellett még a két készülék **közös** tápfeszültség pontjait (GND) kell **összekötni**.

A fizikailag különböző vonalak lehetővé teszik az egyidejű adást, és vételt, vagyis **DUPLEX**.

A működést **ütemező** jelet a **T1** számláló **túlcserdülése** végzi. Az adás sebességét még az **SMOD** bit 0-ba állításával **felezni** lehet.

Két **SBUF** (Serial Buffer) jelű **léptető-regisztert** találunk az ábrán. Az egyik a **kiviendő** adatot, míg a másik a **vett** adatot tárolja. A felhasználói felületről nézve, **mindkettő azonos címen** van!! A különbség, hogy az **adó-regiszterbe** csak **írni**, míg a **vevő-regiszterből** csak **olvasni** lehet.

A 20. ábra alapján tömören áttekintjük a soros **adás**, és a **vétel** működését.

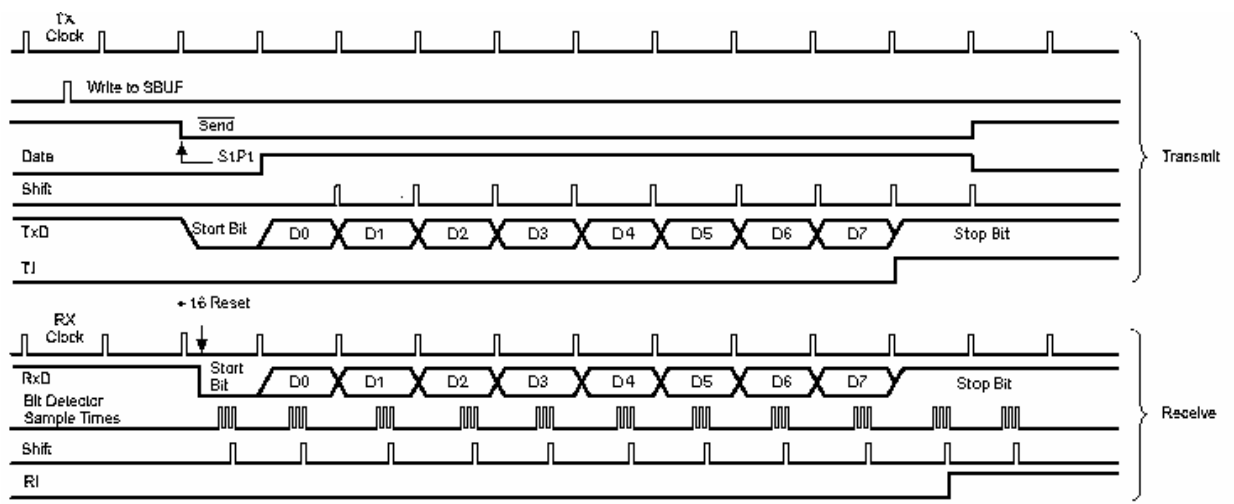


20. ábra

⇒ **Adás**

Az **adás** egy - az **SBUF** -ba irányuló - **adatátviteli** utasítással indul. Ekkor az „**Írás az SBUF** -ba” jelű bemenetre érkező vezérlő jel a belső adatbuszról az **SBUF** -ba tölti a kiviendő **adatot**, a **D** flip-flop -ba írja a **TB0** bit értékét (9. bit), valamint indítja az adás vezérlését (Start).

A további folyamat önállóan megy végbe, amely a 21. ábrán követhető. Az adás-vezérlő állítja elő a **Send**, **Data**, és **Shift** jeleket. Ezek vezérlik a TxD vonalon kilépő – a 19. ábrán vázolt felépítésű adatsort. A nyolcadik adat bit (D7) kilépése után írja 1-be a **TI** bitet, amely jelzi az információ kivitelének **végét**.



21. ábra

⇒ **Vétel**

A soros vonalon **érkező** információ az **RxD** csatlakozópontra jut. Amíg nincs érvényes adat, a vonal **magas szinten** van. A vonalon megjelenő **1 – 0 átmenet** „jelezheti” **bájt** következik.

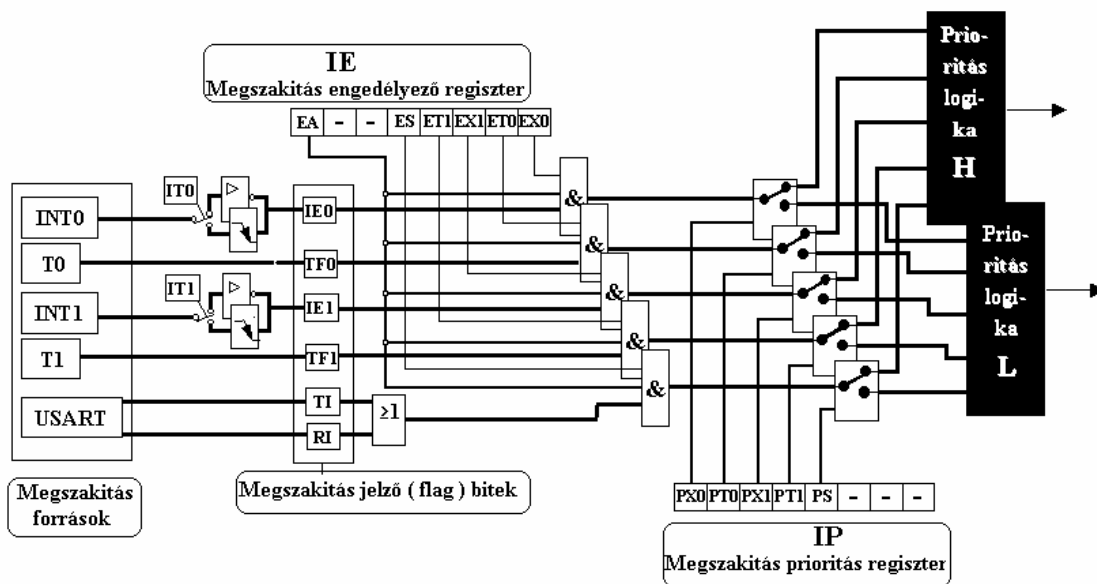
A **vétel-vezérlő** egységet az **élfigyelő** indítja a Start bemenetén, és ez a jel egyúttal törli a **16-os osztót** is. Az osztó a **háromszor** vizsgálja a bejövő jel szintjét (a bit-idő közepén). A **Start-bit** esetében ez a **0**. Amennyiben a három eredmény többsége 0 szintű, akkor valóban új adat-bájt kezdődik, tehát folytatódik az adatsor beléptetése. Amennyiben a vizsgálat nem ad 0-t, akkor egy külső zaj indított, és nem folytatódik a művelet. A nyolc adatbit beléptetése után 1-be vált az RI (Receive Interrupt) bit, amely jelzi az új információ beérkezését. Az új adat párhuzamosan kiolvasható.

2.1.6. Megszakítások

A mikrokontroller megszakításainak segítségével oldhatók meg az események **valós idejű** kezelése. A program megszakítását kérheti :

- külső esemény az INT0, ill. az INT1 lábakra adott jellel,
- a T0, vagy a T1 számláló túlsordulása,
- a soros adatátvitel.

A felsorolt források mindegyikéhez tartozik egy megszakítás kérő (interrupt request) bit, amely 1-be írása kezdeményezheti a megszakítást. A megszakítások mindegyikéhez tartozik a programmemória egy címe.



22. ábra

Mindegyik megszakítás elfogadása egyedileg engedélyezhető, vagy tiltható az IEN (Interrupt Enable) SFR regiszter megfelelő bitjeivel (1 enged, 0 tilt). Ugyanezen regiszter legmagasabb helyértékű (MSB) bitje általános engedélyező bit. Ha ez 0 akkor mindegyik megszakítás elfogadása tiltott, amikor 1 értékű, csak akkor hatásosak az egyedi engedélyezések.

A megszakításforrások között egy prioritási sorrend van, ami azt jelenti, hogy egyidejű megszakításkéréskor a magasabb prioritásút szolgálja ki a CPU. A prioritási sorrend korlátozottan módosítható az IP (Interrupt Priority) regiszter egyes bitjeivel.

⇒ **Megszakítások kiszolgálása**

A megszakítás elfogadásakor a Stack -be a visszatérési cím a PC programszámlálóba a megszakításforráshoz tartozó cím (fix című megszakítás kiszolgálás) íródik, és a CPU e címmel kezdődő szubrutin hívást végez. Ezért ezekre a címekre olyan vezérlésátadó utasítást kell írni, amely a megszakítást kiszolgáló rutin kezdőcímére ugatja a program-folytatást.

A megszakítás elfogadása után a CPU letiltja a további megszakítás engedélyezést, ezért a kiszolgáló rutint a RETI utasítással kell befejezni, aminek hatására töltődik vissza a Stack -ből a PC-be a visszatérési cím, és az általános megszakítás-elfogadás engedélyeződik.

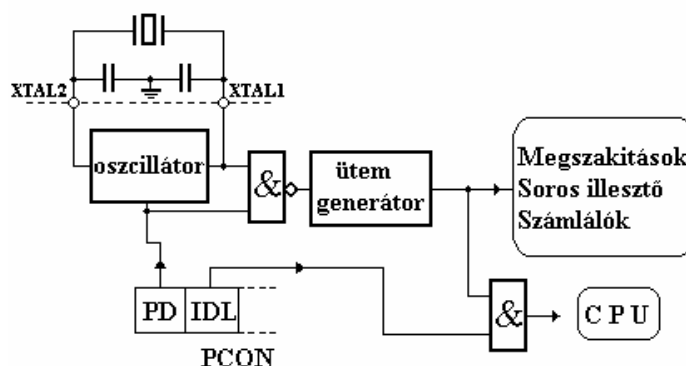
A külső megszakítások kezdeményező jelét legalább két gépi ciklus idejéig kell fent tartani, de a kiszolgáló rutin vége előtt meg kell szüntetni.

A megszakításkérő bitek közül a külső megszakításokhoz tartozók a rutinba ugrás után törölődnek. A számláló túlsordulások, - ha az időzítéses módban üzemelnek - ugyancsak törölődnek. Egyébként szoftver úton kell törölni a megszakítás rutinban.

2.1.7. Teljesítmény felvétel vezérlése

A mikrokontroller teljesítmény felvétele programból állíthat. A két lehetséges üzemmód

- Power Dawn ,
- És az Idle.



23. ábra

2.2. Külső memóriák illesztése

A mikrokontrollerhez **külső program**-, és **adatmemória** is csatlakoztatható. A külső memóriák címzése, és adatforgalma a **P0**, és **P2** portokon keresztül történik.

- A **P2** port adja a címbuszra a **felső** nyolc bitet (A8 ... A15).
- A **P0** porton jelenik meg - **időosztással** - először a cím **alsó** nyolc bitje (A0 ... A7), majd az **adatbusz** (D0 ... D7) funkcióját látja el. A szétválasztáshoz külső tárolót (EXL) kell alkalmazni, amelybe az **ALE** (Address Latch Enable) vezérlőjel lefutó éle írja be a címbiteket.

⇒ **Adat-, és programmemória kezelése**

A **16 bites** címmel **64 Kb**ájt memória érhető el. Ez a memória - kapacitás megduplázódik azal, hogy a programmemóriából történő olvasást - a címkidást követően - a **PSEN** (Program

Strobe Enable) jel, míg az adatmemóriából az olvasást az **RD** (Read), az írást pedig a **WR** (Write) jel vezérli.

Azoknál a kontrollereknél, amelyekben van **belső** programmemória is (pl. a 8051 típus) a belső és **külső** memória **együtt** lehet **64 Kbájt**, és a külső memória kezdő címe a belső után következik.

De mód van arra is, hogy **csak** külső memóriát használjunk. Ehhez az **EA** jelű szelektáló bemenetre **0** szintet kell adni. Ilyenkor a külső programmemória kezdő címe 0000H.

⇒ **Külső perifériák csatlakoztatása**

A mikrokontroller családnál - a mikroprocesszoroktól eltérően - nincs külön perifériát író/olvasó utasítás.

A perifériákat illesztő áramköröket a külső **adatmemória** részeként kell megvalósítani (memóriába ágyazott periféria).

A **bemeneti** - input - egységek jeleit - szintillesztő áramkörökön keresztül - egy **busz-meghajtó** áramkör (pl. 74LS541) bemeneteihez kell csatlakoztatni. A meghajtó kimenetei az adatbuszhoz csatlakoznak. Az áramkör kiválasztásához cím-dekódolót (pl. 74LS138) kell használni, amely az illesztőhöz rendelt címnél előkészíti az átvitel engedélyezést, amely a **RD** jel hatására következik be. Ekkor a külső periféria adatbitjei a kontroller **akkumulátorába** kerülnek.

Kimeneti - output - egységhez kimeneti tárolót kell alkalmazni (pl. 74LS574), amely kiválasztása - a bemenethez hasonlóan – cím - dekódolással történik, és az akkumulátorból kiírandó adat a tárolóba a **WR** jel hatására íródik be. A tároló kimenetei **szint-**, és **teljesítményillesztő** áramkörökön keresztül hajthatják meg a vezérelt egységeket.