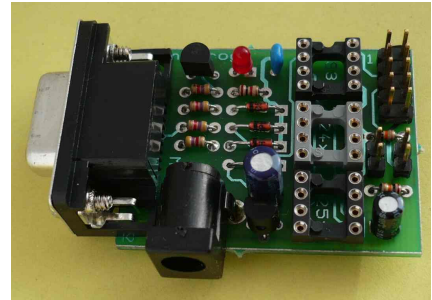


Programozó soros EEPROM-okhoz és ATMELEL 90Sxxxx, AT MEGA, AT TINY mikrokontrollerekhez

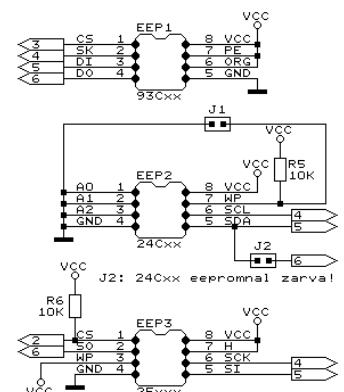
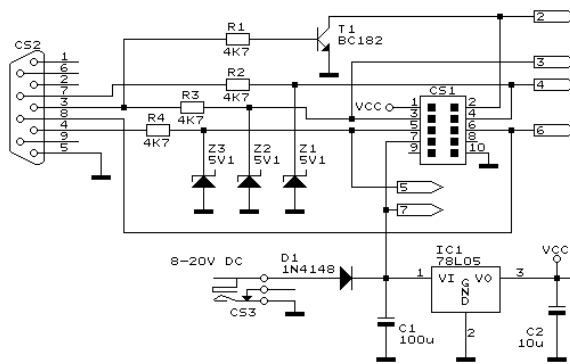
Az Interneten több égető kapcsolás, és program jelent meg, az egyik népszerű áramkör a Pony Programmer.



Az ismertetett áramkör és program lehetővé teszi a legtöbb soros EEPROM - 24xxx, 93xxx, 25xxx - és az ATMELEL AVR mikrokontrollerek többségének a soros, "in circuit" kiolvasását, beégetését.

A programozó áramkör:

Az eredeti kapcsolás koncepciója az volt, hogy van egy nagyon egyszerű alap panel, ami egy csatlakozón a tápot és a vezérlőjeleket biztosítja a programozandó IC-k számára.



És vannak kis adapter panelek, amiket a csatlakozóra kell dugni, hogy az egyes IC családok kezelhetőek legyenek. (24xxx, 25xxx, 93xxx eepromok, PIC és ATMELEL mikrokontrollerek.)

Ezt a felállást annyiban módosítottam, hogy az alappanelre felkerült a leggyakoribb EEPROM típusok, a 24xxx, 93xxx, 25xxx programozó foglalata is, így ezek kezeléséhez nem kell külön adapter.

A SPI jelek egy 5x2-es „tüskére” vannak kivezelve. (CS1) Innen egy darab 10 eres szalagkábel - felszerelve egy 10-es szalagkábeles anya csatlakozóval - kapcsolódhatunk a programozandó mikrokontrollert tartalmazó áramkörhöz, vagy a programozó adapterhez.

Ahogy látható, a felépítés nagyon egyszerű, 3 darab 5.1 voltos zéner illeszti a PC soros portjáról érkező kb. 12 voltos vezérlő jelet az 5 voltos rendszerben programozható eepromokhoz, mikrokontrollerekhez.

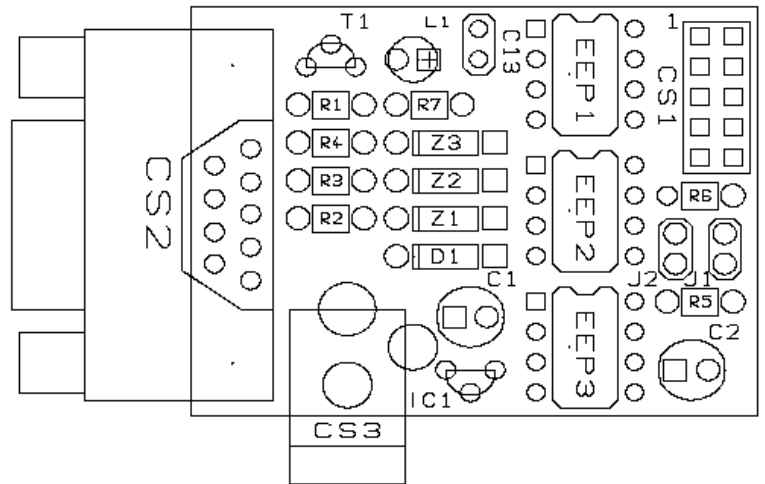
A tápfeszültséget egy 7805 stabilizálja.

A szükséges tápegység paraméterei: 9V-20V egyenfeszültség, 300mA. A fordított tápfesz ellen megvéd a D1-es.

(Egy átlagos - stabilizálatlan - DC 9-12V-os hálózati adapter e fenti igényt általában kielégíti.)

A tápfeszültség jelenlétét az L1 LED jelzi.

A J1 és J2 jumpernek, csak a 24xxx eepromoknál van szerepe. Ha 24xxx EEPROMot akarunk írni/olvasni, akkor - de csak akkor - a J2 legyen zárva. De minden egyéb esetben a J2 legyen nyitva!



A PONYPROG2000.EXE program használata:

A program funkciói: olvasás, égetés, törlés, a beolvasott eeprom vagy mikrokontroller tartalmának lemezre írása, vagy onnan az égetendő tartalom beolvasása. **A program futtatható WIN98 és XP alatt is.**

És akkor a használata, egy konkrét példán keresztül:

Legyen a feladat, egy „PLUS201.BIN” nevű file beégetése egy 24C32-es EEPROM-ba!

- Csatlakoztassuk az égetőt a PC soros portjához

(Ha nem akarunk a számítógép hátsó csatlakozóin matatni, használjunk egy soros port toldó kábelt. Sokkal könnyebb a kezelés, és máskor is jól jön, ha kényelmesen elérhető a soros port.)

- az égetendő 24C32 IC-t rakjuk az EEP2 foglalatba

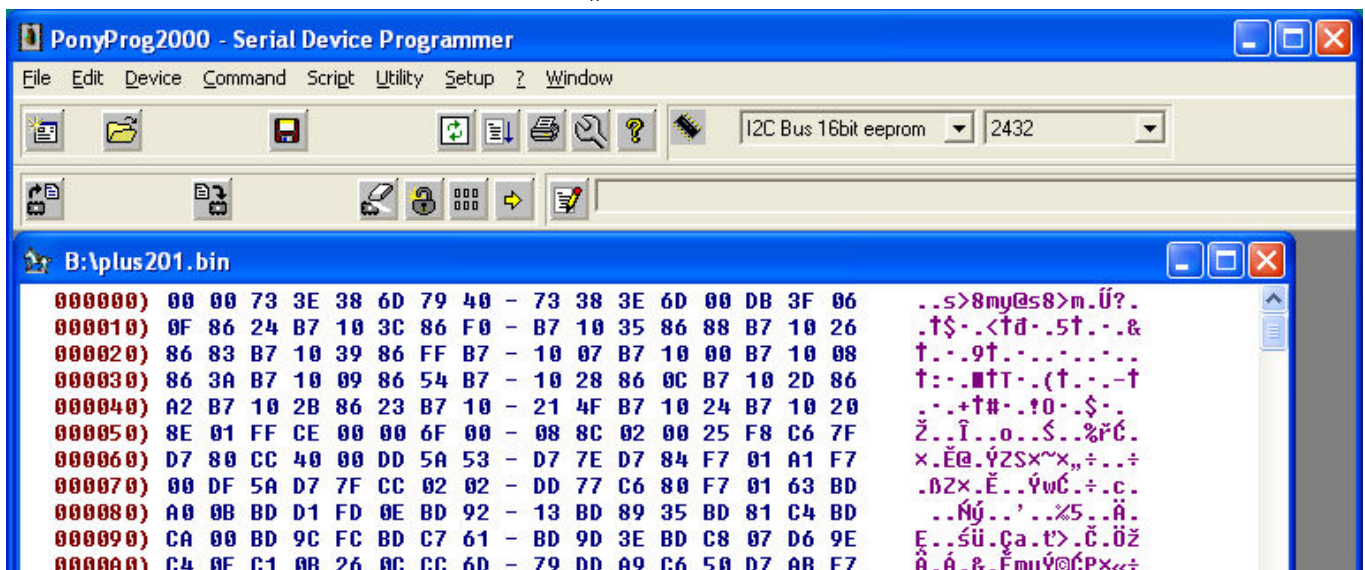
- mint arról szó volt, a 24xxx eepromok kezelésénél a J2 legyen zárva, tehát tegyük fel hát ezt a jumpert,

- **Most, tehát az előbbiek után, kapcsoljuk be - dugjuk be - a tápot. A sorrend fontos! Ahogy a legtöbb számítógépre csatlakozó periféria leírása is tartalmazza, ne kössünk össze táp alatt lévő készülékeket!** Az L1 LED világítása jelzi a tápfesz meglétét.

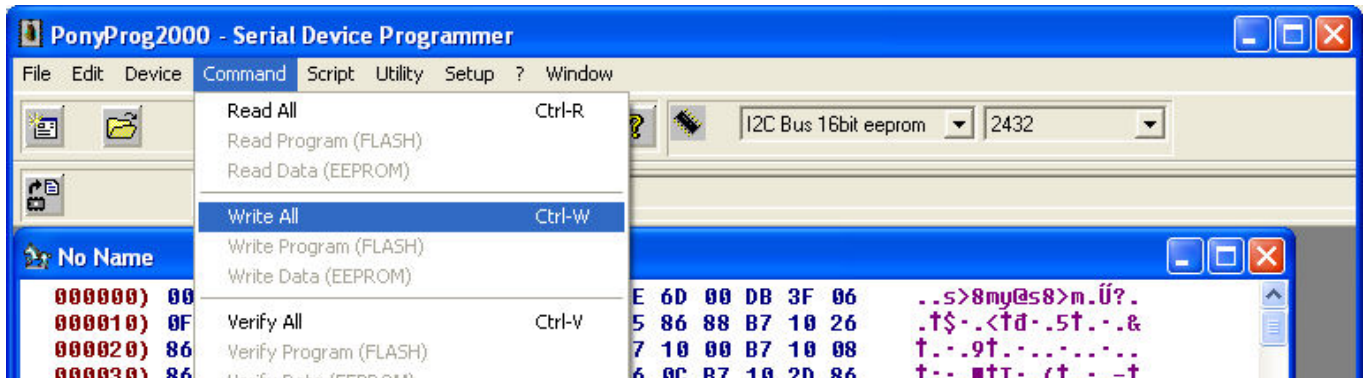
- Indítsuk el a PC programot!

- Válasszuk ki a 2432-es típust. Az "I2C bus 16 bit eeprom" csoportban találjuk meg. (24C32 nincs a menüben, de ne aggódjunk, a 2432 jó választás lesz, a programozása ugyanaz.)

- olvassuk be a "bufferbe" a „PLUS201.BIN” nevű file-t!



- most adjuk ki a "Write all" parancsot:



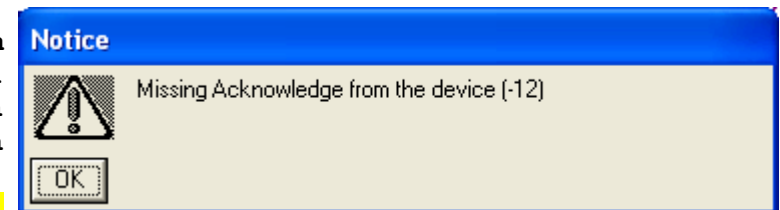
- Az égető program figyelmeztet, hogy minden, az IC-ben esetleg található adat törlődni fog:

Semmi gond, lépünk tovább a Yes-el!



- És most kapunk egy hibaüzenetet... Na persze, nem lett felrakva J1 jumper, ezzel tudjuk alacsony szintre húzni a WP lábat, és így engedélyezni a programozást.

Szóval **tegyük fel a J1 jumpert**, és újra adjuk ki a "Write all" parancsot!



- a program most programoz, majd leellenőrzi az égetést. Ha minden rendben, a Write succesful OK - az írás sikeres - üzenettel örvendeztet meg a program:



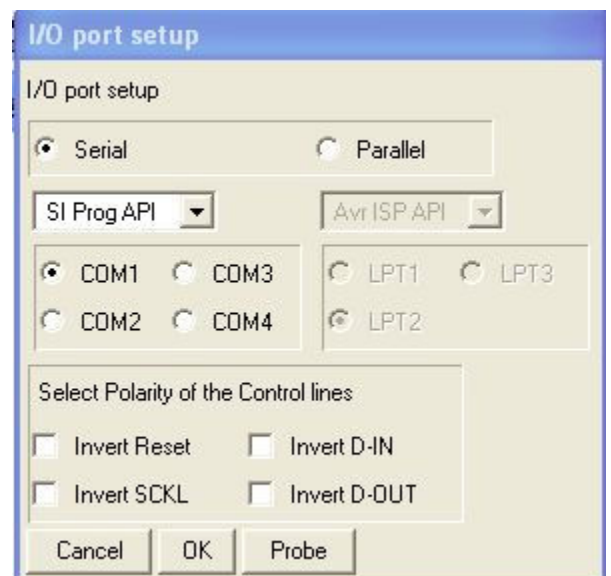
Ha sikertelen volt a próbálkozás:

- nézzük át a csatlakoztatást, a tápot. Világít a LED?

- Jó COM port van beállítva a setupban?

- Ha ez rendben, akkor nem él egy korábban telepített eszköz meghajtó programja a porton?

- Ha 24Cxxx- IC-t programozunk, fenn a J2 jumper? Ha egyéb IC-t, akkor ugye nincs fenn?

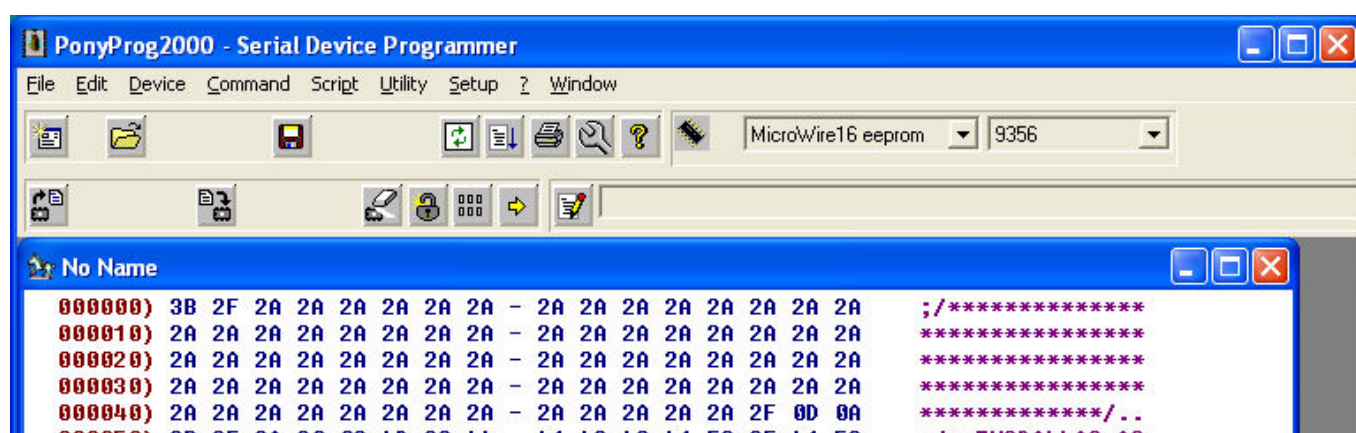


A 93xx soros eepromok égetése:

Na először egy kicsit a 93xx eepromok belső világáról...

A 93xx EEPROM-ok esetében a tárolt adatbitek 8 vagy 16-os egységekben kezelhetők. Egyes típusok csak 16 bites, pl. NATIONAL 9306, 9346, MICROCHIP 93C46B, 93C56B, vagy csak 8 bites, mint a MICROCHIP 93C46A, 93C56A, stb., és vannak mindkét adatformátumban használhatóak. Utóbbiaknál a választás az IC "org" kivezetésre adott jelszinttel lehetséges, a magas szint a 16 bites, az alacsony a 8 bites adatszervezést jelöli ki.

Az égetőnél ez az "ORG" láb magas szinten van, tehát általában 16 bites formátumban kezeli az eepromokat, a típus választásnál ezt vegyük figyelembe, azaz általában a "MicroWire16 eeprom" menüből válasszunk. Ha pl. egy 93c56-ot akarunk kezelni, akkor ez lesz a beállítás:



A csak 8 bites egységekben írható, olvasható típusoknál, mint pl. a MICROCHIP 93C46A, 93C56A az ORG kivezetés állapota nem számít, és ilyenkor a "MicroWire16 eeprom" menüt válasszuk.

ATMEL mikrokontrollerek égetése a PONYprogrammerrel:

Röviden az ATMEL AVR mikrokontrollerekről:

Az ATMEL AVR mikrokontrollerek:

Az ATMEL 90S1200, 90S2313, 90S4414, 90S8515, valamint az ezeket felváltó AT MEGA 8515, AT TINY 2313, és az újabb, AT MEGA 161-168 AVR mikrokontrollerek elektromosan törölhető, és újraprogramozható FLASH program és adat memóriával rendelkeznek.

A soros, és a párhuzamos programozás:

Az ATMEL a legtöbb AVR mikrokontrollerhez két programozási lehetőséget is biztosít, a soros, és a párhuzamos programozást.

A párhuzamos programozás esetén a beprogramozandó adatokat bájtanként írjuk be, azaz egy ütemben 8 bitet. Hátránya, hogy a programozáshoz sok kivezetésre van szükség, ami viszonylag bonyolult programozó áramkört igényel. Előnye hogy valamivel gyorsabb, mint a soros programozás. De talán ennél sokkal fontosabb, hogy bizonyos paraméterek csak a párhuzamos programozással állíthatóak be. (Amiből aztán problémák adódnak, de erről majd később...)

A soros programozási algoritmus szerint a beírandó adat bájtokat bitekre bontva, sorban egymás után, egy órajellel szinkronizálva kell beléptetni. Csak néhány kivezetés kell ez esetben, ezek egy SCK órajel, egy MOSI és MISO adatvonal, és egy RESET jel. Ezt a programozási módot használja a PONY programozó is.

Az előbbiekből adódik a soros programozás egyik előnye, hogy lehetőséget a mikrokontroller áramkörben - in circuit - programozására, és a programozó áramkör hardvere is nagyon egyszerű lehet. Ha egy mikrokontrollert az áramkörben akarjuk programozni, akkor erre már jó, ha a tervezéskor is figyelünk! Érdemes a szükséges portokat egy csatlakozóra kivezetni, és vegyük figyelembe, hogy ezeken a kivezetéseken jönnek (mennek) majd a vezérlőjelek.

A 40 lábú ATMEL AVR IC-k kezelése:

Először is a programozó CS1 csatlakozója, és a programozandó DIL 40 tokos IC összekötése:

Programmer:	IC
1. (VCC) -	40.
2. (RESET) -	9.
4. (SCK) -	8.
5. (MISO) -	6.
6. (MOSI) -	7.
10. (GND) -	20.

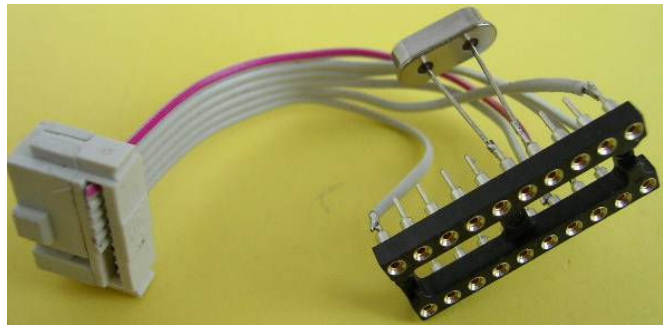
A 18 - 19 lábra egy kvarcot kell kötni!

A 20 lábú AVR IC-k kezelése:

A programozó, és a programozandó DIL 20-as tokozású IC összekötése:

Programmer: IC

- | | | |
|------------|---|-----|
| 1. (VCC) | - | 20. |
| 2. (RESET) | - | 1. |
| | | |
| 4. (SCK) | - | 19. |
| 5. (MISO) | - | 18. |
| 6. (MOSI) | - | 17. |
| | | |
| 10. (GND) | - | 10. |



A 4. - 5. lábra egy kvarcot kell kötni!

A konfigurációs, és „code protect” bitek:

A beégetett program működését alapvetően befolyásolják, az úgynevezett "fuse bitek". A korábbi kibocsátású IC-knél még nem, vagy csak 1-2 ilyen bit volt, az újabbaknál egész sor. Pl. a 90S8515-nél még egy se, az "utódjánál", az AT MEGA 8515-nél már 16 is van. A probléma az, hogy ezek a bitek a forrásprogramban nem adhatóak meg. (Ez egy óriási hiba! Az IC felprogramozása így sokkal körülményesebb, és nagy a hibalehetőség is.) Tehát ha ezek a konfigurációs bitek nem helyesen vannak beállítva, akkor a mikrokontrollerbe égetett program nem, vagy nem helyesen működik!

Mint minden újabb mikrokontrollernél, itt is lehetőség van a beprogramozott adatok kiolvasásának letiltására. Ha "Code protect" bitet beprogramozunk, akkor beégetett tartalom nem olvasható vissza, helyesebben, ha kiadjuk az olvasás parancsot, akkor vagy csupa FF-et látunk - pl. a TINY 2313 esetében - vagy látszólag "kijönnek" adatok - pl. az AT MEGA 8515 esetében - de azok "fals" értékek.

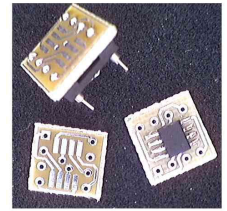
(Gyakran kapom a kérdést, hogy akkor hogy lehetne mégis kiszedni az adatokat, egy titkosított chipből. Sehogy. Pont erre találták ki...) A törlés - erase - funkcióval törlődnek a titkosító bitek - de egyben a chip tartalom is.

Használat USB portról:

Elvileg egy USB/Serial port átalakítóval meg lehetne próbálni. A legtöbb gyártó azonban csak az RXD és TXD vonalakat viszi át, márpedig ennél a speciális alkalmazásnál az RTS és DTR jelet is felhasználja a vezérlés, szóval az USB/RS232-es adapterről működtetés valószínűleg nem fog menni...

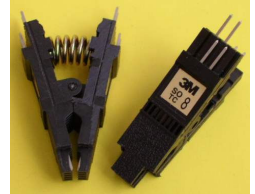
Adapter, csipesz az SMD tokos EEPROM-okhoz :

A 8 lábú DIP IC tokra forrasztott panel végszükségletben, mint "fapados" SMD adapter is használható programozó készülékekhez. Értelemszerűen a DIP tokot csíptessük a karos foglalatba, az SMD tokot pedig az olvasás vagy programozás alatt szorítsuk a panel SMD fóliázatára.



Szintén az SMD IC-k kezelését könnyítik meg az SMD csipeszek.

A csipeszt egy szalagkábeles DIP8-as IC csatlakozóval tudjuk az égetőhöz kötni. Arra figyeljünk, hogy az IC csatlakozóknál a kijövő vezetékek "tükörben" vannak, tehát az első vezeték nem az 1-es, hanem a 8-as. (Tehát 8, 1, 7, 2, 6, 3, 5, 4 lesz a csatlakozóból kijövő vezetékek sorrendje.)



A "csiptetésnél" persze figyeljünk, hogy az smd IC 1-es lába legyen összeköttetésben a programozó foglalat 1-es lábával.



Gyakori kérdés, hogy programozható-e kiforrasztás nélkül, az áramkörben a chip. Esete válogatja, illetve próba szerencse... Szóval ez egyrészt függ az áramköri környezettől, azaz mi csatlakozik az eeprom lábaira - általában egy mikrokontroller - és a programozó tudja-e annak ellenében mozgatni a kivezetések szintjét. Valamint még a tápellátás "kritikus". Azaz hogy a programozó által adott 5 volt elég-e a panel számára. (Hiszen nem csak az eepromot fogja a tápot terhelni, hanem minden 5 voltra csatlakozó IC.)

Szóval lehet hogy 1-2-3 lábat "fel kell emelni" a panelről.

Epilógus:

- Még egyszer leírom - bár az előbbiekből is kiderül - hogy a működtető program(ok) fejlesztése (munkája, érdeme, stb.) tőlem teljesen független, én csak a nyomtatott áramköri tervet, a hardvert, és ezt a dokumentációt készítettem hozzá.

Végül nincs más hátra mint hogy sok sikert kívánjak a használatához. Viszontlátásra: Torkos Csaba 8100 Várpalota Táncsics u. 7. Telefon: napközben: 88/473-784, egész nap: 06/30/9472-294, email: mikroklub@vnet.hu internet: <http://www.eprom.hu> , <http://www.mikroklub.hu> .