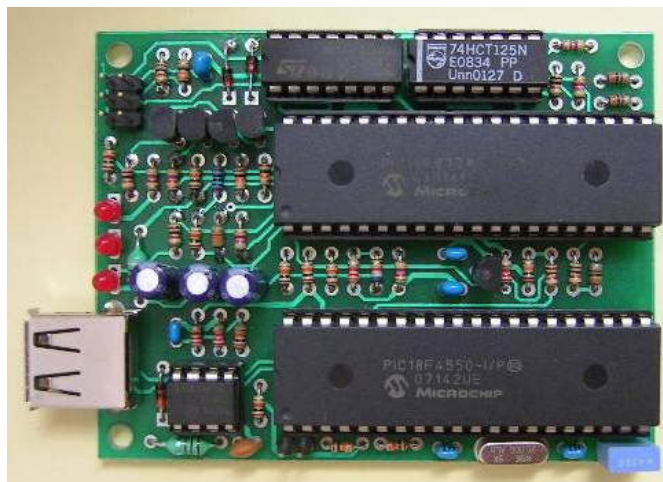


MICROCHIP PIC ICD2 klón (USB portos)

Az MPLAB ICD-2 a Microchip által készített fejlesztő eszköz, amely a PIC mikrokontrollerekhez használható.

Mire is jó ez? Az ICD az "in circuit debugger" rövidítése, azaz egy olyan fejlesztő eszköz, ami az áramkörbe helyezve megkönnyíti a program tesztelést. De mint PIC mikrokontroller égető is tökéletes, hiszen a Microchip folyamatosan fejleszti az ICD2 programját, így a legújabb típusok is bekerülnek a menübe.



Amit lehet vele:

- program letöltés - beégetés - a célkészülékben elhelyezett mikrokontrollerbe, azaz mint PIC égető is funkcionál, igaz csak az újabb típusokra. Az ICD-2 a PIC16Fxxx a 18Fxxx, 12Fxxx családhoz jó, a régi, pl. 16Cxxx típusokat, de a korábbi 16Fxxx típusok többségét sem kezeli. (Pl. a 16F627A benn van a menüjében, de a 16F627 nincs.)

- alkalmas egy program valós idejű - "real time" - nyomkövetésére, vagy lépésenkénti - "step by step" - végrehajtása, és a regiszterek értékének vizsgálatára.

Soros port/USB port, az eredeti, és „mikroklubbos” ICD2:

A Microchip közzé tette az ICD2 áramkör korábbi verziójának a kapcsolási rajzát. Itt még egy spec. illesztő IC végezte az USB kommunikációt. Ezt később lecserélték az USB-s 18F4550 mikrokontrollerre. Az áramkört többen megpróbálták után építeni, az Interneten több dokumentációt találtam e témában. Ezeket átnézve, átalakítva, a speciális IC-ket kiváltva készítettem el először egy soros porton működő, majd a most ismertetendő USB portos „klont”.

Az eredeti, és az utánépített ICD2 összevetése:

A működtetés, a funkciók, a kezelt IC típusok teljesen azonosak, tekintve, hogy a vezérlő mikrokontroller kezelő programja is megegyezik az eredetivel. Ebből az is adódik, hogy a firmware az MPLAB újabb verzióival frissül, így a jövőben megjelenő új PIC típusok is kezelhetők lesznek majd.

A MICROCHIP ICD2 a soros porton, és az USB porton is csatlakoztatható a PC-hez. Az itt szereplő csak az USB porton. (És van egy egyszerűbb verzió, ami csak a soros porton. Erről külön leírás szól.)

A MICROCHIP változatnál a céláramkörrel egy speciális, 6-os telefoncsatlakozón keresztül tudunk kapcsolatot teremteni. Ennél a klónnál egy szokásos tűske/szalagkábeles csatlakozó van erre kiépítve. (Ez szerintem sokkal célszerűbb.)

Soros port / USB port:

Az előbbiek szerint, működtethetjük az ICD2-t a soros, vagy az USB porton is.

Az USB port előnye a gyorsaság. A programozás funkcionál ez nem is annyira lényeges, itt olyan gyorsan lezajlanak a folyamatok, hogy "figyelni kell", hol is tartunk. Ha ellenben debuggolásra is használni akarjuk az áramkört, akkor egyértelmű a nagy sebességű adatforgalom előnye.

A soros porti verzió akkor megfelelő nekünk, ha elsősorban mint égető fog funkcionálni az eszköz. De van egy előnye is a soros portos működésnek: a tapasztalatom szerint stabilabb a kommunikáció. Ha valamiért megszakad az adatforgalom - ki/bekapcsoljuk az áramkört, stb. - a soros porton egy egérekattintással létrejön a "connect", az USB portnál ez a reset nyomkodás után is csak többszöri próbálkozásra sikerül, de van hogy ki/be kell lépni az MPLAB programba, sőt kifagyás is előfordul... (Biztos ami biztos, egy eredeti ICD2-vel is teszteltem a kommunikációt. Az eredmény ugyanaz.)

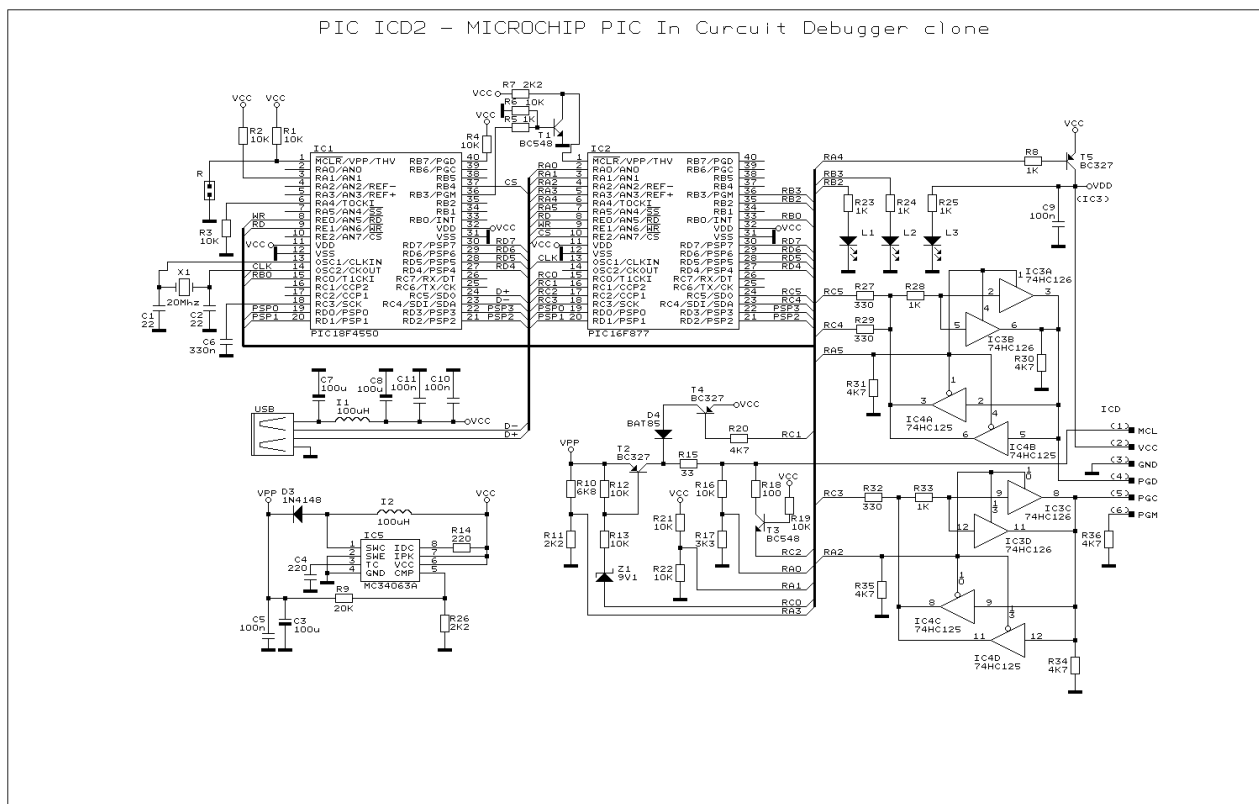
A használata:

A gyakorlati használatban az ICD kapcsolódik egy mikrokontrollerhez, általában a fejlesztendő áramkörben lévőhöz. Az egyik lehetőség az, hogy a fejlesztendő panelt eleve felkészítjük az ICD csatlakoztatáshoz - egy példa erre a "mikroklubbos" PIC DEMO panel - vagy csinálunk egy kis adaptert, a szükséges csatlakozókkal PIC és a fejlesztendő panel közé. (Ilyen is van a mikroklub kínálatban. Ennyi ingyen reklám egyszerre ...)

A debugger egyszerű, és olcsó felépítésű, cserébe áldozni kell a PIC erőforrásaiból. Az ICD a működéséhez, az MCLR/Vpp láb, és az RB6 és RB7 portlábakat használja a programozáshoz, és a kommunikációhoz, ezért ezek - debugger módban - a célkészülékben nem használhatóak. De, ha csak mint "égetőt" használjuk az ICD-t, akkor az égetés után az ICD-t leválasztva, a beégetett program természetesen tudja használni az előbbi portokat is.

Az áramkör:

Az áramkör két mikrokontrollert is tartalmaz. Az ICD funkciókat a működtető programmal beégetett 16F877-es, az USB-s kommunikációt a 18F4550 végzi.

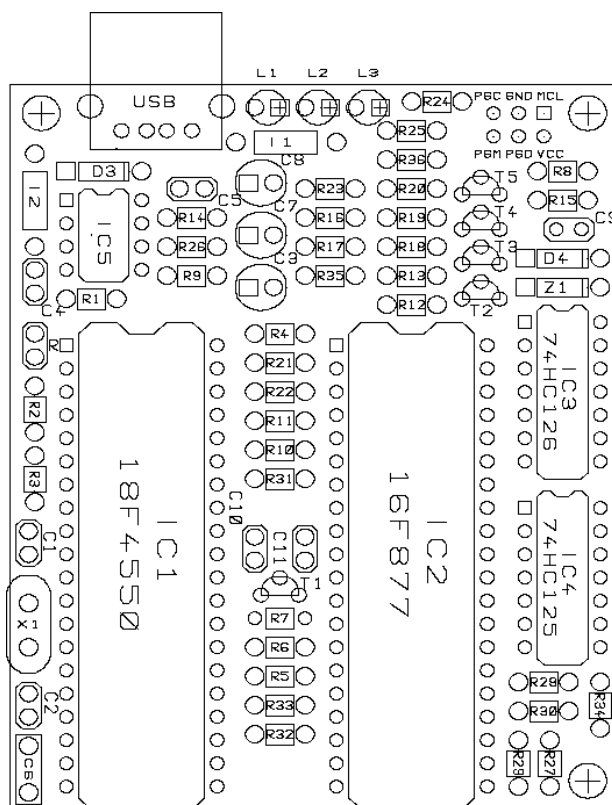


A PC-ICD összeköttetéshez egy USB kábelt használhatunk.

Az égetőfeszültséget az IC5-ös kapcsoló üzemű DC/DC konverter IC állítja elő.

A PIC égetésekor, annak MCLR lábára nulla, vagy öt, vagy a kb. 12.5 voltos égetőfeszültség kerül. Az égetőfeszültség ki/bekapcsolását a T2 végzi. Ha a T2 nyitva, a VPP lábon megjelenik az égetőfeszültség. Ha T2 zárva, de a T4 nyit, akkor a D4-on keresztül itt kb. 4.7 volt lesz. Ha pedig a vezérlő mikrokontroller RC2 portjának alacsony szintre kapcsolásával a T3-at nyitjuk, akkor az MCL kivezetés feszültsége pár tized voltra csökken.

Ha a kész áramkört dobozoljuk, akkor a legegyszerűbb, ha a "G747"-os jelű műanyag dobozt



használjuk, mivel a felfogató csavarok helye az alappanel sarkaiban ehhez vannak igazítva.

Beüzemelés, tápellátás, tesztek:

Jelen verzióban a tápot a PC-ről kapjuk, külső tápot nem kell - nem is lehet - csatlakoztatni.

Erről pár szó: a gyakorlatban bebizonyosodott - számomra legalábbis - hogy az USB-ről leszedni a tápot nem szerencsés.

Az 5 voltos táp a gyakorlatban inkább kb. 4.6 volt, mire bejut az áramkörbe.

Még rosszabb a helyzet, ha USB hubon keresztül csatlakozunk. Én ekkor már csak 4.3 voltot tudtam mérni.

Na ettől még működik az áramkör, de az önteszt már hibát ad.

```
ICD2 Warning: Invalid target device id (expected-0x1f3, read-0x1f0)
...Reading ICD Product ID
Running ICD Self Test
... Failed Self Test. See ICD2 Settings (status tab) for details.
MPLAB ICD 2 Ready
```

A "gondot" az okozza, hogy a kicsi tápfesz miatt az önteszt feszültségmérése is "elmászik", és mind a tápfesznél, mind az égetőfeszültségnél "magas" feszültséget mér a vezérlő 16F877-es, és ezt jelzi a piros betűs üzenetben.



És akkor még egy szépséghiba: az L3 LED jelzi, a csatlakozóra kikapcsolt tápfeszültséget. Csakhogy nekem ez folyamatosan világít, függetlenül attól, hogy a VCC csatlakozóra kiadom, vagy sem a tápfeszt, csak a fényerő változik.

Nem értettem. Aztán rájöttem, hogy az IC3 (74HC126) bemenetein keresztül jut feszültség az IC tápkivezetésére, ezért ég az L3 LED.

Az előbbiek miatt készül egy átdolgozott a verzió, ami e prototípus gyerekbetegségeit orvosolja.

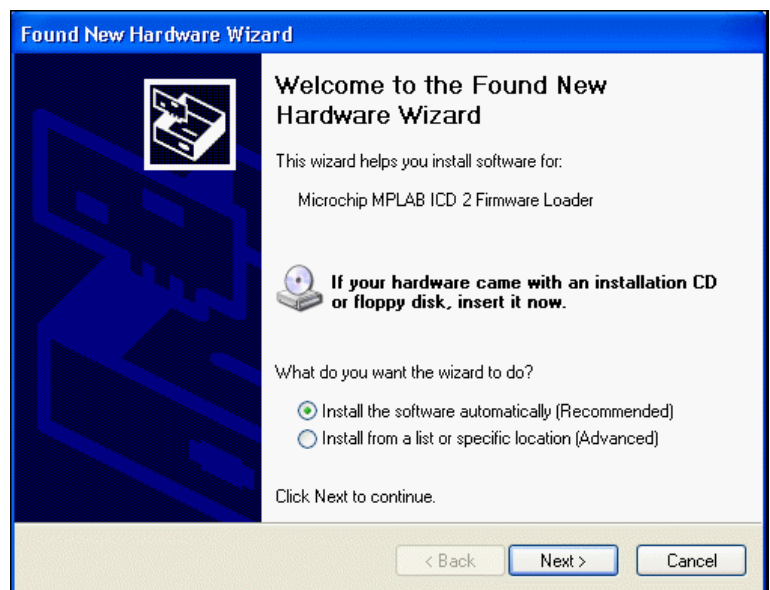
Az USB driver telepítése a windows xp esetén:

- Csatlakoztassuk az USB kábellel az áramkört, a windows "új hardvert" fog találni. Ne keressünk windows frissítést:

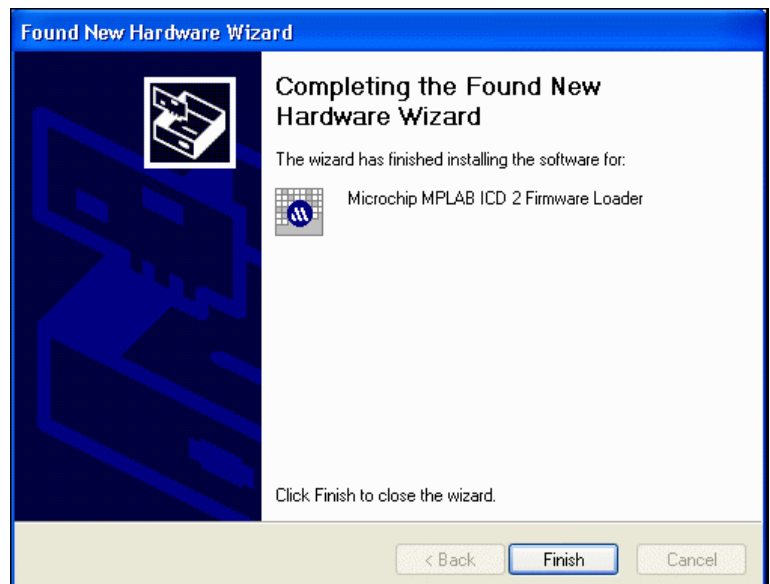
Ha nem jelenik meg ez az ablak, lehet hogy nem kap tápot az USB portról. (Ellenőrizzük le.)



A következő lépésben az automatikus telepítést válasszuk:



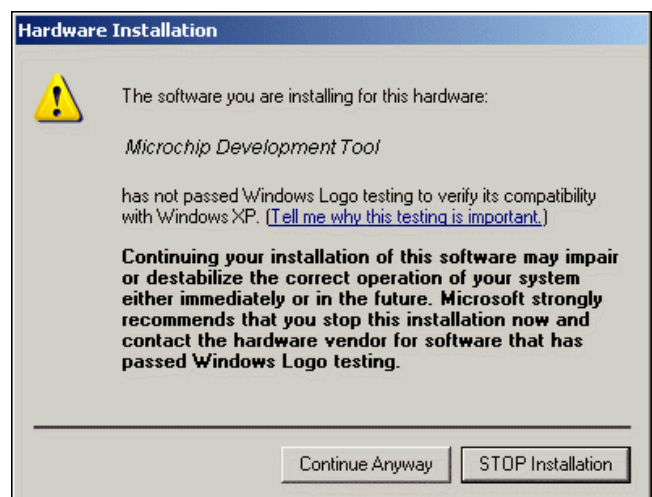
A windows megtalálja a betöltő



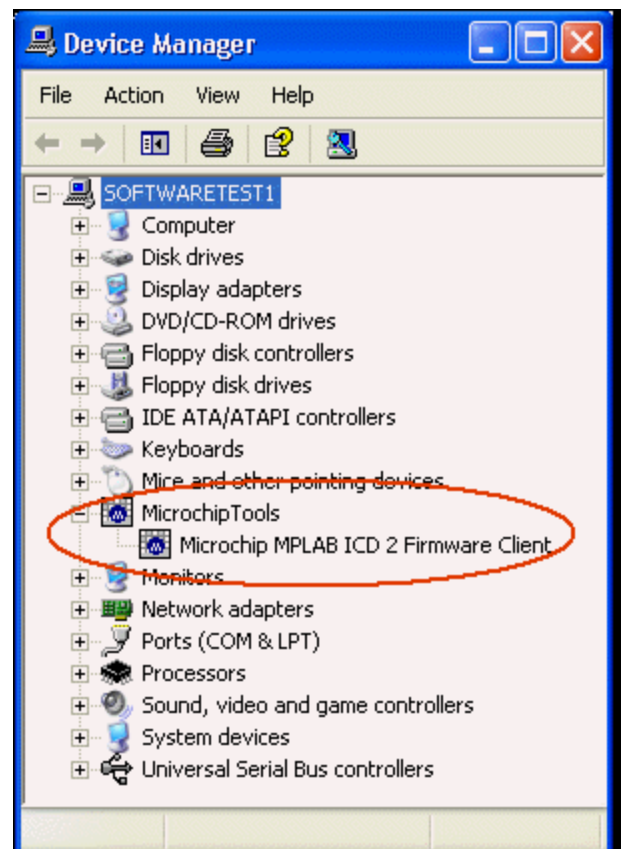
majd a "kliens" programot:



Azzal ne törődjünk, ha az XP figyelmeztet, hogy ez a szoftver nem esett át microsoft tesztjén, folytassuk a telepítést. (A Continue Anyway-t válasszuk.)



Végül az "eszközvezérlőben" ellenőrizzük le, hogy minden rendben:



Ha mint égetőt használjuk az ICD-t:

Az újabb kiadású mikrokontrollerek többsége tartalmazza a soros programozás lehetőségét. Ez 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 ICD is használható, mint soros programozó, igaz csak az általa "ismert" PIC típusokhoz.**

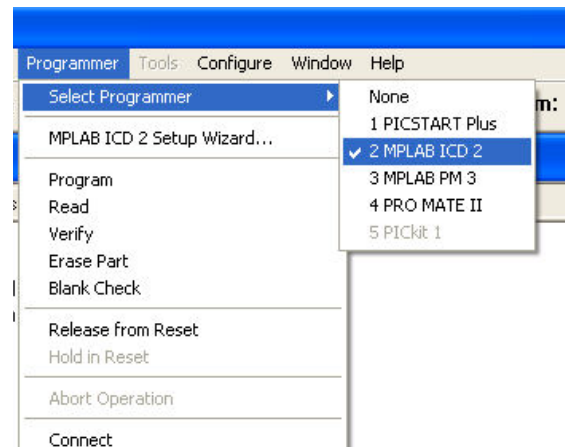
Lássuk ezt egy példán keresztül, mondjuk a PICOK04.HEX file-t égezzük be egy 16F877-be! (Ez egy "futófény" programocska - a PICDEMO panelra készült mintaprogramok egyike - ami az RB0-7 portokat kapcsolgatja be/ki, sorban egymás után. A hex file a mikroklub CD-n a mikroklub\picoktat könyvtárban található)

Akkor a lépések sorban:

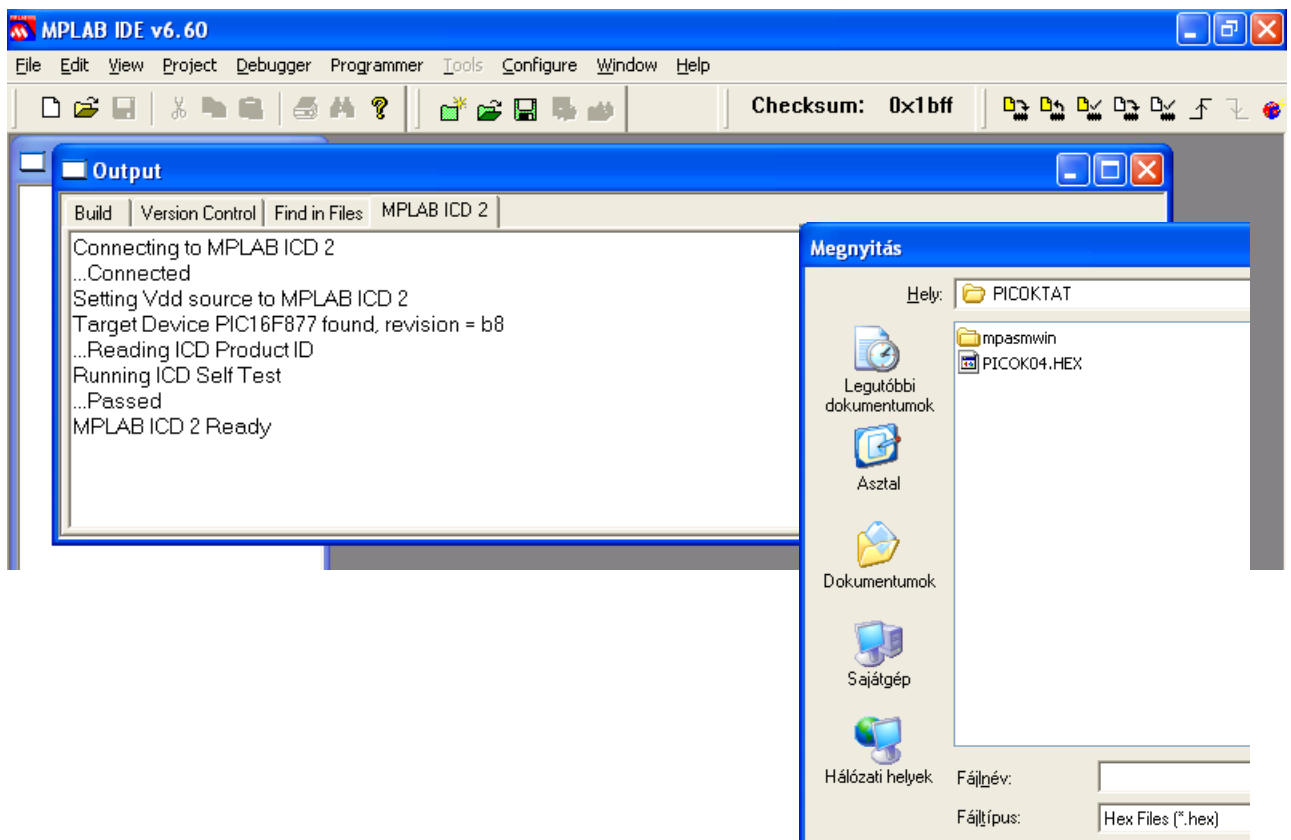
- Csatlakozzunk az ICD-vel a PC-hez, az égetendő áramkörhöz, vagy PIC-hez, és biztosítsuk a tápot.

- Indítsuk az MPLAB-ot!

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

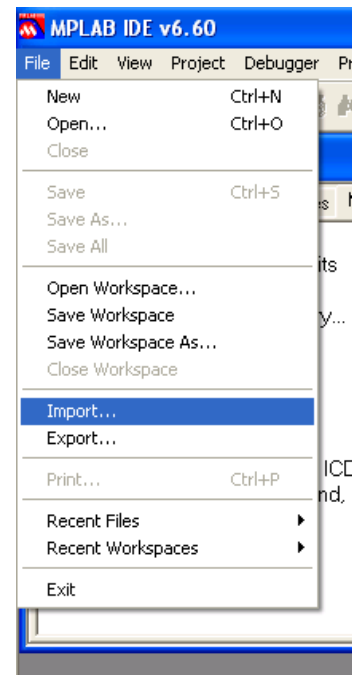


Az MPLAB felveszi (megpróbálja) a kapcsolatot az ICD-vel. Kicsit pislákol az L2 LED, majd a monitoron pedig megjelenik az ICD ablak:

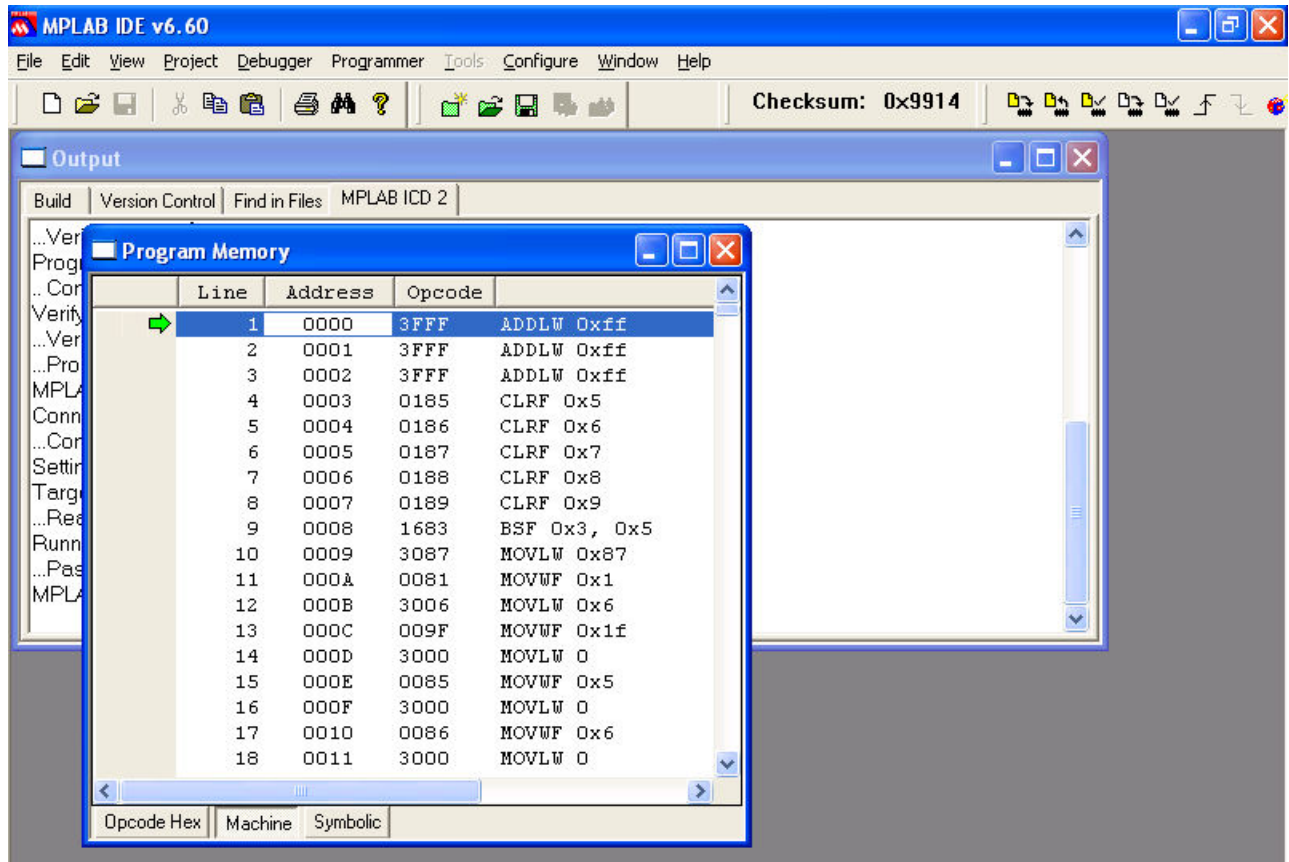


Olvassuk be az égetendő .HEX fájlt! A „File” menü „Import” menüjére kattintva, tudunk a fájlok között válogatni. Keressük meg a PICOK04.HEX-et :

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

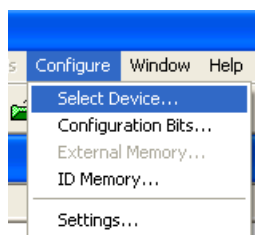


Ha beolvastuk be a file-t, akkor megnyithatunk egy ablakot, hogy lássuk mit is olvasunk/égetünk be! Menjünk a "View" menübe, azon belül a "Program Memory"-ra kattintsunk:

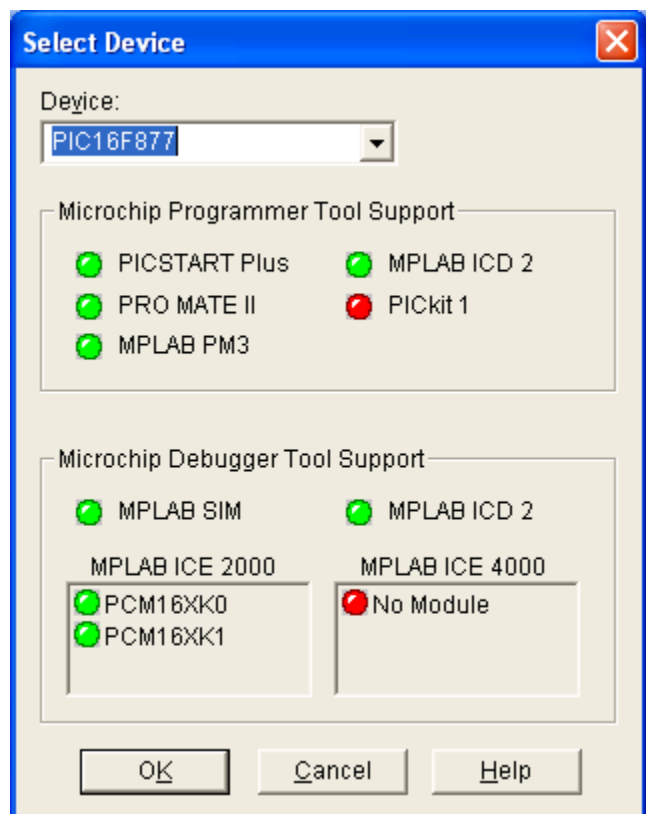


Ahogy fent látjuk, megnyitott ablakban megjelennek az égetendő adatok, sőt, az MPLAB mindjárt vissza is fordítja az adatokból az assembly utasításokat, és azokat is megmutatja.

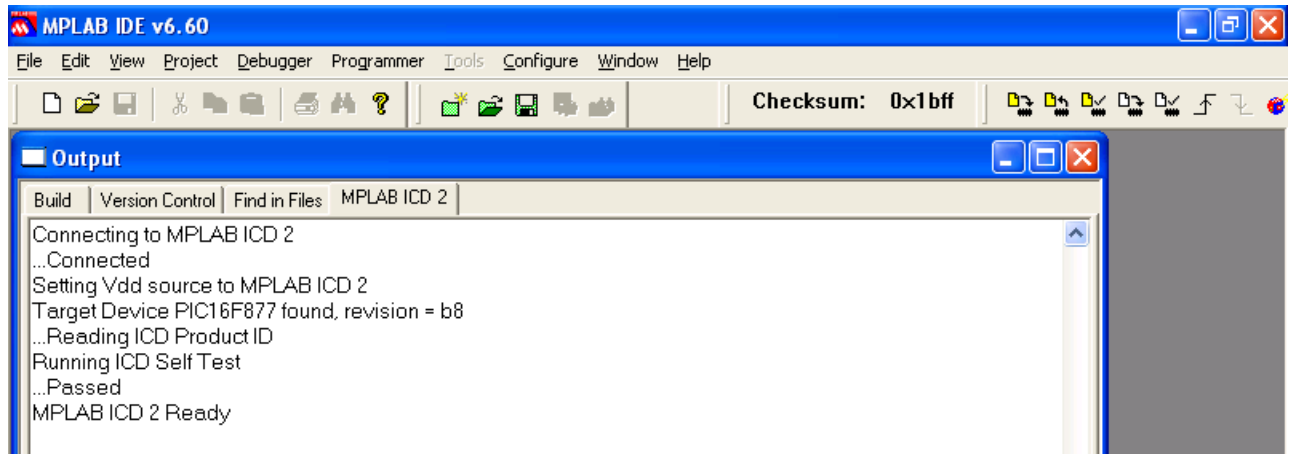
Válasszuk ki a PIC típust, amit égetni akarunk! Kattintsunk a „Configure” menü „Select Device” ablakra:



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



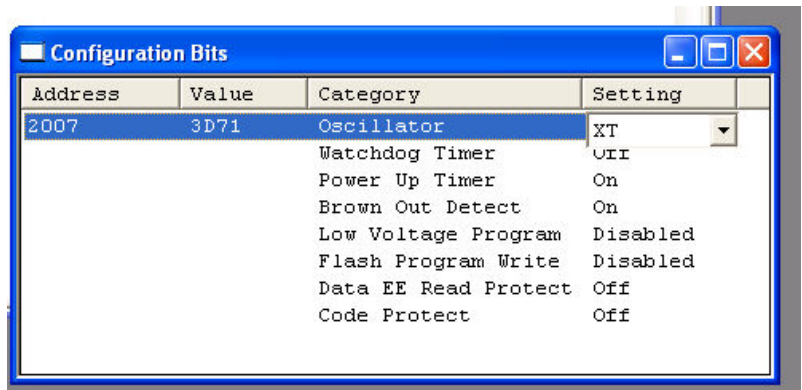
Az ICD rögtön le is ellenőrzi, hogy tényleg a beállított IC lóg a kábel végén:



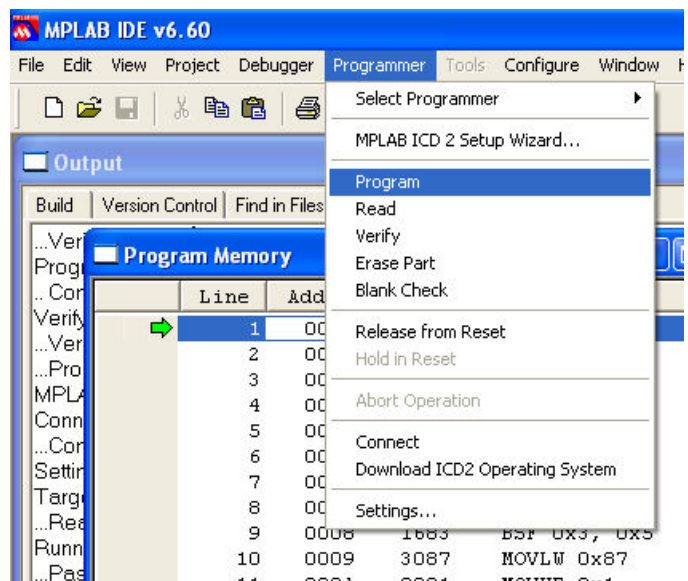
Ahogy fent látható, az ICD 16F877-et talált, ez rendben is van.

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

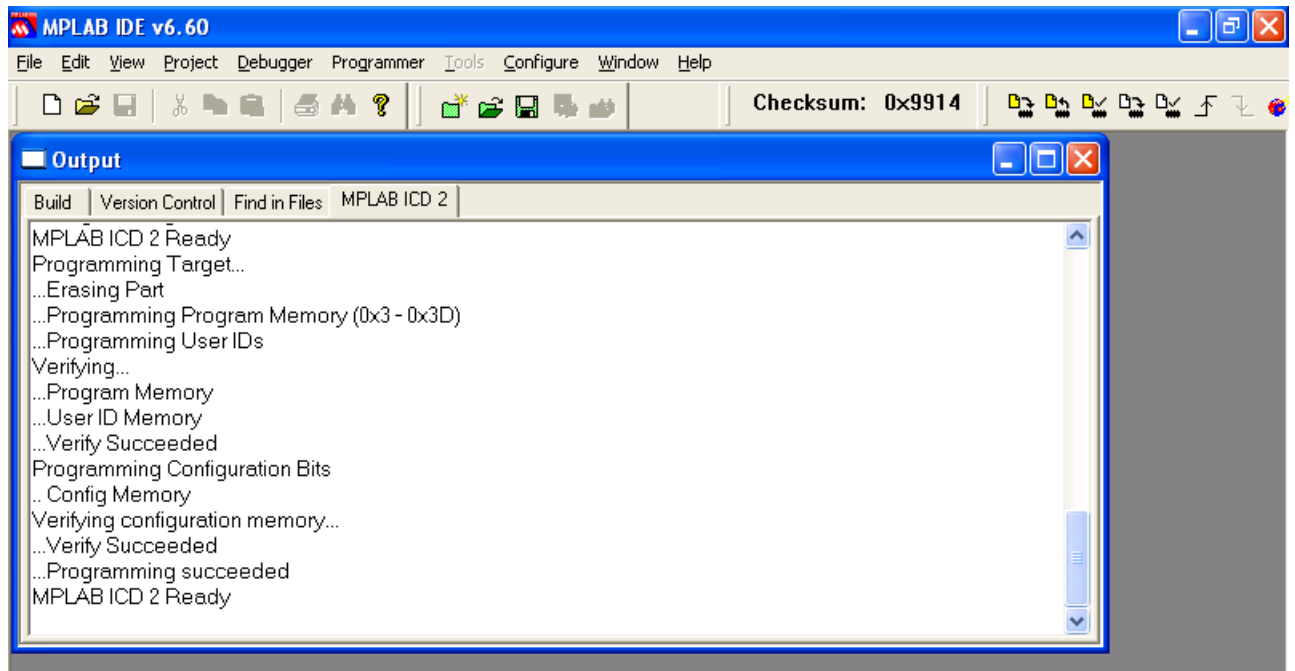
Szóval végezzük el mikrokontroller kiválasztást, és a "fuse" beállításokat. Ha titkosítani akarjuk a tartalmat, akkor válasszuk a "code protect" On opciót. (Itt több lehetőség közül választhatunk, a teljes programmemória, vagy annak csak egy részének a titkosítása is kijelölhető.)



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 ezt láttuk, akkor minden rendben. Egy kattintás a „Release from Reset-re,



és már fut is a program. (Ha a PICDEMO panel volt a cél áramkör, akkor szó szerint fut a program, ugyanis indul a futófény az RB portra kötött LEDeken...)

Készült egy programozó adapter is, amivel az ICD2 kiegészítve már valóban egy komplett PIC programozó. Az adapter a DIP tokos 8, 18, 28 és 40 lábú mikrokontrollerekhez használható. Ezek a széles vágatú programozó foglalatba csíptetendők. Az áramköréről egy külön leírás szól, az ADAPICD.PDF .

Kapcsolódó dokumentációk:

Létezik egy egyszerűbb, soros portról üzemelő verzióró is. (ICD2.PDF)

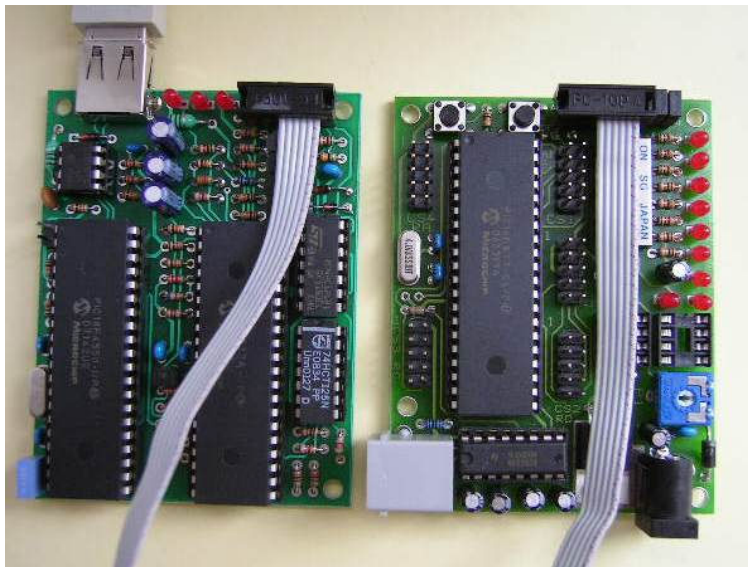
Az ICD teljes értékű PIC programozó lesz, a karos IC foglalatú programozó adapter csatlakoztatásával. (ADAPICD.PDF)

Egy konkrét példán keresztül, azaz az MPLAB-ban egy MICROCHIP-es példa program lefordításával, egy "project" létrehozásával, annak letöltésének folyamatával foglalkozik az MPLAB6.PDF dokumentáció.

Szintén a témához kapcsolódik a PICDEMO panel dokumentációja. (PICDEMO.PDF)

Valamint PIC16F87x mikrokontrollerrel működik a vezérlési/automatizálási feladatokra készült PICPLC16, és a PICPLC8 áramkör. Ezekről egy-egy külön leírás szól. (PICPLC8.PDF, és PICPLC16.PDF)

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



Epilógus:

A korábban leírt szépséghibák miatt, készül egy átdolgozott a verzió, ami e prototípus gyerekbetegségeit orvosolja. Azaz a tápellátást egy dugasztáp biztosítja, és javítva lesz a VCC kijelzés.

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>