

A VP-299, VP-598, VP-798, VP-998 univerzális IC programozók

A Wellon cég sokat tudó IC programozó készülékei a Kínai Népi Demokratikus Köztársaságból. Nézzük, mit tudnak:

VP-598:

Több mint 23.000 (!) támogatott IC típus. EPROM, EEPROM, , Atmel, Intel, Microchip, Signetics, Zilog mikrokontrollerek kezelése, de GAL és PALCE IC-eket is programozhatunk vele.

USB-s csatlakozás, nagyon gyors adatforgalommal. A tápfeszültséget is az USB adja, de lehetőség van külső táp csatlakoztatására is.



VP-798:

A VP-598-nál többet tudó, több mint 37.000 IC típust támogat.



VP-998:

a wellon cég csúcs programozója, csaknem 105.000 támogatott IC típus.



VP-299:

Végül a legkisebb, a VP-290. Ennek csak 40 lábú a programozó foglalat, de ha a nem a ritka, száz lábú epromokat kell égetnünk, ez is elég lehet.



VP-ISP soros programozók:

A programozó család tagjai még a VP-ISP soros programozók, amelyek a sorosan, tehát néhány csatlakozóponton keresztül írható, olvasható IC-khez használhatóak. Általában a soros eepromok, és az újabb mikrokontrollerek tudják a soros program letöltést.

Az égető hardvere itt a lehető legegyszerűbb, nincs is programozó foglalat, csupán az IC programozáshoz szükséges csatlakozót, és kábeleket biztosítja.

Az égetőt ugyanaz a PC program kezeli, mint ami a „nagy testvéreket”, de természetesen szűkebb a kezelt IC-k listája - „csak” pár ezer – és nem tud olyan speciális funkciókat, mint az IC tesztelés, stb.

A VP-ISP10 programozóról az eltérő felépítés, és használat miatt egy külön leírás szól.



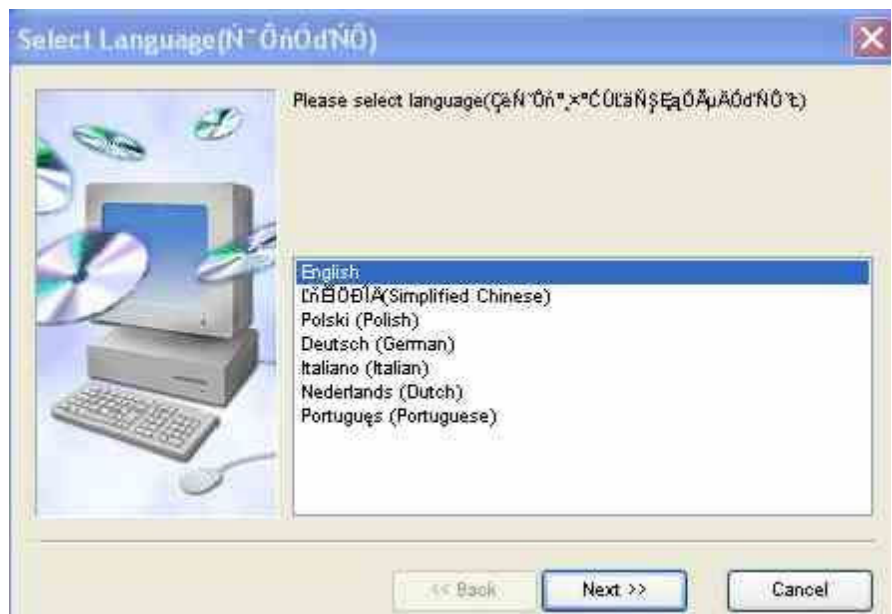
A VP-299/598/798/998 típusra egyaránt érvényes tulajdonságok:

- 1.5 - 5 voltos IC-k támogatása
- IC csatlakozás vizsgálat. Az égető teszteli a behelyezett IC kivezetéseit, ha úgy találja, hogy érintkezési hiba van, ezt jelzi. Pl. a „sok lábú”, smd tokos TSOP, PSOP adapterrel égethető IC-knél ez nagyon hasznos.
- A 48 lábú, karos IC foglalatnak köszönhetően, az összes támogatott, DIP tokos IC adapter nélkül égethető. (azaz nem kell külön adapter pl. a mikrokontrollerekhez, a 16 bites epromokhoz, stb.)
- Windows 98/2000/XP/Vista/Win7 alatt működtethető. Ugyanaz a PC szoftver használható mindkét égetőhöz, a program felismeri a csatlakoztatott égetőt, és a működését ahhoz igazítja.
- Formatervezett, kicsi, strapabíró műanyag doboz.
- Képes a gyártmány kód kiolvasására.
- PLD-k – GAL, PALCE, stb. - programozása. Ismeri az összes JEDEC formátumot (ABEL, CUPL, PALASM, TANGO PLD, OrCAD PLD)
- A 74xxx, 40xxx, 45xxx TTL és CMOS IC-k, valamint RAM-ok tesztelésére is alkalmas.
- Az égetőt egy mikrokontroller vezérli. Ennek talán a legnagyobb előnye az, hogy stabilak, a windows-tól függetlenek a programozási folyamatot meghatározó időzítések.
- Sorozat programozás funkció.
- Ingyenes szoftver frissítés.
- Egy két, IC programozóktól szokatlan plusz szolgáltatás: négyszög jel generálás, frekvencia mérés, „logikai analízátor”. (Kb. 100.000 Hz-ig.)

A szoftver, és az USB driver telepítése:

- Először is: még a programozó csatlakoztatása előtt installáljuk a működtető programot! (A mellékelt CD-n a SETUP.EXE)

Mondanom se kell, hogy az angol nyelvet válasszuk.



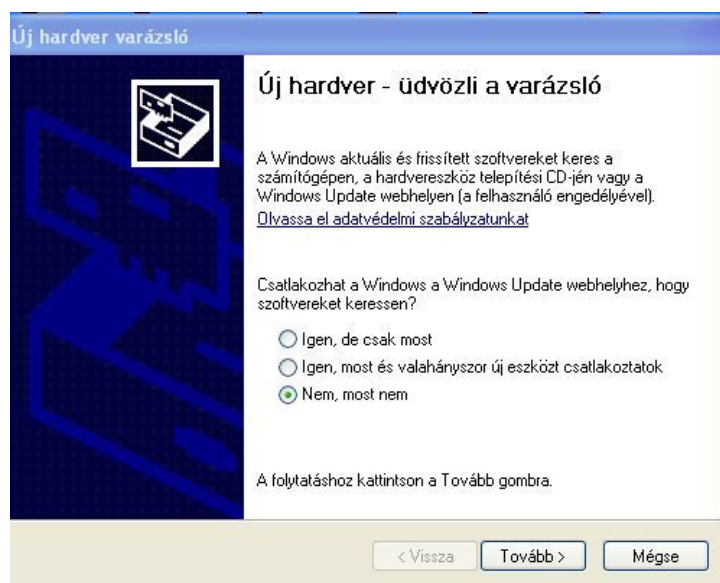
- A PC program installálása után, de még a program futtatása előtt csatlakoztassuk az égetőt, egy szabad USB porthoz. Ha USB elosztón - HUB-on - keresztül csatlakoztatjuk a PC-hez az égetőt, akkor szükség lehet a külső tápegységre. (9V DC. Az égető tartozéka.)

Szóval a legjobb, a tápellátás szempontjából is, ha egy közvetlen - nem HUB-os - USB aljzatba csatlakoztatjuk az égetőt.

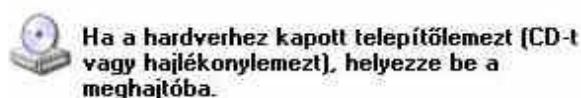
A csatlakoztatás után telepedik fel az égető meghajtó programja, a windows új hardver-t fog találni:



- „Nem, most nem” kell hogy a windows egy webhelyhez csatlakozzon:



- Aztán telepítés „beavatkozás nélkül” :



Mit tegyen a varázsló?

- ☒ Telepítse a szoftvert beavatkozás nélkül (ajánlott)
- ☐ Telepítsen listából vagy adott helyről (haladóknak)

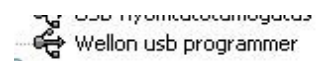
A folytatáshoz kattintson a Tovább gombra.

Azzal ne törődjünk, ha az XP figyelmeztet, hogy ez a szoftver nem esett át microsoft tesztjén, folytassuk a telepítést:





Ha minden rendben, a végén az eszközvezérlőben, az „USB-vezérlők” között, meg találhatjuk a programozónkat:



Fontos dolgok a használathoz:

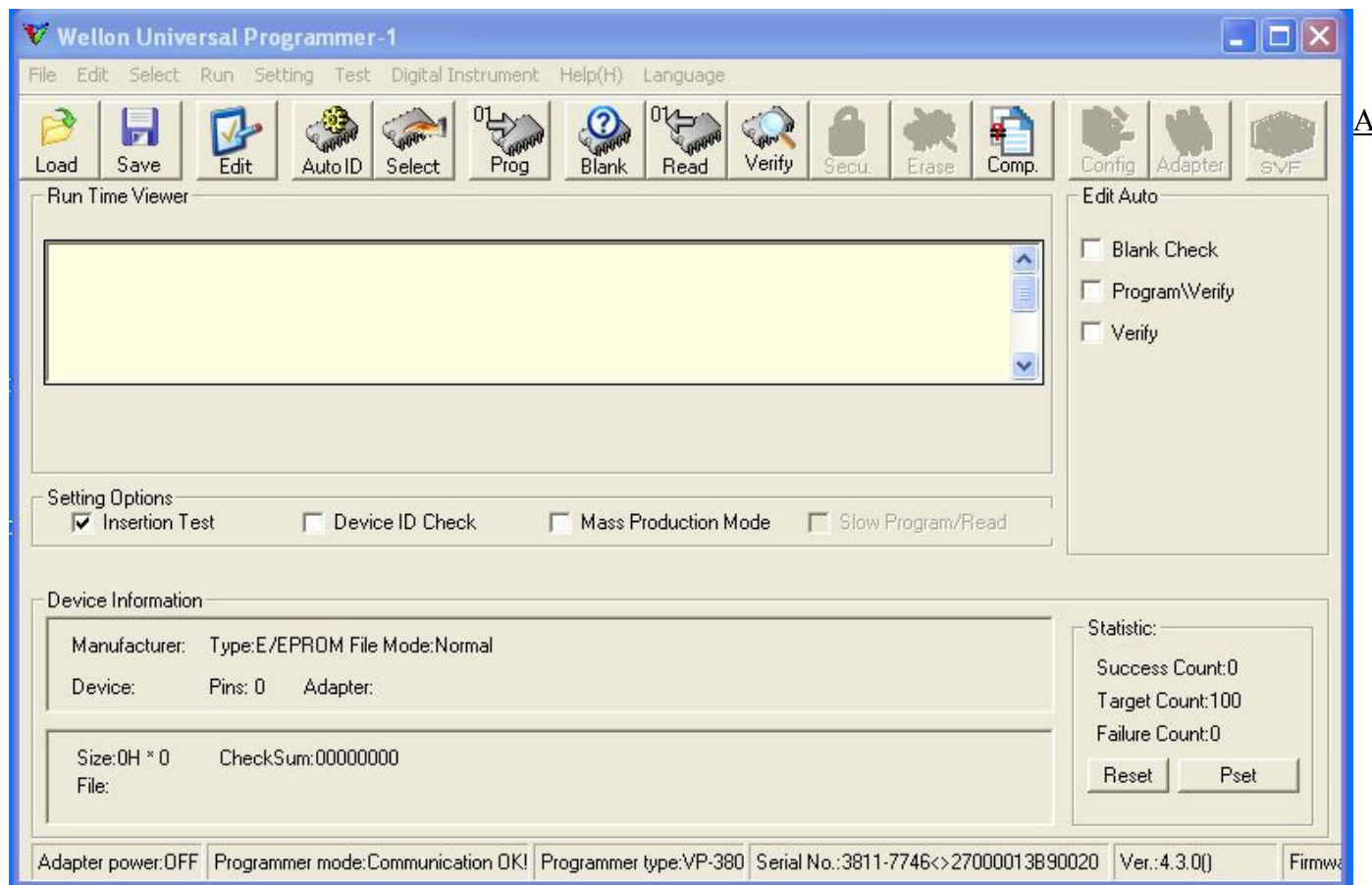
Ne csatlakoztassuk, vagy húzzuk le az égetőt a PC-ről, ha külső táppal használjuk. (Ha külső tápot akarunk használni, előbb csatlakoztassuk az USB-re, dugjuk be a tápcsatlakozót, és csak ezután dugjuk a dugasztápot a konnektorba.)

A bekapcsolás alatt ne legyen IC a foglalatban, illetve ne hagyjunk IC-t a programozó foglalatban, ha kilépünk a programból! Hogy miért is olyan fontos ez? Az égető figyeli a programozott IC tápáram felvételét, ha ez túl magas, beavatkozik. De ha a PC bekapcsolásakor már benn van egy IC a foglalatban, az égető pedig az USB-n keresztül tápot kap, akkor a bekapcsolási folyamatban ellenőrizetlenül juthat ki valamilyen feszültség a foglalatra – az égető honnan is tudná, milyen IC van a foglalatban, ráadásul a vezérlő mikrokontroller, elektronika, feléledéséhez is kell egy kis idő – ami szerencsétlen esetben a foglalatban hagyott IC, de akár az égető meghibásodását is okozhatja!

A PC program használata:

Az égetőt és a PC-t kössük össze az USB kábellel!

A PC program automatikusan felismeri, és ellenőrzi a csatlakoztatott égető hardvert. Ugyanaz a PC program kezeli a VP-380, és a VP-990 égetőt is, a PC program azonban az induláskor azonosítja a típust, és a működését – pl. az IC típus választásnál – ahhoz igazítja.



képernyő felosztás:

A felső ikon sor elemeire kattintva indíthatóak a leggyakrabban használt műveletek.



Középen az „üzenőfal”, a programozási folyamatok, üzenetek láthatóak itt.

Jobb oldalt a programozáskor használatos műveletekből állíthatunk össze egy automatikusan végrehajtódó művelet sorozatot. (Pl. IC törlés, törlés ellenőrzés, programozás, stb.)

A legalsó sorban az égető állapotjelzése, azaz hogy kap-e külső tápot, aztán a kommunikáció OK, majd az égető típusa – VP-380 – gyári száma, program verziója.

A program "ikonok" és funkciói:



"Load fájl to buffer" : file beolvasás a bufferbe, azaz annak a munkaterületnek, memóriatartománynak a feltöltése adatokkal, ami majd az IC-be kerül.



"Save buffer to a file" : a buffer mentése egy fájlba. Azaz pl. egy IC-ből kiolvasott, és a memóriába - amit itt buffernek nevezünk – elhelyezett adatokat menthetjük el.



"Auto ID" : elektronikus azonosító kiolvasása a chipből. (Nem minden mikrokontroller, vagy memória rendelkezik kiolvasható azonosítóval.)



"Select chip" : IC típus választás.



"Program" : a bufferben lévő adatok beégetése - beprogramozása, írása - a chipbe.



"Blank" : üresség teszt, azaz töröltség ellenőrzés. (Első sorban az UV fénnel törölhető IC-knél van jelentősége.)



"Read" : az IC tartalmának kiolvasása. (És a „bufferbe” helyezése.)



"Verify" : az IC-ből kiolvasott adatok összehasonlítása a buffer adatokkal.



"Security" : titkosítás, lezárás. Vannak IC típusok, amelyeknél tiltható kiolvasás, vagy a programozás. Az ezt beállító „titkosító” biteket programozhatjuk be.



"Erase chip" : az elektromosan törölhető IC-knél adható ki ez az utasítás, és törli a chipet.



Compare : Az IC-ből kiolvasott adatok, és a „buffer” összehasonlítása.



"Config" : csak azoknál az IC típusoknál van értelme - aktív - ahol működést beállító biteket is lehet programozni. (Pl. Microchip, vagy ATMEL mikrokontrollerek.)



Adapter : ha adapter kell az adott IC-hez, annak kiosztása tekinthető meg itt.

A programozáskor használt parancsok kiadhatóak egy műveletsorként is. A jobb oldalon van az automatikus programozás parancs mezője - Edit auto - ahol kiválaszthatjuk a programozás során végrehajtandó művelet sort.

Az „Erase” a törlést, a „Blank Check” a törlés ellenőrzést, a „Program” a programozást, vagy mondhatjuk úgy is, az írást, a „Verify” pedig a visszaellenőrzést jelenti.

Ha pl. egy 29F010 flash EPROM-ot akarunk törölni, és utána ellenőrizni is a törlést, majd égetni, végül ellenőrizni az égetést, akkor a jobb oldali ablakban jelöljük ki ezeket a műveleteket:

Edit Auto

- ☒ Erase\Blank Check
- ☐ Blank Check
- ☒ Program\Verify
- ☐ Verify

Ha sok IC-t kell azonos tartalommal beprogramoznunk, akkor lehet hasznos a „Mass Production Mode” - tömeggyártás mód – opció. Ekkor – de csak ez esetben – jelöljük be „Mass Production Mode”-ot:

Setting Options

- ☒ Insertion Test
- ☒ Device ID Check
- ☒ Mass Production Mode
- ☐ Slow Program/Read

Rakjuk be, az IC-t a programozó foglalatba, és már indul is a programozás. (A programozó észleli hogy beraktuk az IC-t.)

A végén kéri a következőt – ezt a kék LED villogásával is jelzi - egyben jelzi hogy hányadiknál tartunk:

Status:

Success Count: 2
Target Count: 100
Failure Count: 0

Reset Pset

Azt is be lehet állítani a „Pset”-re kattintva, hogy mennyit szeretnénk, és ha elérjük a kívánt darabszámot, akkor az elértük a célszámot üzenetet kapjuk:

>>Target Count is arrived!!

Pset

- ☒ Auto Program
- ☐ Verify
- ☐ Blank Check
- ☐ Erase

- ☒ Target Count enable
- Success Count:
- Target Count:

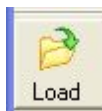
„Slow Program/Read” : tapasztalatból mondom, pipáljuk ki. Alig lesz lassabb a folyamat, de volt már hogy órákat töltöttem egy IC írásának próbájával, és csak nem ment. Aztán észrevettem ezt kockát, és kipipálva hirtelen minden jó lett. Na ez a sok idő...

☒ Slow Program/Read

Amióta ezt a leírást csináltam, már magyarul is tud a program. Ha ezt állítjuk be, akkor ez pl. így néz ki:

☒ Lassú programozás/olvas

Egy fájl betöltése a „bufferbe”:



A File/Open paranccsal, vagy az ikonra kattintva tudunk egy égetendő tartalmat beolvasni a bufferbe.

Egy „szokásos” böngésző menüben válogathatunk a fájlok között.

A fájl kiválasztása után egy újabb ablak jelenik meg:

Egyrészt megadhatjuk, hogy a beolvasandó fájl egy „sima” adat csomag – ez a bináris, azaz .BIN fájl - vagy valamilyen speciális fájl formátum, mint pl. a .HEX vagy .JED fájl.

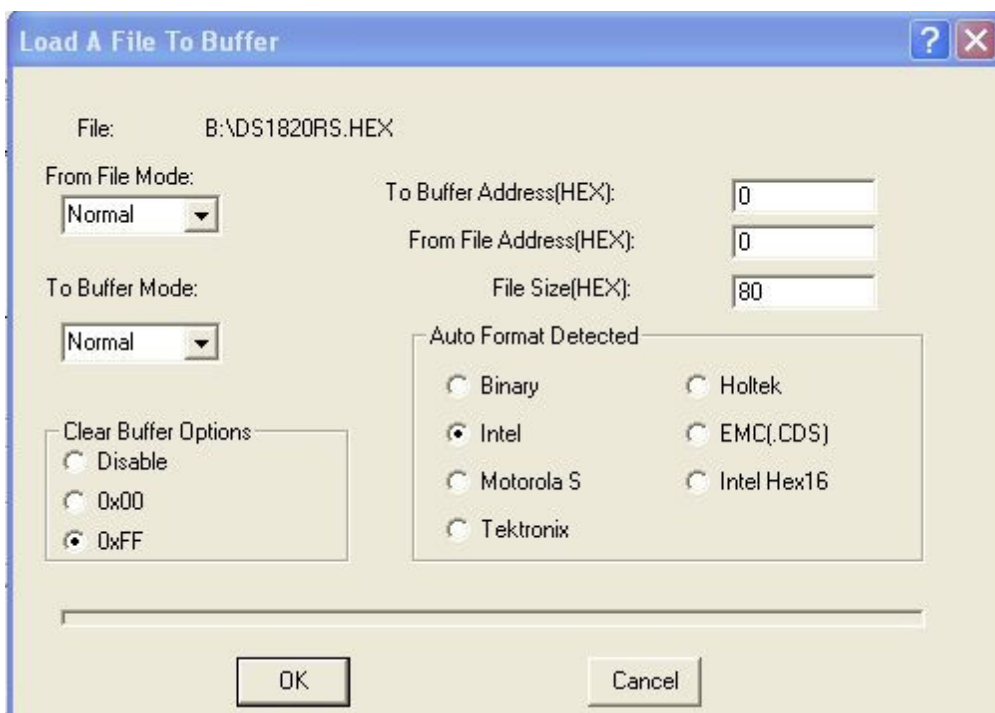
De a HEX, JED, stb. formátumú adatfájlokat a program automatikusan feldolgozza - „kicsomagolja” tulajdonképpen egy bináris fájlra alakítja - és „buffer ablakban” már az égetendő adatok látszanak.

Aztán választható az is, a „Clear Buffer Options” ablakban, hogy az üresen maradt helyet milyen adattal töltsse fel, azaz 00-val, vagy FF-el, vagy ne csináljon semmit a maradék helyel, „Disable”. Ezt az opciót általában hagyjuk az alapértelmezett FF-en.

Pl. ha egy 32Kbájtos memóriába 4K-s adatot töltünk, akkor ugye marad 28K üresen, ami így FF-el .

És van még egy lehetőség, hogy az adatokat ne a memória elejétől töltsse be, hanem egy adott címtől. Ez a cím adható meg To Buffer Address(HEX) ablakban. Normál esetben persze maradjon a 0 kezdőcím.

De általában is elmondható, hogy egy „szokásos” IC programozásnál ne változtassuk meg az alapbeállításokat.



A „buffer” editor:



Ha az "Edit"-re kattintunk, akkor az átmeneti tár – a buffer – tartalmát látjuk, azaz az IC-ből vagy egy fájlból beolvasott adatokat.

Buffer Editor																
Fill Random Swap Search Next Copy CheckSum Print Show Exit																
ADDRESS	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
00000000H	23	28	FF	FF	FF	FF	FF	FF	FE	00	03	0E	83	01	FF	00
00000010H	0C	1D	1E	28	0C	11	64	00	A5	0B	1E	28	0A	30	A5	00
00000020H	01	30	A0	06	A6	0B	1E	28	04	30	A6	00	98	1C	1B	28
00000030H	18	12	00	00	18	16	01	30	86	06	A0	14	7F	0E	83	00
00000040H	FE	0E	7E	0E	09	00	85	01	86	01	07	30	9F	00	83	16
00000050H	8F	30	81	00	30	30	85	00	02	30	86	00	0C	15	18	15
00000060H	19	30	99	00	98	16	83	12	98	17	18	16	20	30	84	00
00000070H	80	01	84	0A	84	1F	38	28	0A	30	A5	00	04	30	A6	00

Az adatokat nem csak nézegethetjük, de át is tudjuk írni, azaz „editálhatjuk” a tartalmat. Az átírt bájtok színe kék lesz.

A „Fill” paranccsal egy adat kóddal tölthető fel a buffer, a „Random” paranccsal pedig véletlen szerű értékekkel. (Gondolkodtam ez utóbbi parancs értelmén, de pl. próba égetéseknél, programozó adapterek tesztjénél jól jön.)

A „Swap”, azaz felcserélés menüre kattintva a szomszédos adat bájtok helyet cserélnek. Azaz az első a másodikkal, a harmadik a negyedikkel, és így tovább. Na, ennek mi értelme? Lehet a jogos kérdés. Pl. a 16 bites epromoknál ugye két egymást követő 8 bites adat adja meg a 16 bit tartalmát. De arra nincs semmi előírás, hogy a bitek sorrendje 15.-8. és 7.-0., vagy előbb vannak az alacsonyabb helyi értékűek, azaz 7.-0. és 15.-8. Ez attól függ, milyen programmal tárolták le azokat, stb. És néha jó lenne megcserélni a sorrendet.

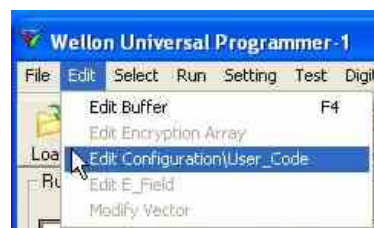
Kereshetünk az adatok közt hexa, vagy ASCII kód szerint – szöveget - a „Search” paranccsal. A „Next”-el folytathatjuk a keresést.

A „Copy”-val egy kijelölt terület tartalmát másolhatjuk egy másik kijelölt területre.

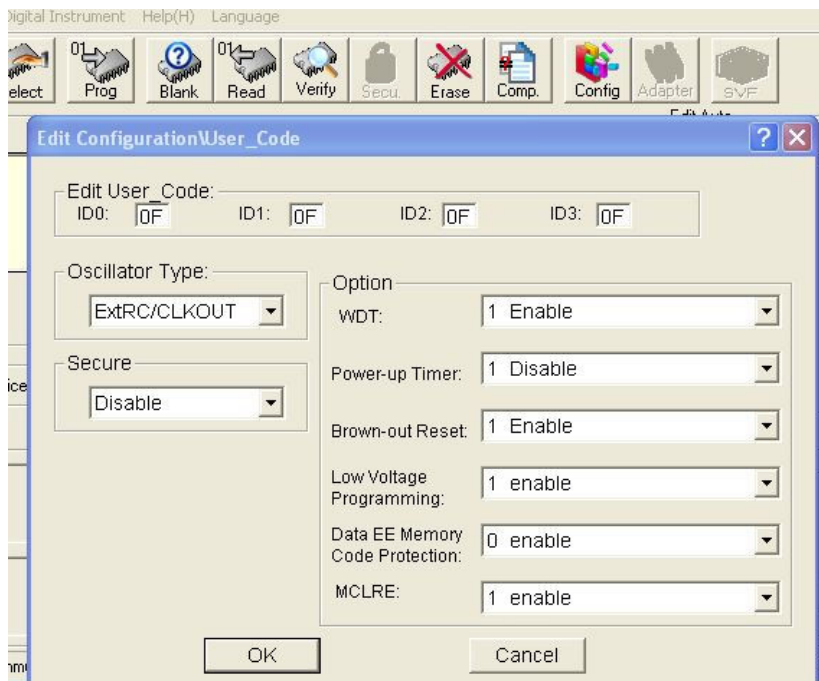
A beolvasott adatokból egy „Check Summa”-t, azaz ellenőrző összeget is kiszámol a program, ez látható a SUM=xxxx . (Az ellenőrző összeg lényege, hogy valamennyi adat bájtot összegzi a program, aminek az eredménye a „cskksum”. Ebből következik, hogy ha csak egy bájt értéke is változik, akkor az adott tartalomhoz tartozó ellenőrző szám is más lesz.

Adat memória, konfiguráló bitek:

Vannak több programozható területtel rendelkező IC-k, ahol a „Code Buffer” mellett van EEPROM memória is, vagy/és konfiguráló bitek. Ezek az „Edit” menü „Edit E_field”, és „Edit Configuration\User_Code” almenük alatt tekinthetők meg, illetve módosíthatóak. Elvileg. Mert nekem csak a Config biteket sikerült szerkesztenem pl. a Microchip PIC-ek esetében, azoknál a típusoknál is, ahol pedig van eeprom adat memória is.



Pl. a Microchip PIC-eknél program memória mellett van egy működést beállító bitmező is. Ezek tartalma tekinthető meg a " Config" ikonra kattintva.



Vannak IC-k, amelyek tartalma titkosítható – azaz később nem olvasható ki belőlük a program – ezeknél a „Security”, azaz titkosítás is kijelölhető. (Pl. a GAL-oknál, PALCE-knél aktív az ikon. A mikrokontrollereknél a konfiguráló biteknél kell beikszelni a program memória kiolvasásának tiltását.)



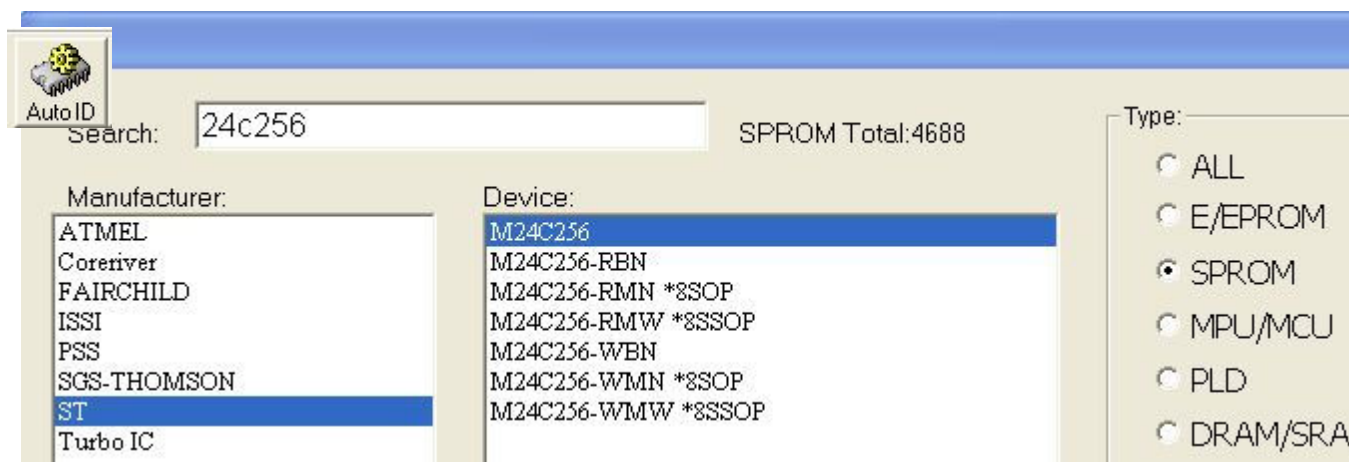
A programozandó IC kiválasztása:

Ahogy arról az előbbieken már szó volt, a "Device" menüben jelölhető ki, milyen IC-vel akarunk foglalkozni.



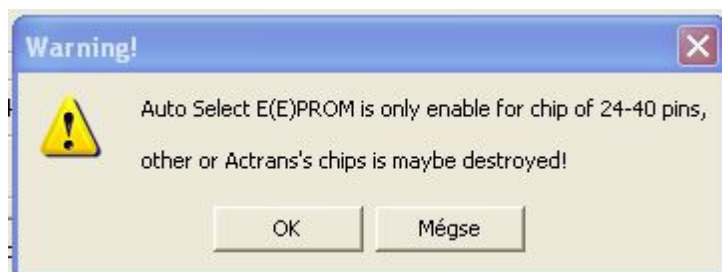
Egy egér kattintással ki kell jelölnünk, hogy milyen IC csoportból akarunk választani – EPROM, EEPROM, mikrokontroller, soros EEPROM, stb.

A kezelendő IC típus gyors beállítását segíti a "típus keresés". Tehát, ha pl. egy ST gyártotta M24C256 EPROM-ot akarunk égetni, akkor a 24C256-ot beírva a kereső sorba, akkor megjelöli az „SPROM” típust – a Serial Prom rövidítéseként, bár szerintem a serial eeprom jobb megnevezés lenne - és megjeleníti a gyártó cégeket. Ezután tudjuk gyártók típusaira szűkíteni a kínálatot, és végül választani:



EPROM és EEPROM típusoknál kiadható az automatikus chip azonosítás.

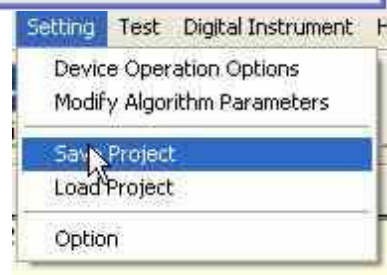
De – erre figyelmeztet a program – csak 24-40 kivezetéses (E)EPROM esetében adjuk ki ezt a parancsot, mert más IC- károsodhatnak!



Egy „Project” mentése:

Ahogy arról szó volt, vannak olyan programozható IC-k, amik egy csomó programozható beállítási lehetőséget is tartalmaznak.

A programnak van egy nagyon hasznos funkciója, a Setting menüben a „Load Project” és „Save project”. Itt minden, a programozáshoz szükséges beállítás elmenthető.



Tehát ha pl. beállítottuk egy mikrokontroller típusát, a beállító bitjeit, behívtuk a bele égetendő adatokat, akkor ezt mind egy helyre, egy „project”-be menthetjük. (És persze később beolvashatjuk.)

A programozandó IC behelyezése a foglalatba:

Ahogy a készülék dobozán is látható, az az „alapértelmezés”, hogy a foglalatban lefelé kell sorolni az IC-eket, azaz amennyire csak lehet, lefelé.

A PC program ezt annyira egyértelműnek veszi, hogy külön nem is jelzi ki.

Szintén egy ábra fogja megmutatni, ha adapter kell az IC-hez.

Az adapterekről később majd bővebben. A legtöbb plc32-es tokos IC-hez az 1:1 bekötésű, DIP32-PLCC32 adapter kell, a legtöbb PLCC44-es tokos IC-hez pedig egy DIP40-PLCC44 adapter.

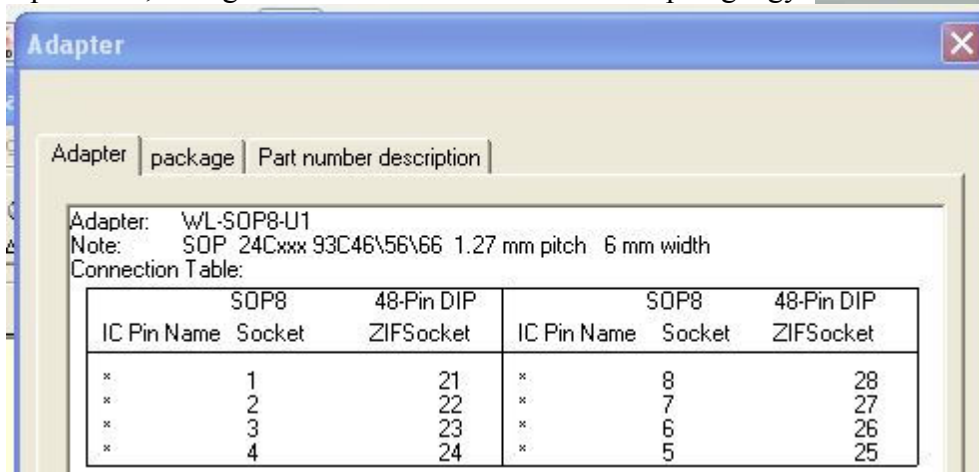
Az adaptereknél megjelenő ábra első látásra ijesztőnek tűnik, pl. egy smd tokos – SO8 tok – 24C16-ot kiválasztva ez jelenik meg:

1-es láb az égető 21-eséhez, 8-as a8-ashoz.

(???) De ha jobban ránézünk, gyakorlatilag 1:1 bekötésű adapter kell, amit a DIP tokos változathoz teljesen hasonlóan lefelé ütköztetve az 1-es lába valóban a karos foglalat 21., kivezetésébe kerül. (Egy ábra a sokkal érthetőbb lenne.)

A programozó képes meghatározni az esetleges kontakt hibát is a behelyezett IC-nél.

Ha a kiadott művelet – beprogramozás, kiolvasás - sikertelen, akkor megpróbálja azonosítani a lehetséges kontakt hibát, és azt ki is jelzi.

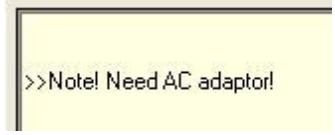


Tápellátás:

Mint arról korábban szó volt, a tápot az USB portról veszi az égető. Külső tápegység kizárólag akkor kell, ha olyan USB elosztón - HUB - keresztül csatlakoztatjuk, ami nem ad ki elég tápot.

Vagy, ha olyan IC típust választunk, ami esetleg több száz mA-ert vesz fel égetéskor.

De ilyenkor a program figyelmeztet is erre:



>>Note! Need AC adaptor!

Ami tetszik, és ami nem:

- Rengeteg EPROM , EEPROM típust tud, a mikrokontrollereknél az ATMEL „egzotikus” típusait is, plusz GAL-okat, PALCE-t is, és mindezt külön adapter nélkül. (A PLCC, smd IC-khez persze kell átalakító, hogy a karos foglalatba csíptethessük.)

- Kijelzi, ha nem ismeri fel az IC-t, hogy hol keressünk kontakt hibát. Ez pl. az smd adaptereknél nagyon hasznos.

- TTL IC-k, RAM-ok is tesztelhetők vele.

- Nagyon gyors.

- „project”-ként menthetjük a beállított ic típust, és a beállító biteket. Ez igen csak jól jön pl. az ATMEL mikrokontrollereknél, ha végre sikerült végre belőni a processzor működését meghatározó beállításokat.

- EPROM, EEPROM-ból ki tudja olvasni a chip gyári azonosítóját, és így beállítani az IC típust. (Már amelyiknek van ilyen.)

- A szoftver lehetne kicsit „kezesebb”. Pl. a hibajelzéseket írhatná pirossal, vagy a szöveg üzenet mellett egy képpel is kijelezhetné, hogy – ha kell - milyen adaptert vár a foglalatban.

– A PIC és ATMEL memóriáknál az eeprom adat memóriát nem lehet kiolvasni, írni, szerkeszteni. (Szerencsére erre ritkán van szükség.)

- **A WP-380 a PSOP44 tokos IC-ket nem kezeli.** Nem sok ilyen IC van, de az autó chiptuninggal foglalkozóknak fontos lehet a 16 bites 29F400/800 típus, márpedig WP-380 csak a TSOP48 tokozásút kezeli, a PSOP44 tokost nem. A VP-590, és a VP-990 -el égethetjük a PSOP44 tokos IC-ket is.

Összefoglalva nagy tudású, gyors égetők, elképesztő alkatrész választékkal, a kezelő szoftver néhány apróbb szépséghibájával.

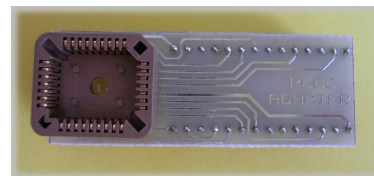
Adapterek PLCC, smd tokozású EPROM-okhoz:

Adapterek PLCC tokozású IC-khez :

Az adapterek lehetővé teszik a PLCC tokos epromok, mikrokontrollerek kiolvasását, égetését. Kiválasztjuk az IC típust – mivel DIP átalakítót használunk, a DIP tokos verziót jelöljük ki – majd az adaptert egyszerűen be kell dugni az égető karos IC foglalatába, aztán ugyanúgy égetjük, vagy olvassuk, mint a normál, DIP tokozású alkatrészt.

DIP28-PLCC32 adapter:

A 28 és a 32 lábú DIP tokozású EPROM-ok PLCC tokja egyaránt 32 kivezetésű, de a lábkiosztás eltér, ezért külön adapter kell pl. a 2764/128/256/512-es, és a 27010/020/040-es EPROM-okhoz.

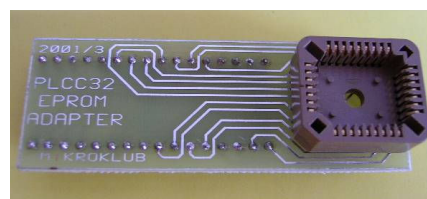


Tehát a DIP28-PLCC32 adapter kell pl. a 27256, 27512 EPROM-hoz. Szóval általában azokhoz az IC-khez, amelyeknek 28 kivezetéses a DIP tokozású kivitele.

Egy fontos dolog: Az adapter ugye a DIP tok kiosztására alakítja a PLCC tokozást. Tehát nekünk, ha a programozó külön felkínálja a típusválasztékban a DIP, és a PLCC tokozású változatot, akkor nekünk a DIP-est kell választanunk.

DIP32-PLCC32 adapter:

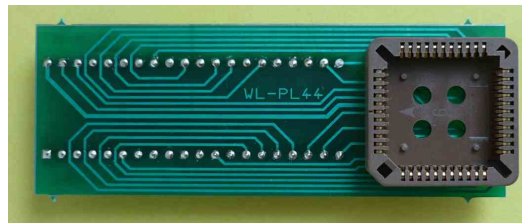
A legtöbb PLCC32-es tokos IC-hez az 1:1-es DIP32-PLCC32 adapter kell.



DIP40-PLCC44 adapter:

A wellon programozókhoz a PLCC44 tokos IC-k többségéhez egy DIP40-PLCC44 adapter kell.

Erre WL-PL44 néven hivatkozik a program. Ilyen adapter kell pl. a 16 bites 27c1024/4096 epromokhoz, az ATME1 89C51, 52 mikrokontrollerekhez. (Általában olyan IC-k plcc tokos változatához, amelyek a DIP tokos verziója 40 kivezetéses.)



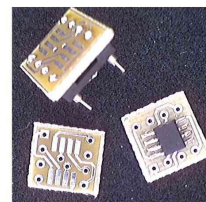
PLCC kisedő csipesz:

Amilyen könnyű eltörni a plcc IC foglalatot az IC kifeszetezése közben, olyan nehéz azt kicserélni a panelen... Az ilyen balesetek elkerülését segíti a plcc kisedő csipesz:



Adapter smd tokozású IC-khez:

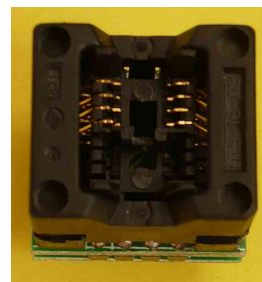
A 8 lábú DIP IC tokra forrasztott panel végszükségletben, mint "fapados" SMD adapter is használható programozó készülékekhez. Értelmszerűen a DIP tokot csíptessük a karos foglalatba, az SMD tokot pedig az olvasás vagy programozás alatt szorítsuk a panel SMD föliázatára.



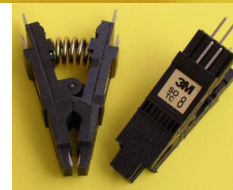
DIP8-SMD8 ZIF foglalatos programozó adapter:

Az előbbi egyszerű adapter ZIF IC tokos megfelelője. (A "zero force insert" a könnyített IC behelyezést jelenti most.)

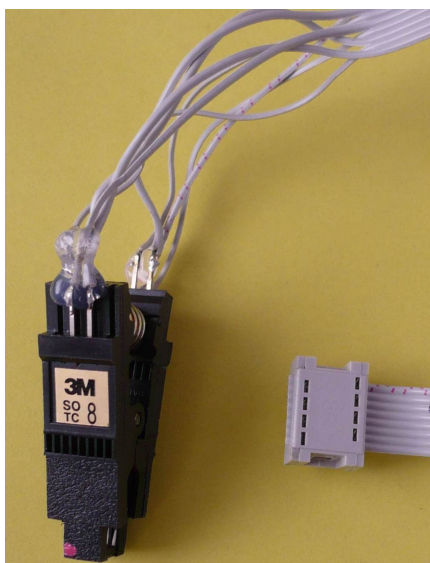
Szintén az SO8 tokos IC-khez használható. A tokozatot felülről megnyomva, felemelkedik a nyolc "lábacská" ami a behelyezett smd alkatrész kivezetéseit fogja meg. Az IC így behelyezhető, a rugós befogást visszaengedve, az aranyozott érintkezők megfogják az IC lábakat.



Szintén az SMD IC-k kezelését könnyítik meg az SMD csipeszek. A csipeszt 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.)



egy



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 áramkörüi környezettől, azaz mi csatlakozik az eeprom lábaira - általában egy mikrokontroller - és a programozó tudja-e annak ellenében mozgatni a kivezetések szintjét.

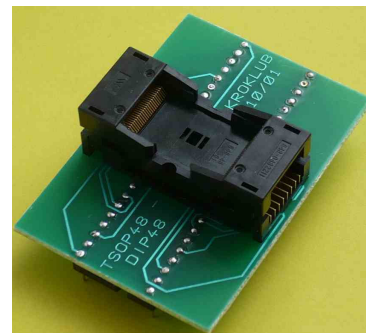
Valamint még a tápellátás "kritikus". Azaz hogy a programozó által adott 5 volt elég-e a panel számára. (Hiszen nem csak az eepromot fogja a tápot terhelni, hanem minden 5 voltra csatlakozó IC.)

Szóval lehet hogy 1-2-3 lábat "fel kell emelni" a panelről. Vagy - ez a biztos - le kell forrasztani az IC-t a panelről, és úgy "meg csípni".

Adapter TSOP48 tokozású IC-khez:

Az adapterkártya lehetővé teszi a tsop48 tokozású IC-k kiolvasását, beégetését. (Pl. 29F200, 29F400, 29F800) Az adapterbe kerül az EEPROM, majd azt egyszerűen be kell dugni az égető 48-as karos IC foglalatába, ezután pedig ugyanúgy égetjük, mint egy normál, DIP tokozású alkatrészt.

Az adapter teljes leírása a TSOP48.PDF-ben olvasható.



Kapcsolódó dokumentáció, szakirodalom:

A 27xxxx EPROM-okról, és azok törléséről szól az EPROM.PDF, a soros EEPROM-okról az EEPROMOK.PDF. (A CD-n a MIKROKLB\MIKROPO\DOCUMENT könyvtárban találhatóak.)

Egy UV fényes törlő lámpa leírása található az UVTORLO.PDF file-ban. (A CD-n a MIKROKLB\UVERASER könyvtárban.)

A PLCC EPROM-okhoz használható adapterekről PLCCEP.PDF szól. (A CD-n a MIKROKLB\MIKROPO\ADAPTERS\PLCCEP könyvtárban találhatóak.)

Az előbbi leírások, programok letölthetőek a lenti honlapcímről, vagy megtalálhatóak a „mikroklub cd”-n.

Végül nincs más hátra, mint hogy sok sikert kívánjak a használathoz. Torkos Csaba, 8100 Várpalota Táncsics u. 7. Telefon, napközben, 8-16 óráig: 06/88/473-784, vagy - egész nap, 8-22 óráig - a 06/30/9 472-294 számon. Email: mikroklub@vnet.hu Internet: <http://www.mikroklub.hu> , <http://www.eprom.hu>