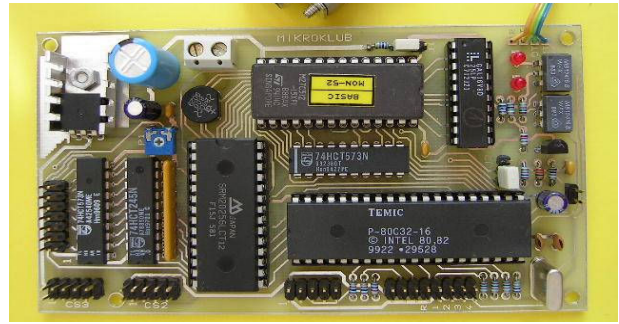
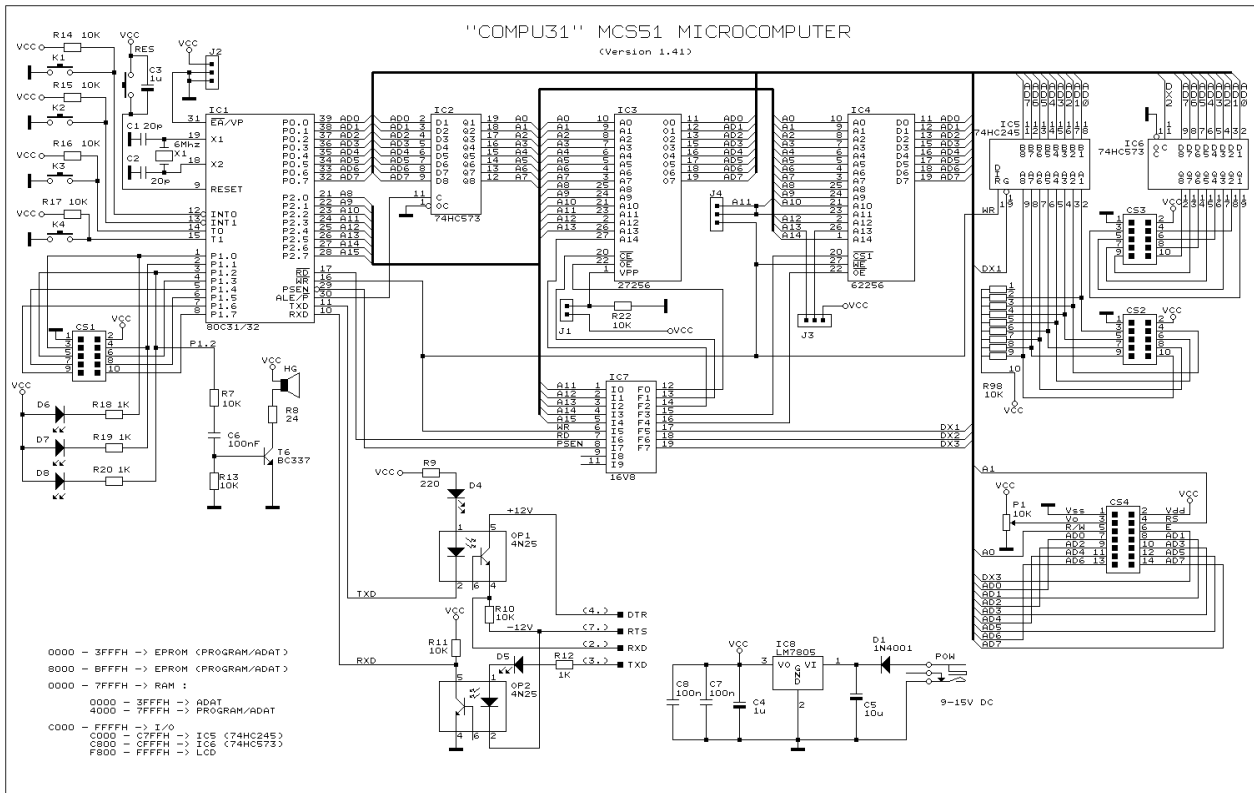


## Mikrogép 8031/8032 mikroprocesszorral (COMPU31)

Manapság szinte minden korszerű elektronikus készülékben mikroprocesszor, mikrokontroller működik. A mikrokontroller elnevezést az INTEL vezette be, ami alatt olyan áramköri egységet ért, amelybe a mikroprocesszor és néhány periféria funkciót is egybe integráltak. Az MCS51-es család tagjai 1980-ban születtek, és jelenleg a legelterjedtebb típusokká váltak. A kényelmes programfejlesztést segíti az ismertetésre kerülő RAM memóriával is ellátott, IBM PC-vel a soros porton keresztül összeköthető mikrogép is, amelybe a PC-n fejlesztett gépi kódú, vagy 8052-AH-BASIC program letölthető, és azonnal futtatható.



Az áramkör központi egysége a 8031 vagy 8032-es processzor. Ha a BASIC programozásra is igényt tartunk, akkor 8032-es processzort kell használnunk. A tápegységen kívül az alaplaphoz került egy 2764, 27128, vagy 27256-os EPROM, egy 6116, 6264, vagy 62256-os RAM, egy 8 bites kimeneti, és egy 8 bites kétirányú I/O port, valamint a PC-MIKRO kapcsolatot támogató RS232 interfész. A portok csatlakozókra vannak kivezetve, azokra különböző perifériák, billentyű, LED vagy intelligens LCD kijelző, stb. csatlakoztatható.



A készülék memóriakiosztása megegyezik az ELEKTOR-ban megjelent 8031/8032-es egy paneles számítógépével, így az arra írt példaprogramok, monitor és terminál programok szinte változtatás nélkül alkalmazhatóak.

A software rendszer az IBM PC-n futó terminál, és a mikrogépen futó monitor vagy BASIC programokból áll. Magát a fejlesztendő programot az IBM PC-n írjuk meg, és BASIC parancs sorokat, illetve assembly esetén a fordítás utáni a tárgykódot egy terminál program segítségével az RS232-es porton keresztül küldjük a mikrogép memóriájába.

A rendszer működéséhez szükséges órajelet egy kvarcoszcillátor biztosítja. A C1 és C2 kondenzátor a rezgés biztos beindulásához szükségesek, értékük 10-30 Pf közé eshet. A soros adatátvitel működéséhez szükséges BAUD RATE generátort, valamint az időméréssel kapcsolatos alkalmazásokban a szükséges másodperces, perces, órás stb. időalapokat a processzor belső órajelének - ami a kvarcfrekvencia tizenketted része - leosztásából kapjuk. Ezt az osztást a program végzi, a processzorba épített timer áramkörök segítségével.

Az epromba égetett assembler monitor program 6 Mhz-es frekvenciát vesz alapul. (Használhatunk 12 Mhz-es kvarcot is, ez esetben a soros adatátvitel baud sebességét is meg kell duplázni.)

A C3-as kondenzátor feladata, hogy bekapcsoláskor egy reset impulzus képződjön. A kondenzátorral párhuzamosan beépíthetünk egy reset gombot is.

A 8031 és 8032 típusban nincs beépített programtároló, így a külső ROM/EPROM csatlakoztatása elkerülhetetlen. Mivel a programszámláló (PC) 16 bites, a bővítés 64 Kbájtig végezhető Minden utasításelhozó ciklusban a tárcím alsó 8 bitje a kétirányú P0 portán (P0-7), a felső 8 bitje pedig a 2-es porta 0-7 bitjén jelenik meg (P2.0-7). Az ALE jelzésű kimenet jelzi hogy a cím érvényes, a címbitek eltárolhatók. A nyolc alsó címbitet az ALE jel írja az IC2-es latch regiszterbe, és azt a következő beírásig tárolja. A cím kiküldése után a processzor beolvassa a P0-on megjelenő adatot.

A tárolómemória felosztását, valamint a perifériák engedélyező jeleinek előállítását az A11-A15 címvonalak, valamint a RD, WR, és PSEN felhasználásával az IC7-es GAL állítja elő. Tekintve hogy a bemenő jelekből a GAL-ba égetett logikai hálózat segítségével szinte tetszőleges számú, és variációjú címzési rendszer alakítható ki, a következő címkiosztások is csak egy-egy lehetséges példák. Ha a mikrogépen assembly és BASIC programokat is akarunk fejleszteni, valamint az ELEKTOR-ban megjelent assembly példaprogramokat, monitor programot is futtatni akarjuk, akkor a memóriafelosztás a következő legyen: az EPROM (itt feltételezzük hogy 27256-os EPROM-ot használunk) tárolókapacitása két 16 Kbájtos részre van osztva, az egyik a 0000-3FFFH, a másik a 8000-BFFFH terület. A processzor az alsó 16 Kbájtot csak a PSEN jel aktivizálásával tudja olvasni. (A PSEN jel aktív, azaz alacsony szintű, ha a processzor a külső memóriából akar utasítást vagy adatot beolvasni.) A felső 16 Kbájtot a PSEN, de a RD jel aktivizálásával is olvashatja, ide tehát program, de adat is kerülhet. A RAM terület (62256-os, 32 Kbájtos RAM-ot használva) a 0000-7FFFH tartományba esik. Mint látható a RAM és az EPROM terület a

0000-3FFFFH tartományban fedi egymást, a PSEN jel nulla szintjénél a programtárolóhoz, (EPROM), magas szintje esetében az adatmemóriához (RAM) szól a kontroller. A RAM terület 4000-7FFFFH tartománya egyaránt helyet adhat programnak vagy adatnak, itt ugyanis a RAM és EPROM terület nem fedi egymást, a processzor a PSEN, és a RD jel segítségével is olvashatja. Ebbe a 16 Kbájtos területbe kerülhetnek a tesztelésre váró gépi kódú programok.

Szintén a GAL végzi a külső I/O eszközök címzését. Az IC-be égetett logikai hálózat a C000-FFFFH 16 Kbájtos területet nyolc részre osztja az A11-15 címvonalak felhasználásával. A címkiosztás a következő: a C000-C7FFh az IC5, a C800-CFFFh az IC6, az F800-FFFFh az LCD modul elérési címe.

A J1 jumper zárásával az EPROM 1-es lábára kapcsolhatunk öt voltot, ha pedig a jumper nyitott, azt az R22 alacsony szintre húzza. Ennek akkor van jelentősége ha 27512-es EPROM-ot használunk, ahol ez az A15-ös címbemenet. Attól függően, hogy ide alacsony vagy magas szintet kötünk, a 27512-es alsó vagy felső 32K területe érvényesül. Így lehetőség van pl. a gépi kódú programfuttatását segítő monitor, vagy a soros vonali terminállal működő BASIC rendszer közti választásra. A jumper váltása azonos hatást vált ki, mintha két 27256-os EPROM-ot csere-berélnénk a foglalatban. Fontos! Ha 2764, 27128, vagy 27256-os epromot használunk akkor R22 elhagyható, a J1 pedig rövidzárral helyettesítendő, hogy az EPROM egyes lábára (Vpp) 5 volt kerüljön.

Ha csak assembly nyelven akarunk programot írni, és letölteni, valamint megelégszünk nyolc Kbyte RAM és EPROM memóriával - ez az esetek túlnyomó részében bőven elég - akkor az EPROM memóriát a 0000-01FFFFh, a PSEN, és a RD jel aktivizálásával is olvasható RAM memóriát a 4000-5FFFFh címre helyezzük. (Azaz a GAL-ba ezt megvalósító logikai hálózat kerül.)

A mikroszámítógép a külvilággal a be- és kimeneti csatlakozási pontokon keresztül tart kapcsolatot. A P0 és a P2 portokat lefoglalja a külső EPROM olvasása. A processzor P1.0-7, valamint az IC6-ből kialakított kimeneti, és az IC5-ből kialakított kétirányú bővítő portokat használhatjuk a külső I/O eszközök, pl. a kijelző, és a billentyűk meghajtására.

Az IC5 adatot olvashat be, vagy az adatbusz tartalmát jeleníti meg az A1-8 ki/bemeneti pontjain, attól függően hogy olvasás vagy írás műveletet hajtunk végre a C000-C7FFH címtartományban. Az IC6 mint 8 bites kimeneti port használható. Ha a C800-CFFFH területre adatot írunk, az e 8 bites regiszterbe tárolódik, és a Q1-8 kimeneten megjelenik.

#### Az LCD modul csatlakoztatása.

Az intelligens, programozható LCD kijelzők csatlakoztatását segíti az adatbusz és vezérlőjelek kivezetése. Az LCD kijelző modul tulajdonképpen egy komplett, RAM memóriával is ellátott mikroszámítógép, amely egy külső eszközzel, - jelen esetben a mikrogéppel - párhuzamos adatforgalmat tud bonyolítani. Röviden a programozható LCD modulokról :

Az alkalmazott cél mikroprocesszor típusa leggyakrabban HD44780. (Ezekről a kijelzőkről egy részletes leírás jelent meg az 1994-es Rádiótechnika évkönyvben.) A legtöbb egy vagy két soros LCD modul egyszerűen az adatbuszra köthető. Az LCD felprogramozásához szükséges utasításokat, adatokat az LCD adatbuszára kell juttatni, azt az engedélyező E (enable, azaz engedélyezés) bemenetre adott pozitív impulzussal bekapuzni. Az, hogy a küldött adat az utasítás vagy az adat regiszterbe kerül, az RS kivezetés (register select, itt adat vagy utasítás regiszter választást jelent) állapotától függ. Az LCD adatregiszterei olvashatóak, az R/W bemenetre ez esetben magas, íráskor alacsony szintet kell adni. (Ha az LCD-t csak írni akarjuk, ezt a kivezetést egyszerűen a földpontra köthetjük.) Az AD0-7 a mikrogép adatbuszára, az E (enable, azaz engedélyezés) bemenet a címdekódolóra (a GAL-ra), az RS kivezetés a processzor A1, az R/W az A0 címvonalra csatlakozik. A GAL-ban az F800-FFFFH terület van kijelölve az LCD-nek, ide írva tudjuk a modult felprogramozni, adatokkal feltölteni. Az előbbiekből kiderül, hogy az írás és olvasás, valamint az adat és utasítás regiszter között az eltérő elérési címmel tudunk választani. Az F800h címen az LCD utasítás regisztere írható, az F801h címen a címszámláló olvasható, az F802h címen az adat regiszter írható, F803h-on olvasható. A P1 potméter a kontraszt beállításra szolgál.

Megjegyzendő, hogy a legjobb olvashatóságot néhány típusnál úgy kapjuk, ha ezt a bemenetet egyszerűen a +5 voltra kötjük. (A CS4 túsoros csatlakozón a kivezetések sorrendje megegyezik a leggyakrabban használt és kapható LCD modulok csatlakozópont kiosztásával.)

A processzor INT0, INT1 kivezetései megszakításkérő, a T0, T1-es időzítő/számláló bemeneteknek is használhatók. Ide különböző I/O eszközök, jeladók, vagy pl. (ahogy a kapcsolási rajzon is látható) billentyűk csatlakozhatnak. Valamennyi előbb felsorolt port a tápfeszültséggel együtt kétsoros "tűskés" csatlakozó aljzatokra van kiosztva. (CS1, CS2, CS3) Ezekre szalagkábel csatlakozó dugható.

A P1.0-3 portokkal egy-egy LED diódát is meghajthatunk.

A K1-K4 billentyűkre és a RESET gombra csatlakozó vezetékeket, valamint a D6-8 LED-eket vagy közvetlenül forrasszuk az alappanel forrpontjaiba, vagy azokat felrakhatjuk egy külön kis panelre, amit az alaplappal egy 16 pólusú túsoros csatlakozóval köthetünk össze.

A soros adatátvitel az OP 1-2 optocsatolókon keresztül valósul meg. Az optók egyrészt galvanikusan leválasztják a mikrogépet az IBM PC-ről, másrészt a soros átvitelhez használt plusz-mínusz 12 voltos feszültség és az 5 voltos TTL szintek közti szintátvitelt is megoldják mindkét irányban.

Mivel a mikrogép tápegysége nem ad ki plusz-mínusz 12 voltos feszültséget, azt magáról az átvitelben résztvevő PC-ről vesszük le, úgy hogy a két kimenő RS232-es vezérlő vonalat (DTR, RTS) plusz illetve mínusz 12 voltra húzzuk. (Ezt a PC-n futó - Mikroklub által fejlesztett - terminál programok elvégzik) A D4 LED az adást, míg a D5 a vételt jelzi. Az összeköttetéshez egy négyeres szalagkábelt használunk, amit a következők szerint kell bekötni:

Az OP1 4. lábán jelenik meg a mikrogép által adott jel, ezt a PC RXD bemenetére, (2.), míg a PC TXD kimenetéről (3.) érkező jel az R12 ellenálláson és D5 LED-en keresztül az OP2 1-es lábára kerüljön. A -12 voltot az RTS (7.) a +12 voltot a DTR (4.) kivezetésről vehetjük le. (A zárójelben levő számok a PC-n található szabvány 9 pólusú RS232 csatlakozó aljzatának kivezetéseit jelentik. A 25 pólusú csatlakozón az RXD a 3., a TXD a 2., az RTS a 4., a DTR pedig a 20. kivezetés !)

Ha a kábelezést jól végeztük, akkor a terminál program indítása után a D5 LED felvillanásai jelzik a PC adását, akkor is ha a mikrogép nincs bekapcsolva. Ha pedig a mikrogép be van kapcsolva, és a PC adásaira válaszol, akkor azt jelzi a D4 felvillanása.

Figyelem ! A PC-re csatlakozó vezetékek bekötésénél figyelmesen dolgozzunk, a vezetékek felcserélése szerencsétlen esetben meghibásodást okozhat a soros port áramkörében.

A P1.2 port kivezetése vezérli a T6-os tranzisztort. (A BASIC rendszer "akusztikus kimenete" a P1 port 2. kivezetése.)

A tápegység egyszerű felépítésű. A D1 graetz-el egyenirányított, és C5-el megszürt feszültséget egy 7805-ös áramkör stabilizálja. A kisfeszültség előállítására megfelelő pl. egy csengőreduktor. Figyeljünk a 7805 melegedésére, ha szükséges, szereljünk rá hűtőszárlót. Ha hálózati transzformátornak csengőreduktort alkalmazunk, akkor próbáljuk meg az 5 voltos kapcsolokról levenni a váltó feszültséget, és ha az IC-k megkapják az 5 voltot, akkor a stabilizátor IC-re nem kell hűtőborda, csavarozzuk a panelra.

#### Néhány gyakorlati tanács a készülék összeépítéséhez :

Ellenőrizzük le a panelt olyan szempontból, hogy az egymáshoz közel eső fólia csíkok közt nincs-e rövidzár. (Főleg a két IC láb között elmenő vezetésekre kell figyelni.) Ezt a műveletet az alkatrészek beforrasztása után megismételhetjük. Az IC-eket lehetőleg rakjuk foglalatba. Egy esetleges későbbi javításnál ez a többletköltség bőven megtérül. Az IC-k 1-es lába, valamint a polaritásfüggő alkatrészek pozitív sarkának forrpontra szögletes. Ha mindent rendben találunk, kapcsoljuk be a készüléket, és mérjük le az IC-k tápfeszültségét. (5 volt +/- 2-3 tized voltot kell lennie.) Az epromba írt monitor program a P1.0 porttal vezérelt D6 LED-et kb. egy másodperces ütemben ki-be kapcsolja, jelezve a program futását.

#### Az assembler/BASIC program fejlesztő rendszerek

Az assembler és a BASIC programozást támogató programokról egy-egy külön leírás szól, az AS51.TXT, valamint az AHBASIC.TXT fájlokban.

Az "ELEKTOR" magazin az 1991/3 számában közölte a 8032-es mikrogépét. (Valamint a későbbi számokban különböző bővítéseket) A mikrogéphez készítettek egy monitor programot, valamint példaprogramokat. Ezen programok használatáról szól az ELEKTOR.TXT

Mint arról már volt szó, a GAL-ba eltérő címzési rendszerek hozhatók létre, íme néhány a lehetséges változatokból, megjelölve a lehetséges felhasználói területet:

G1 : (32K EPROM, 32K RAM Assembly/BASIC/"ELEKTOR" fejlesztő rendszerhez)

0000-3FFFh -> EPROM (Program/adat)  
 8000-BFFFh -> EPROM (Program/adat)  
 0000-3FFFh -> RAM (Adat)  
 4000-7FFFh -> RAM (Program/adat)

G2 : (32K EPROM, 32K RAM, Assembly programokhoz)

0000-7FFFh -> EPROM (Program/adat)  
 0000-7FFFh -> RAM (Adat)

G3 : (8K EPROM, 8K RAM, Assembly fejlesztő rendszerhez)

0000-1FFFh -> EPROM (Program/adat)  
 4000-5FFFh -> RAM (Program/adat)

G4 : (8K EPROM, 8K RAM, Assembly programokhoz)

0000-1FFFh -> EPROM (Program/adat)  
 0000-1FFFh -> RAM (Adat)

G5 : (32K EPROM, 8K RAM, BASIC/Assembly programokhoz)

0000-3FFFh -> EPROM (Program/adat)  
 8000-BFFFh -> EPROM (Program/adat)  
 0000-1FFFh -> RAM (Adat)

Az I/O eszközök elérési címe mindegyik változatban azonos:

C000-C7FFh -> IC5 (74LS245)  
 C800-CFFFh -> IC6 (74LS573)  
 F800-FFFFh -> LCD

Az előbbi tartalommal beégetett GAL-ra a G1-5 azonosítóval hivatkozhatunk.

#### A "perifériák" :

Szinte valamennyi alkalmazói programnak szüksége van valamilyen külső eszközre, (perifériára), ami alkalmas program változóinak megjelenítésére, módosítására, a program eseményeinek visszajelzésére, stb. A legegyszerűbb periféria eszközök - mint pl. egy LED - is informálhat minket a program futásáról, működhet mint hibajelzés, üzemállapot jelzés, stb.

Csak felsorolva azokat az eszközöket - kisebb áramköröket - amelyek a mikrogép tükresoros csatlakozóira dughatók:

- LED sor: Nyolc LED mutatja egy kimeneti port állapotát. Az egyik legegyszerűbb, mégis egyik leghasznosabb periféria áramkör.

- LED sor DIP kapcsolóval: Itt is nyolc LED mutatja egy kimeneti port állapotát, de a LED-ek egy-egy kapcsolóval rá illetve lekapcsolhatóak az adott portra.

- 8 kapcsoló: egy nyolcas "DIP kapcsolóval" tudunk magas vagy alacsony szintet adni egy bemenetre. Nagyon egyszerű, nagyon hasznos.

- 4 nyomógomb, 3 LED, reset gomb panel: a COMPU31/34 panelhoz készült, nyomógombokat és LED-eket tartalmazó panel.

- LCD, 4 nyomógomb, 3 LED, reset gomb panel: a COMPU31/34 panelhoz készült, LCD-t, nyomógombokat és LED-eket tartalmazó panel.

- mátrix billentyű panel: 12 billentyű mátrixba kötve.

- 4 szegmens LED kijelző : négy számjegyes kijelző, ahol a számok multiplex üzemmódban jeleníthetők meg.

Az alábbi áramkörök önállóan is működőképes, mikrokontrolleres kapcsolások, de mint periféria is használható más mikrokontrolleres panelokhoz, pl. a COMPU31-hez.

- Infra bemenet : Az áramkör egy TV/Video/Műholdvevő távirányítójának jelét fogadja, dekódolja, a lenyomott billentyű kódja egy tűskesoros csatlakozóról levehető, így könnyen illeszthető egy mikrokontrolleres áramkörhöz. Az áramkör programozható, azaz a 0-9, számjegy, a LE/FEL, és a "standby" billentyűk funkciója osztható ki. tehát helyettesíthető vele egy billentyűzet Az áramkörrel egy külön leírás szól.

- Hang kimenet : Az áramkör az ISD cég hangtároló memória chipjeire épül, kiegészülve egy ATMEL 90s2313-al. A tárolt hangminták közül egy billentyűzettel, vagy TTL jellel választhatunk, így könnyen illeszthető egy mikrokontrolleres áramkörhöz. Az áramkörrel egy külön leírás szól.

A COMPU31-es panelnak elkészült két további változata is. A COMPU34 panelra még rákerült egy soros EEPROM, egy analóg bemenet, és két kapcsoló relé. A COMPU35 panel is hasonló a COMPU31-hez, egy soros EEPROM-al és három kapcsoló relével lett bővítve (Ezekről külön leírás szól.)

### Az assembler program fejlesztő rendszer

Az assembly programok betöltését, futtatását lehetővé tevő, úgynevezett monitor program, a 27c512-es EPROM felső 32K-s részébe van égetve, tehát az assembly programfejlesztéskor a J1 jumper legyen zárva. (A J1 jumperről korábban volt szó.)

A Mikroklubbos fejlesztőrendszer MIKRO.EXE <file> módon indítható, ahol a file az éppen fejleszteni kívánt assembler program. Pl. a teszt.as program esetében ezt írjuk a parancssorba: mikro teszt.as



(A file kiterjesztése AS legyen, pl. TESZT.AS a mikrogépet tesztelő program neve.) A MIKRO program almenüi a FILE, az EDIT, FORDITAS, EPROM, és TERMINAL. A menüpontokból hívhatók a szövegszerkesztő, az assembler fordító, epromégető programok, amelyek lehetővé teszik, a tetszőleges szövegszerkesztővel megírt, assembly nyelvű program írását, lefordítását, végül a most ismertetett mikrogépbe töltését. (Az eprom égetés, és emulátor funkciókhoz persze kell a mikroklubbos eprom égető, vagy az emulátor.)

A program „DOS-os” – de windowsból is megpróbálhatjuk a futtatást, legalábbis egy 2.4 GHz-es, XP-s gépen sikeresen próbáltam, a parancssorból indítva. Ha a XP alól nem sikerül a futtatás, próbáljuk meg egy DOS rendszerlemezről indítani a gépet. Ezzel biztosan kizárjuk, hogy a windows, vagy egy – most felesleges – meghajtó program befészkelje magát. (Nem kell rendszerlemez csinálnunk, a működtető programot tartalmazó lemez egyben egy DOS rendszerlemez is. Csak be kell tenni a floppyba, bekapcsolni a PC-t, és a floppyról „boot-olni”)

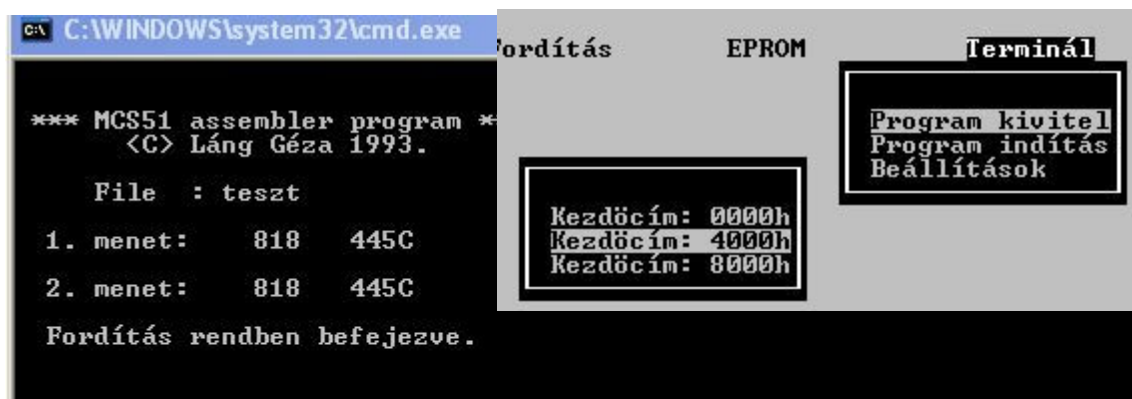


Egy példa, fordítsuk le, és töltsük be, a mellékelt TESZT.AS programot:

- Indítsuk a programot, azaz gépeljük be: mikro teszt.as

- Fordítsuk le a programot, azaz lépünk a „Fordítás” menüre a kurzorral

Ha  
hiba  
nélkül



sikerült a programot lefordítani a program tárgykódjainak RAM-ba töltése a TERMINÁL menü meghívásával lehetséges. A "Program kivitel" almenü meghívása után a kezdődik a betöltés. Az előbb leírt memória kiosztás szerint a fejlesztett program mérete maximum 16 Kb-át lehet, és a 4000-7FFFH RAM tartományba kerül, ide töltsük.

(Azt, hogy program kezdőcíme a 4000H lesz, a fordítóprogrammal is közölni kell, az assembly programban az ORG \$4000 utasítással. Szóval a 0000H helyett a 4000H címre kerüljön a RESET rutin, 4003h-ra az INTO megszakítás kezelő rutin, és így tovább, azaz a program belépési címeit 4000H-val el kell tolni.)

Mivel a teljes 16 Kb-átos terület feltöltése viszonylag sok időt venne igénybe, a PC program az "üres" azaz tisztán FF tartalmú területet nem küldi át.

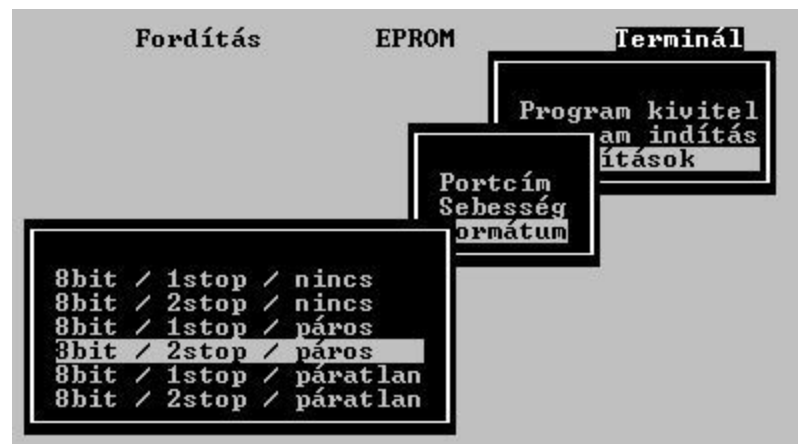
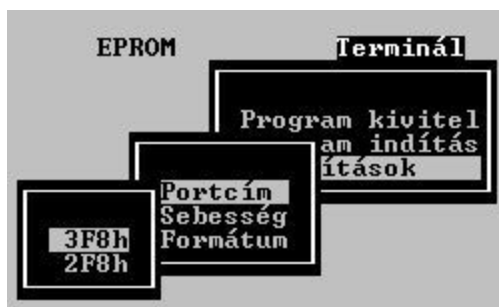


A PC és a mikrogép monitor programja az átvitel hibamentességét ellenőrzi (Paritás és "check summa" ellenőrzés.)

- Végül pedig indítsuk a mikroba betöltött programot a PC-ről, kattintsunk a „Program indítás” menüpontra! A mikrogép monitorja a vezérlést a 4000H címre adja, esetünkben elkezd futni a tesztprogi. (A monitor program a mikro következő bekapcsolása, vagy a reset gomb megnyomásakor jut majd szerephez.)

A MIKRO programban a soros adatátvitel sebessége és formátuma, valamint a port címe konfigurálható. A mikrogepen futó monitor program, 6 megás kvarccal 2400 baud-os sebességet, (12 Mhz-es kristály esetén 4800 baud!) 8 bites adatot, 2 stop bitet, és páros paritást vesz alapul. Tehát most csak ezt a beállítást használjuk.

Attól függően hogy melyik soros portot akarjuk használni, válasszuk a 3F8 (COM1), vagy a 2F8 (COM2) portcímet.



A mikrogepre csatlakoztatott a LED-ek a fenti folyamatot a következő módon jelzik:

- D6 LED villog, D7, D8 világít : mikrogep kész a program fogadására, (Ha a mikrogepre programozható LCD modul is csatlakozik, akkor azon a klub címe és telefonszáma jelenik meg, felváltva.)

- D6 villog, D7 kialszik, D8 LED világít : a program letöltés folyik,

- D6 villog, D7, D8 LED kialszik : program letöltés befejeződött, indítani lehet.

Ha a három LED egyszerre világít, akkor HW hiba (pl. RAM hiba) van. Itt jegyzem meg, hogy a processzor és a RAM típusa nem lehet teljesen független egymástól. Az után építések során kiderült, hogy egyes RAM vagy processzor típusok nem igazán akarnak együtt dolgozni, bizonyos memóriatartományban az írás/olvasás hibás volt. Szerencsére az előbbi hiba igen ritkán fordul elő

Ha a program letöltés megszakad, vagy azt el sem tudjuk indítani, akkor abban lehet hogy az egér program a "ludas". (Előfordult olyan mouse driver program amelyik nem elégedett meg a saját soros portjával, a másokra is rátelepedett.)

Röviden, a mikrogep kipróbálására is alkalmas a TESZT.AS programról: A mikrogepen futó program a D6, D7, D8 LED-eket

futófényszerűen kapcsolgatja. A kétirányú I/O portra (74LS245) másodpercenként adatot ír, valamint a portról egy olvasást is végrehajt. A 74LS573-al kialakított kimeneti portra 4 szegmenses multiplex vezérlésű LED kijelző csatlakozhat. A teszt program a program indulása óta eltelt percek, másodpercek jelzi ki, ha ide csatlakoztatjuk a LED kijelzőt. A program, a futása óta eltelt időt, valamint az I/O bővítő portról (74LS245) beolvasott adatot a soros vonalon továbbítja a PC-nek. Ahhoz hogy az előbbi adatok a képernyőn megjelenjenek a PC-n a TESZTPC.EXE programot kell futtatni. Ha a panelra rácsatlakoztattuk az intelligens LCD modult, akkor annak felső sorában a \*8031/32-es mikro\*, az alsó sorban a MIKROKLUB felirat, valamint a másodperc kijelzésnek kell megjeleníteni.

## Az MCS 52 BASIC rendszer

A BASIC rendszer indításának feltétele hogy a mikrón az MCS-52 AH BASIC rendszer fusson, tehát az eprom 0000-1FFFH 8 Kbájtos területére a BASIC interpeter legyen beégetve. (Ha 27512-es BASIC/Assembler epromot használunk, akkor a J1 jumper lehúzásával nulla voltot kapcsolunk az 1-es lábba, hogy az eprom alsó 32K területe - és az odaégetett BASIC - érvényesüljön.) Tehát a „bézikezésnél” a J1 legyen nyitva!

Két, PC-s kezelő program is készült a BASIC rendszerhez. Mind a kettő DOS-os, de valószínűleg az XP alól is használhatóak.

### A TERMIN.EXE program:

A PC-n indítsuk el a TERMIN.EXE programot. Kiterjesztésként meg kell adni hogy melyik soros portot (COM1 vagy COM2), és az átviteli sebességet. (600, 1200, 2400, 4800 baud.) Pl. a TERMIN COM1 1200 programindítással a PC az első soros porton 1200 baud-os átviteli sebességgel veszi fel a kapcsolatot a mikrogéppel. A BASIC egy automatikus baud sebesség felismerő rutin segítségével képes felismerni a processzor RXD bemenetére sorosan érkező adat átviteli sebességét. Ehhez mindössze annyit kell tennünk hogy a "SPACE" billentyűt leütjük. A mikrogép válaszol, és a képernyőn meg kell jelennie az \*MCS51 (tm) BASIC V.1.1\*, a következő sorban pedig a > jelnek.

```

MCS-51 BASIC terminál program
<c> MIKROKLUB Várpalota 1995.
    Tel.: 30/472-294
    8100 Várpalota
    Pf.: 65

F1 - Help
F2 - Program letöltés
F3 - Program kiolvasás
F4 - Editálás

F5 - LIST parancs
F6 - RUN parancs

F9 - EPROM file létrehozás

*MCS-51(tm) BASIC V1.1*
READY
>

```

A kapcsolat tehát létrejött, nekiláthatunk a program begépeléséhez, vagy a lemezzről beolvasott file kiküldéséhez. A terminál program automatikusan .BAS kiterjesztésű fájlokat keres, illetve a memóriából kiolvasott BASIC programhoz .BAS kiterjesztést ad.

Egy példa: a mellékelt P2.BAS program letöltése:

A terminál program alapértelmezésben a 25 soros, 40 oszlopos szöveges megjelenítési módot használja, de ha a /80 kiterjesztést is megadjuk akkor a 80 karakter/sor üzemmódra kapcsol. A mikrogépen futó interpreter program a sorosan vett adatokat visszaküldi a PC felé (echo), így a programbetöltés ellenőrizhető.

Ha a LIST parancs végrehajtásakor, vagy a program letöltéskor visszaküldött BASIC programsorok első karakterei nem látszanak, annak oka a PC, vagy a video kártya viszonylagos

```

*MCS-51(tm) BASIC V1.1*
READY
>

Program letöltés.
File név [.bas]: p2
10 INPUT A
20 XBY(0C800H)=A
30 GOTO 10

OK.
10 INPUT A
>20 XBY(0C800H)=A
>30 GOTO 10
>→
RUN
>RUN
?

```

lassúsága. Ez esetben válasszunk alacsonyabb baud sebességet, vagy segíthet a 40 karakter/sor formátum használata.

#### BASIC program mentése EPROM-ba:

A véglegesnek ítélt program (vagy programok) epromba égethető, onnan beolvasható. Sajnos a BASIC interpreter nem tartalmaz LOAD vagy SAVE utasítást, egyetlen programmentést támogat, a program epromba égetését egy, a processzorral vezérelt epromégető segítségével. Ez a megoldás azonban belső ROM-os vagy EPROM-os processzort, és egy jóval bonyolultabb (és drágább) hardware-t igényelne. A probléma megoldására a következőt találtuk ki: A mikrogépbe betöltött és lefordított BASIC programot egy gépi kódú rutin kiolvassa, és lemezre menti. Az így kimentett program egy epromégetővel tetszőleges helyre égethető (A BASIC az első programot a \$8010 címen keresi.) A lefordított, és beégetett BASIC program aztán a ROM paranccsal futtatható, illetve a RAM paranccsal a memóriába olvasható.

- TERMINAL program indítása (r.bat, SPACE) - BASIC program betöltése (pl. pl.bas)

- F9-el mentés fájlba (ekkor hívja a \$2100 címre égetett rutint a terminál, és ez küldi a programot a PC-nek.) Ha az átvitel rendben folyik, (a kontroll szumma képzés nem jelez hibát), akkor a terminál program minden 8 byte után kiír egy "\*" karaktert. - Létrehozza a megadott nevű fájlt (pl. pl.ep).

- Ezt a fájlt kell az EPROM-ba égetni a \$8010 címre

- Ezután a ROM paranccsal áttérni az EPROM programokhoz, majd a RUN paranccsal futtatni. A BASIC interpretert tartalmazó EPROM-ba be van égetve egy BASIC "reklám" program mintának, ami az előbbi módon futtatható.



### A TERMINT.EXE terminál program:

Készült egy újabb terminál program is, az INTEL AH BASIC rendszerhez szükséges kiegészítő funkciókkal. Ez a TERMINT.EXE. Ez a program is "dos-os", menü vezérelt. A használat:

- Kössük a PC soros portjára a mikrogépet.
- Kapcsoljuk be a mikrogépet úgy, hogy a BASIC rendszer fusson rajt. (Azaz a 27c256-ba a BASIC interpreter legyen, illetve ha 27c512-est használunk, a jumperrel e szerint válasszunk a két 32Kbyte-ra osztott tartalomból. Konkrétan a "mikroklubos" égetésű 27c512-nél a J1 jumper legyen nyitott.)
- Indítsuk el a TERMINT.EXE programot.
- Az első indításnál a program bekéri a használandó soros port azonosítóját, és a kívánt baud sebességet. (Ezt fájlba menti, a legközelebbi indításnál már ezt veszi alapul.)



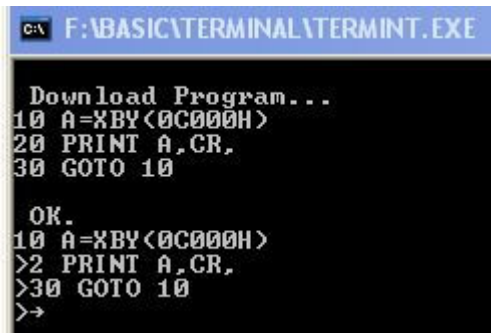
- Lépünk be a "User menübe", az alt-U billentyűkombinációval, vagy egy egérekattintással.

- Vegyük fel a kapcsolatot a mikrogéppel. Erre szolgál az "Init mikro" menüpont. (F9 billentyű) Mint arról korábban szó volt, a BASIC egy "space" karaktert vár a soros porton, amiből meg állapítja a baud sebességet. Az "Init mikro" tehát egy "space" karaktert küld a PC-ről a megadott soros porton, a megadott bauddal. A mikrón a D5 LED felvillanása jelzi a PC adását.



- A "space"-re válaszul a BASIC bejelentkezik, elküldi a \*MCS51 (tm) BASIC V.1.1\* karaktersorozatot, ami megjelenik a PC monitoron. Ha nem, akkor vizsgáljuk meg az összeköttetést, - villan a LED ? - azt hogy BASIC rendszer fut-e, rendben van-e a táp, stb.

- Ha minden rendben, töltsünk be egy BASIC programot! Az F2 leütése után válogathatunk a fájlokból, vagy beírhatjuk a nevét.

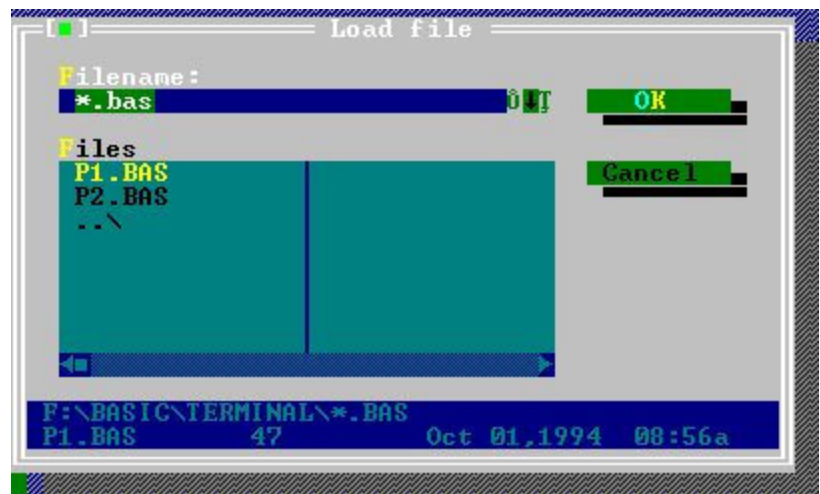


```

C:\ F:\BASIC\TERMINAL\TERMINT.EXE
Download Program...
10 A=XBY(0C000H)
20 PRINT A,CR,
30 GOTO 10

OK.
10 A=XBY(0C000H)
>2 PRINT A,CR,
>30 GOTO 10
>→

```



- Futtathatjuk a programot a RUN parancs (plusz ENTER) begépelésével, vagy az ezzel azonos hatású F6 billentyű leütéssel.

A program listázható az F5 billentyűvel - a BASIC LIST parancsot adja ki - kiolvasható az F3 billentyűvel. Az F4 billentyűvel a mikrogép RAM memóriájából olvasható ki a "tokenizált" BASIC program, amire akkor van szükség ha a BASIC programot az EPROM-ba akarjuk égetni, és azt szeretnénk ha automatikusan induljon a bekapcsolás után. (Ennek részleteiről már volt szó.)

### Az "ELEKTOR-os" monitor program:

Sokan megvették az ELEKTOR 1991/3 számában közölt 8032-es mikrogép monitorját, és példaprogramokat tartalmazó lemezét.

Ha az ott közölt példaprogramokat futtatni, akkor az EPROM-nak természetesen az ELEKTOR-os monitor programot kell tartalmazni, és a forrás programok lefordításához a lemezen található EASM51.EXE programot kell használni. A kvarc az



eredeti panelon 12 Mhz-es, de persze 6 Mhz-es kvarccal is futhatnak a programok, csak az RS232 adatátvitel sebesség megadásánál ezt figyelembe kell venni.

Ha 12-es a kvarc, a baud 2400, ha a fele akkora órajel frekvenciát használjuk, 6 Mhz-et, akkor 1200 legyen a baud!

Válasszuk pl. BSP11.A51 példaprogramot. Fordítsuk le az EASM51 BSP11 parancs kiadásával, majd indítsuk a V24COMG terminál programot. (Az eredeti, V24COM nevű Elektoros monitor programot kis mértékben módosítani kellett, mivel ez a mikrogép a soros átvitelhez szükséges +/-12 voltot a PC-től kapja, és ennek megfelelően kell állítani a DTR és RTS kimeneteket.) Azt hogy melyik soros portot, és mekkora adatátviteli sebességet használunk, a programnév kiterjesztésében kell megadni.

Pl. ha V24COMG BSP11.HEX -COM1 -2400 kiterjesztéssel indítjuk a terminál programot, akkor a kommunikáció az első soros porton történik, 2400 baud-os sebességgel.

Ha fut a V24COMG terminál program, és a mikrogép csatlakoztatva van, akkor a táp bekapcsolása után bejelentkezik az Elektoros rendszer:

```
--EMON51-- version 1.0 (15.8.1991) RAMTOP=44
```

Ha mégse, ellenőrizzük, hogy jó com portot jelöltünk ki, baudot, stb.

A mikrogépbe a lefordított program kódjait a Ctrl D billentyűk leütésével tölthetjük le, majd az X 4100 paranccsal futtathatjuk. (BSP11.A51 program a 4100H hexa címen kezdődik)

Ez a kis program a processzor Timer megszakítását mutatja be, szabályos időközökben egy "\*" karaktert küld a PC képernyőjére

```
A:\ELEKTOR>easm51 bsp11

EASM51 of file (BSP11) start pass 1 pass 2
LISTFILE written...HEXFILE written...
(57 lines, 0 errors, 76 bytes)
EASM51 exitcode=0

A:\ELEKTOR>v24comg bsp11.hex -com1 -2400

V24COM          ESC : exit program
CTRL F: enter filename ; CTRL D : start download
DOWNLOAD OF :BSP11.HEX
<<10 4100:90 41 34 74 02 75 30 40 12 02 00 75 0C 06 75 8A 047A>>
<<10 4110:06 75 89 22 75 51 20 75 50 64 75 52 00 D2 8C D2 0634>>
<<10 4120:A9 D2 AF 74 2A 75 30 01 12 02 00 E5 52 60 FC 75 068A>>
<<10 4130:52 00 80 EF C0 D0 C0 E0 D5 50 0C 75 50 64 D5 51 0871>>
<<0C 4140:06 75 51 28 75 52 01 D0 E0 D0 D0 32 053E>>
FILE TRANSFER COMPLETE
```

Végül nincs más hátra, mint hogy sok sikert kívánjak az építéshez, használatához. Viszontlátásra: Torkos Csaba 8100 Várpalota Pf 65 Telefon: napközben: 88/473-784, egész nap: 06/30/9472-294, email: mikroklub@vnet.hu internet: <http://www.mikroklub.hu>