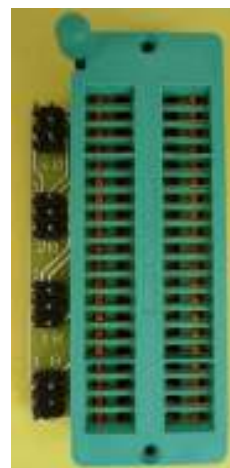


Programozó adapter MICROCHIP PIC mikrokontrollerekhez
MICROCHIP ICD2 programozó/debuggerhez
PICKIT2 programozóhoz
Willem égetőhöz

Az újabb kiadású mikrokontrollerek többsége tartalmazza a soros programozás lehetőségét. A lényeg, hogy a betöltendő adatokat sorban, bitekre bontva léptetjük be a mikrokontrollerbe. Mivel ez az algoritmus csak pár IC kivezetést igényel, lehetőséget ad arra, hogy a mikrokontrollert az áramkörben - in circuit - programozzuk. Ez nagyon előnyös, pl. programfejlesztésnél, mivel nem kell a mikrokontrollert minden egyes program verzió kipróbálásához kiemelni a foglalatából az égetéshez.

Az előbbiekből adódik az is, hogy a programozó áramköri hardver is nagyon egyszerű lehet, nem is csoda, hogy nagyon sok soros IC programozó jelent meg az interneten, a kereskedelemben.



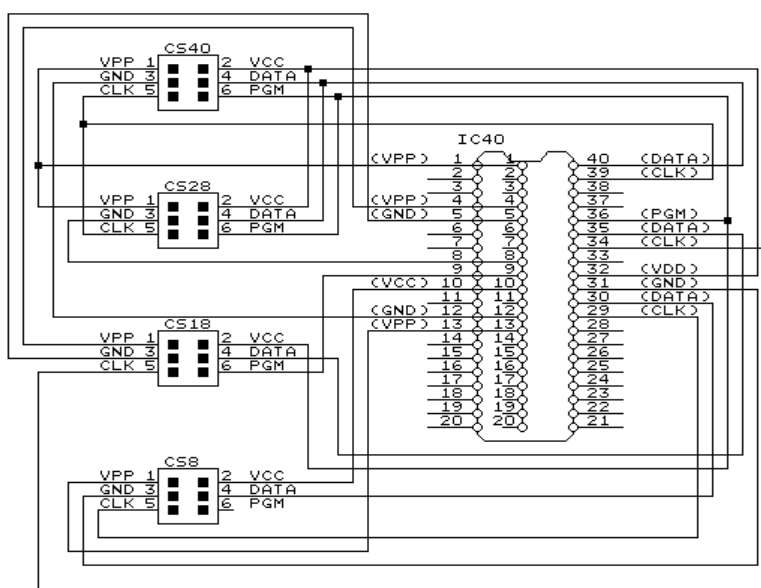
Több IC programozó már csak a csatlakozási pontokat biztosítja a PIC programozáshoz. Ez elég is ha pl. áramkörben programozzuk a mikrokontrollert, de a DIP tokos IC-k "szokásos" programozásra - berakom az IC-t a foglalatba, beprogramozom, kivesszem - önállóan nem használhatóak. De ezzel az adapterrel igen...

Az áramkör:

A gyakorlati használatban a programozót csatlakoztatni kell a mikrokontrollerhez. Konkrétan az RB6, RB7, az MCLR kivezetéseken keresztül folyik a programozás, és persze biztosítanunk kell a tápot. (GND, VCC) Az egyik lehetőség az, hogy a fejlesztendő panelt eleve felkészítjük az ICD, vagy más soros programozó csatlakoztatásához, azaz a szükséges kivezetéseket egy csatlakozóra kivezetjük. (Példa erre a "mikroklubbos" PIC DEMO panel, vagy a PICMIKRO mikrogép) Vagy - és most erről lesz szó - kell csinálnunk egy kis adaptert, ami programozó kivezetéseit a PIC megfelelő kivezetéseire csatlakoztatja.

Az ismertetésre kerülő adapter bármely soros PIC programozóhoz használható, a csatlakozó kiosztása azonban az ICD-hez igazodik. (Szerencsére ezt a kiosztást megtartotta a PICKIT2, és a Willem programozó is.)

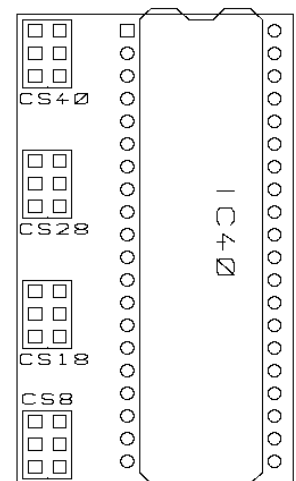
Az adapter a DIP tokos 8, 18, 28 és 40 lábú mikrokontrollerekhez használható. Ezek a széles vágatú programozó foglalatba csíptetendők. Csak hogy mindegyik tokozásnál más és más helyre esnek a programozó kivezetések. Ezt a problémát úgy oldja meg az áramkör,



hogy négy csatlakozó van az adapteren. A CS40-es csatlakozó a 40 lábú PIC-ek megfelelő kivezetéseire vezeti a programozó jeleket. Ugyanígy a CS28 a 28, a

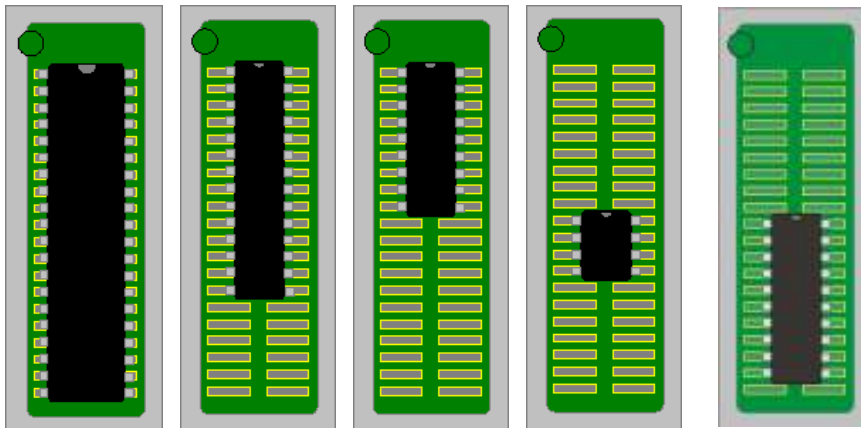
CS18 a 18, és a CS8 a 8 lábú PIC-ek megfelelő lábait köti össze a programozóval.

A programozó és az adapter összeköttetéséhez egy 6 eres szalagkábel használhatunk. A panelon 4 darab 3x2-es tűske került, amire tűskére nyomható 10-es szalagkábel csatlakozó kerülhet. Ahogy arról szó volt, a csatlakozópontok az ICD-nél szokásos sorrendben követik egymást. Tehát az első a VPP, ami a mikrokontroller MCLR lábára, a VCC és GND értelemszerűen a tápkivezetésekre, a DATA az RB7, a CLK pedig a mikrokontroller RB6 portjára csatlakozik. Némely típusnál az RB3 (PGM) lábat a programozás alatt alacsony szintre kell húzni, ezért ez is be lett kötve. (Ha megnézzük az ICD kapcsolási rajzát, akkor látjuk, hogy itt egy ellenállás van a GND felé.)



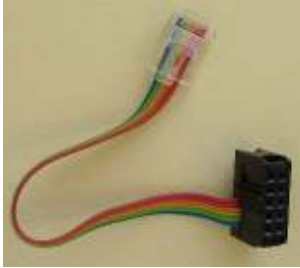
A 40, a 28, és a 18 lábú PIC-eket úgy rakjuk a foglalatba, hogy az 1-es lábuk a programozó foglalat 1-es nyílásába kerüljön.

A 8, és a 20 lábúakat, pedig úgy, azok 1-es lába a foglalat 10. nyílásába menjen:

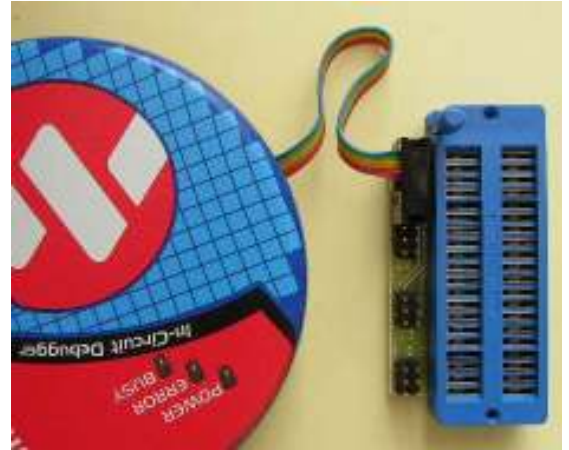


MICROCHIP ICD2

A MICROCHIP PIC-ekhez - többek között - az MPLAB ICD is használható, mint soros programozó. Az ICD2 szinte az összes újabb típust "ismeri", és mivel az ICD2 működtető programját folyamatosan fejleszti a MICROCHIP, bízhatunk abban, hogy a később megjelenő típusokkal se lesz gondunk. (Az MPLAB tartalmazza a „firmware”-t, amely az újabb verziókkal frissül.)



Az adaptert az RJ6-os "telefondugó" végű kábellel köthetjük az eredeti, MICROCHIP gyártmányú ICD2-höz.



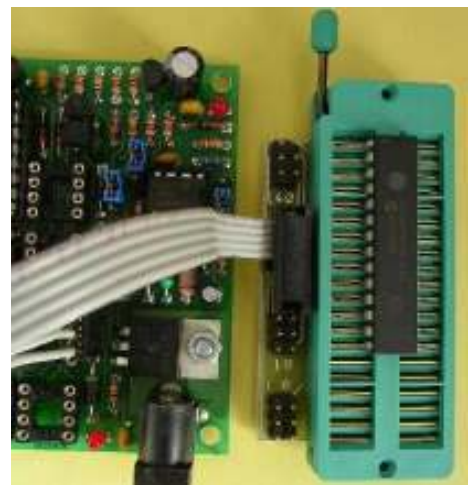
PICKIT2

Szintén MICROCHIP fejlesztésű soros programozó. Egy 6-os tűskealjzat a kimenő csatlakozó.



Willem programozó

Ennek a népszerű égetőnek is van soros PIC égető csatlakozója, egy 6-os tűske. A csatlakozó kábel egyik oldala így egy 6-os hüvelysor, a másik felén a normál, 10-es szalagkábel csatlakozó.



A használat:

Kössük össze a programozót, és az adaptert. Az előbbiek szerint, az adapteren azt a csatlakozót használjuk, amelyik az adott lábszámú mikrokontrollerhez tartozik.

Ha tehát egy 28 lábú, mondjuk 16F876-ot akarnánk programozni, akkor így fog kinézni az összeköttetés:



Adjunk tápot a programozónak.

Helyezzük be a PIC-et a programozó foglalatba.

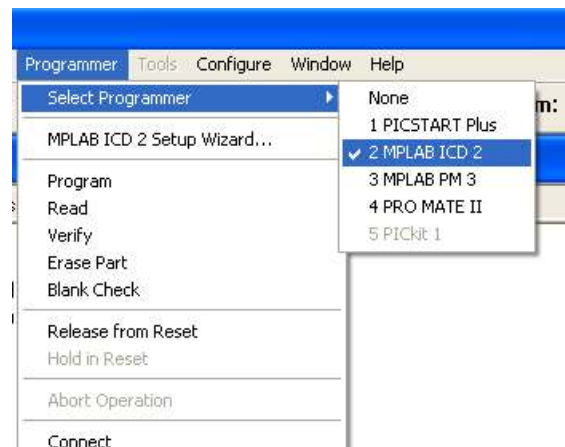
Ha az ICD a programozó, akkor ne felejtsük el rákapcsolni a tápfeszültséget a csatlakozóra.

Mondjuk, a PICOK04.HEX file-t égezzük be egy 16F877-be! (Ez egy "futófény" programcska - a PICDEMO panelra készült mintaprogramok egyike - ami az RB0-7 portokat kapcsolgatja be/ki, sorban egymás után. A hex fájl a mikroklub CD-n a mikroklb\picokat könyvtárban található) Akkor a lépések sorban:

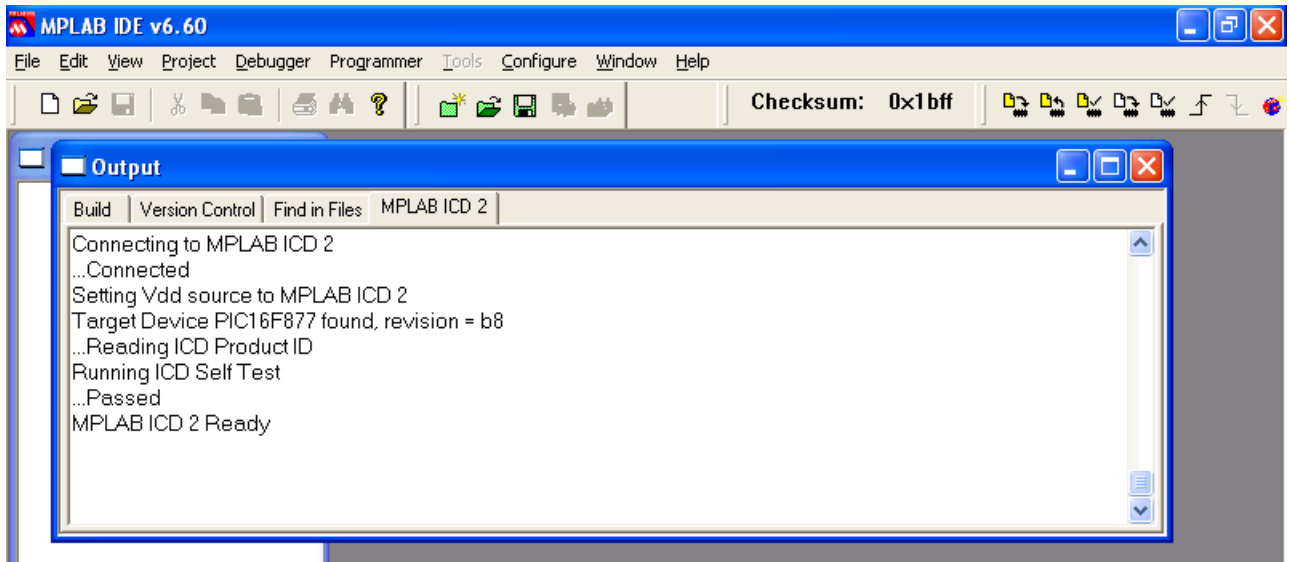
- Csatlakozzunk az ICD-vel a PC-hez, az égető adapterhez a CS40 csatlakozón, és biztosítsuk a tápot.

- Indítsuk az MPLAB-ot!

- Lépünk be a "Programmer" menübe, és ott jelöljük ki az "MPLAB ICD 2"-t:

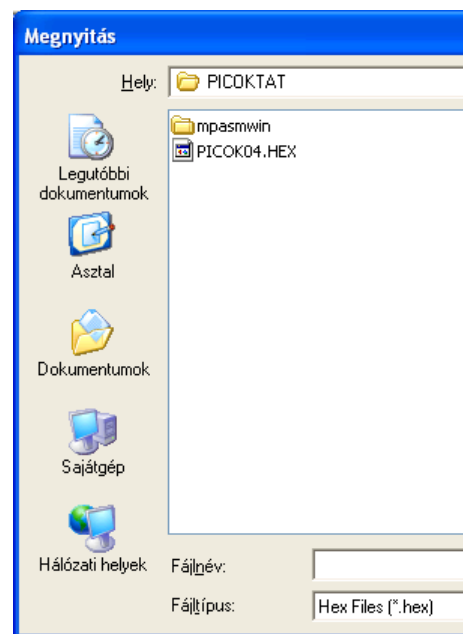
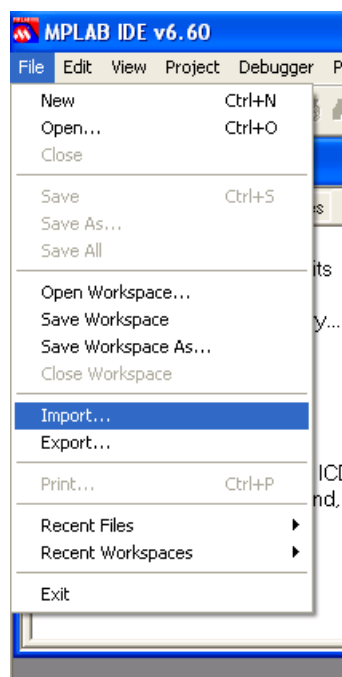


Az MPLAB felveszi (megpróbálja) a kapcsolatot az ICD-vel. Kicsit pislákol az L2 LED, majd a monitoron pedig megjelenik az ICD ablak:

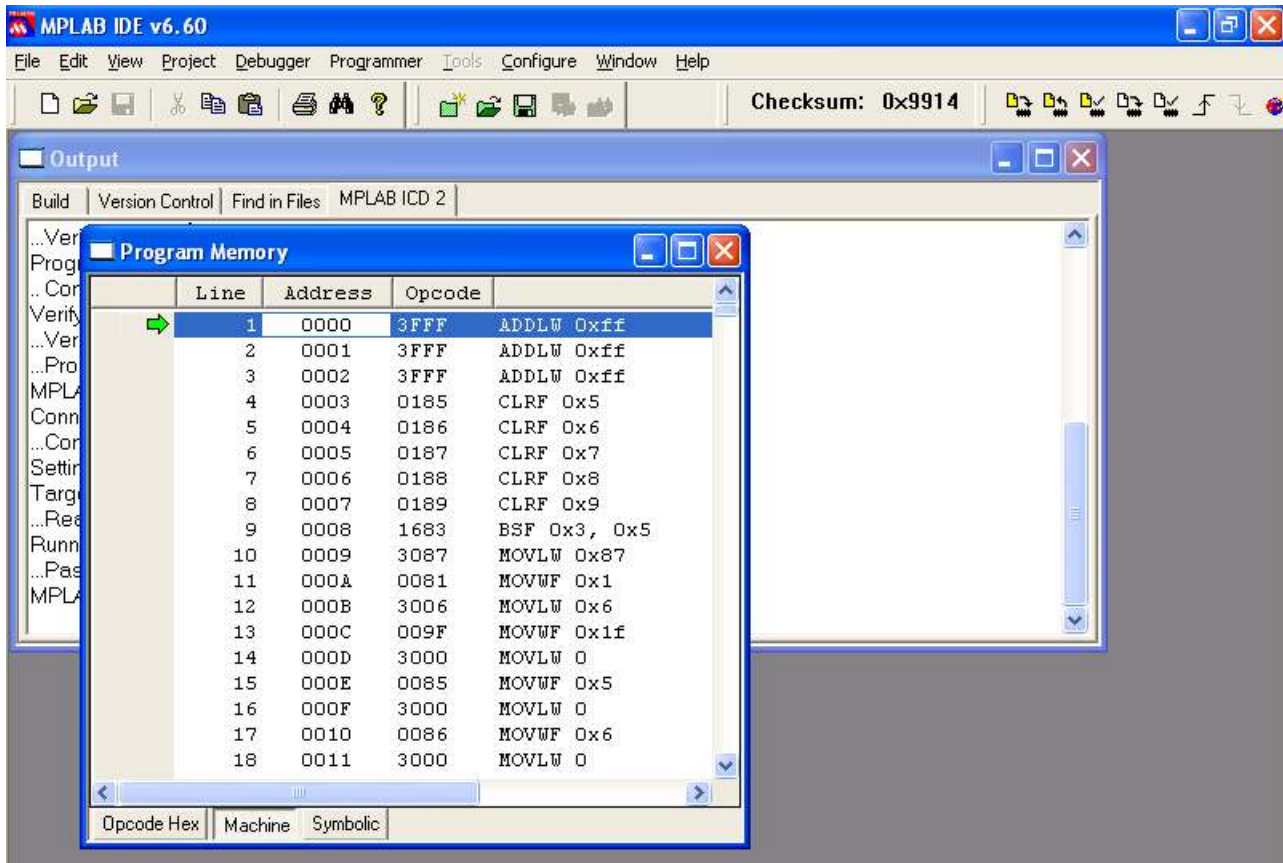


Olvassuk be az égetendő .HEX fájlt! A „File” menü „Import” menüjére kattintva, tudunk a fájlok között válogatni. Keressük meg a PICOK04.HEX-et :

(Ne a File Open menüt használjuk, ez esetben csak az Import a megfelelő!)

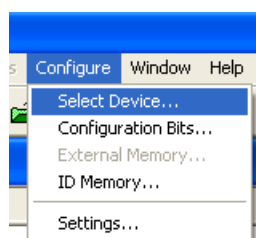


Ha beolvastuk be a file-t, akkor megnyithatunk egy ablakot, hogy lássuk mit is olvasunk/égetünk be! Menjünk a "View" menübe, azon belül a "Program Memory"-ra kattintsunk:

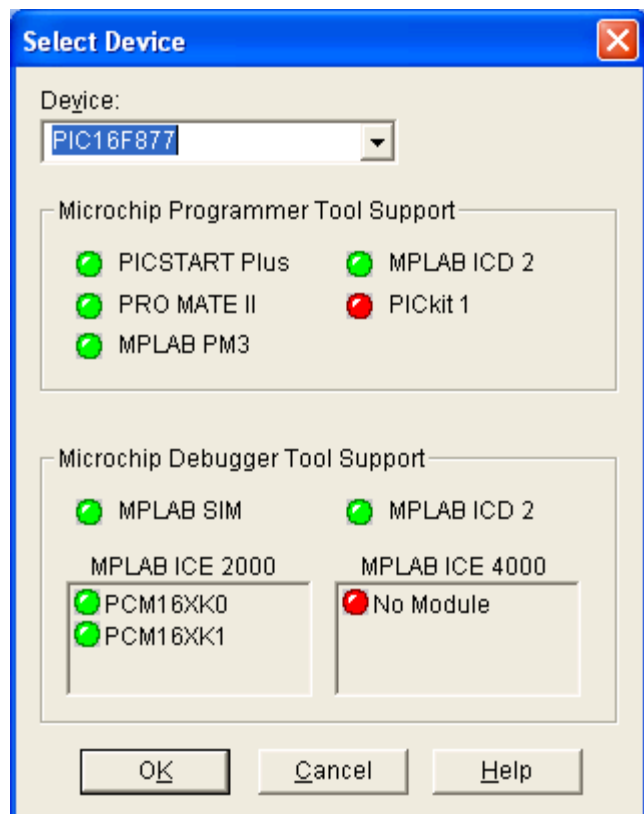


Ahogy fent látjuk, megnyitott ablakban megjelennek az égetendő adatok, sőt, az MPLAB mindjárt vissza is fordítja az adatokból az assembly utasításokat, és azokat is megmutatja.

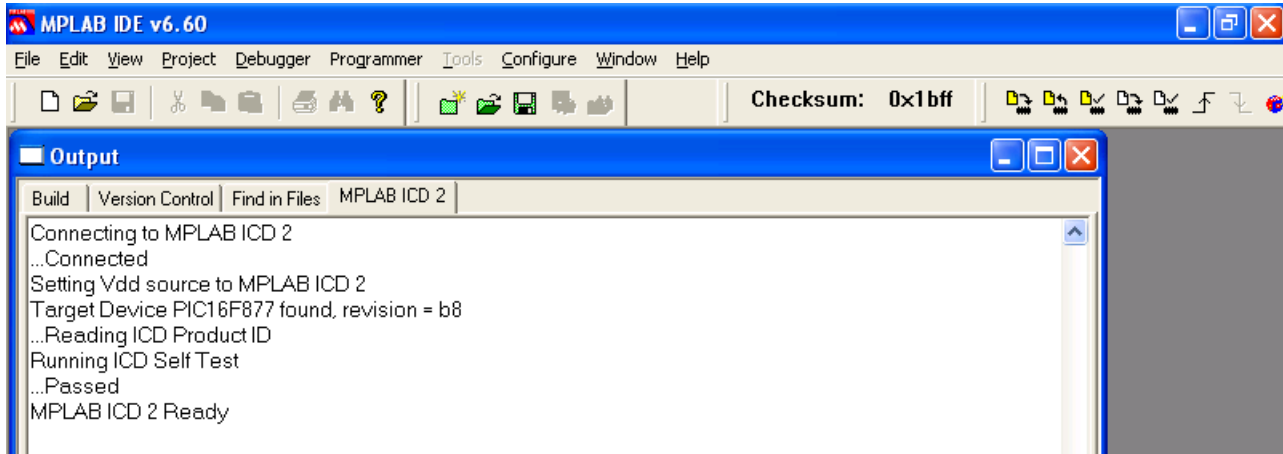
Válasszuk ki a PIC típust, amit égetni akarunk! Kattintsunk a „Configure” menü „Select Device” ablakra:



Ott láthatjuk éppen milyen PIC van beállítva, és válasszuk ki a PIC16F877-et!



Az ICD rögtön le is ellenőrzi, hogy tényleg a beállított IC lóg a kábel végén:

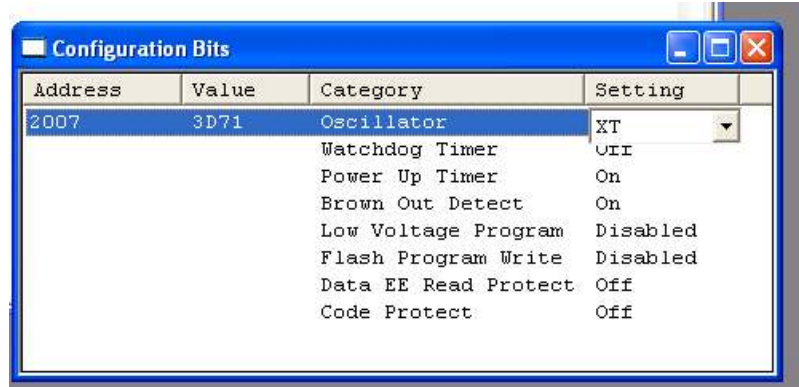


Ahogy fent látható, az ICD 16F877-et talált, ez rendben is van.

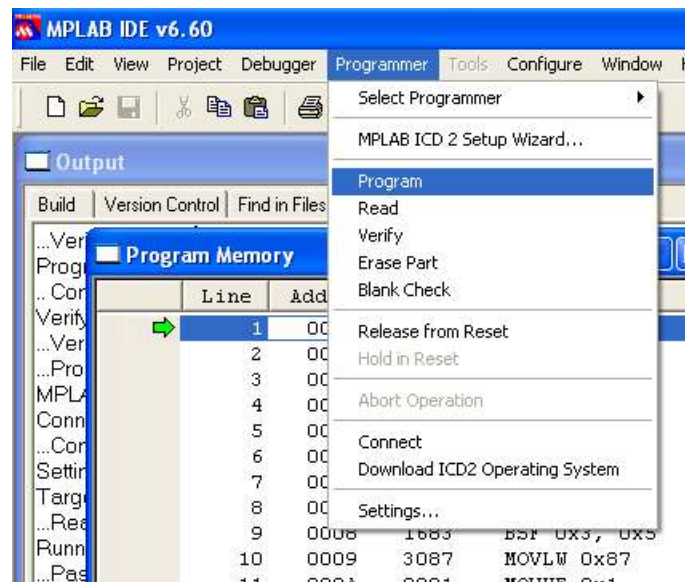
Megnézhetjük, valamint ha akarjuk, állíthatjuk a fuse-okat:

Szóval végezzük el mikrokontroller kiválasztást, és a "fuse" beállításokat. Ha

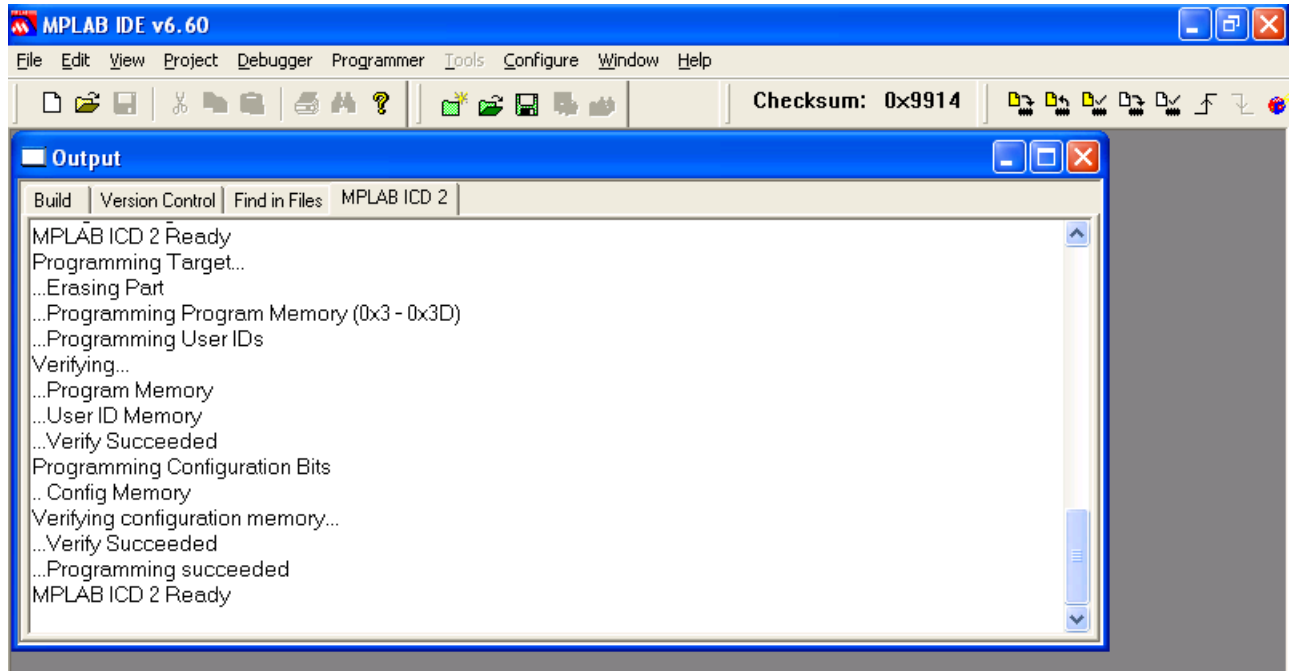
titkosítani akarjuk a tartalmat, akkor válasszuk "code protect" On opciót. (Itt több lehetőség közül választhatunk, a teljes programmemória, vagy annak csak egy részének a titkosítása is kijelölhető.)



És akkor következhet a programozás! Adjuk ki programozás parancsot, azaz kattintsunk az ICD ablak Program pontjára.



Az ICD törli, majd programozza a programmemóriát, az azonosító bájtot, aztán ezt visszaellenőrzi, és beégeti a fuse biteket. Az ICD státusz ablakában, sorban a következőket fogjuk látni:



Ha titkosítva volt az IC, akkor előtte törölni kell, de ha be van ikszelve az "Erase before programming", akkor úgyis mindig van egy törlés.

Ha az előbbieket láttuk, akkor minden rendben.

Kapcsolódó dokumentációk:

A témához kapcsolódik az ICD2 panel dokumentációja. (ICD2.PDF)

A „DAVID TAIT” , és a „Willem” programozónak is van soros PIC programozó csatlakozója, így természetesen ezekkel is használhatjuk ezt az adaptert. A különbség csak a csatlakozóban van, ez esetekben a programozón egy egysoros tűske csatlakozó van.

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

Végül nincs más hátra mint hogy sok sikert a használathoz. 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>