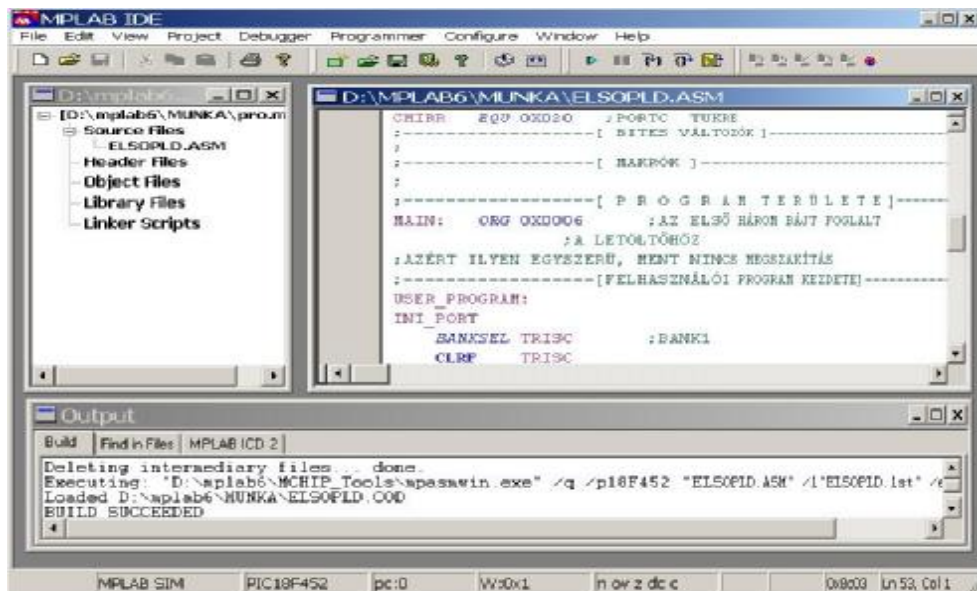


A következő oldalon lévő ábrán az MPLAB egy menüképe látható.



A most következő összefoglalás nem helyettesíti a kézikönyvet, és nem törekszik a teljességre. A menükkel való ismerkedésnek a legjobb módja azok kipróbálása. Feltételezzük továbbá, hogy az olvasó jártas a Windows alapú környezet használatában.

Az ilyen, „mindent az egyben” típusú programot szokták **integrált fejlesztőkörnyezetnek** (Integrated Development Environment – **IDE**) is nevezni.

Az MPLAB parancsainak kiadására három lehetőségünk van:

- a menürendszerből választjuk ki,
- a hozzájuk rendelt billentyűkombinációt használjuk,
- az eszköztár megfelelő ikonjára kattintva aktivizáljuk.

Az MPLAB-ban végzett munkával kapcsolatban először két fontos fogalommal kell megismerkednünk: a **projekttel**, illetve a **munkakörnyezettel (workspace)**.

<b>New</b>	<b>Ctrl+N</b>	Új forrásfájl készítése.
<b>Open...</b>	<b>Ctrl+O</b>	Meglévő forrásfájl megnyitása.
<b>Close</b>		Bezárás
<b>Save</b>	<b>Ctrl+S</b>	Mentés
<b>Save As...</b>		Mentés más néven
<b>Save All</b>		Az összes megnyitott fájl mentése
<b>Open Workspace...</b>		Munkakörnyezet megnyitása
<b>Save Workspace</b>		Munkakörnyezet mentése
<b>Save Workspace As...</b>		Munkakörnyezet mentése más néven
<b>Close Workspace</b>		Munkakörnyezet bezárása
<b>Import...</b>		HEX kiterjesztésű fájl tudunk behívni
<b>Export...</b>		HEX fájl mentése
<b>Print...</b>	<b>Ctrl+P</b>	Kiválasztott fájl nyomtatása
<b>Recent Files</b>		
<b>Recent Workspaces</b>		Az utoljára használt fájlok, illetve munkakörnyezet leíró fájl listája van a menüpont végén.
<b>Exit</b>		Kilépés az MPLAB-ból

A **projekt** koncepció azt jelenti, hogy az adott fejlesztéshez tartozó fájlokat egy ún. projektfájlban tartjuk nyilván. Ha ezt a fájlt megnyitjuk, a nyilvántartott fájlok a fejlesztéskor azonnal betöltődnek és használhatók. A **Project** menüvel lehet a projekteket kezelni.

A **munkakörnyezet** a kiválasztott mikrokontrollerről, a debug, illetve programozó eszközökről, a megnyitott ablakokról, és azok helyéről és egyéb IDE beállításokról hordoz információt.

Installálás után legördülő menüket tartalmazó menürendszer jelenik meg, amelynek első eleme a **File** menüpont:

### AZ MPLAB SZÖVEGSZERKESZTŐJE (EDIT MENÜ)

A szövegszerkesztés karakterekkel (és az ezekből képzett, bonyolultabb szövegobjektumokkal) végzett műveletek összessége. A leggyakoribb műveletek:

**Kijelölés (selection).** Kijelöljük azt a szövegrészt, amely a következő műveletben majd részt vesz. A kijelölt objektum lehet karakter, sor vagy logikailag osztályozva szó, mondat, bekezdés. A kijelölés alatt egy szövegobjektum kiválasztását értjük.

**Beillesztés (insert).** Egy szövegobjektum beillesztése egy megadott helyre (pl. maga a begépelés).

**Törlés (delete).** A már előzőleg kijelölt objektumokkal végezhetjük el. Törlés esetén az objektum eltűnik a dokumentumból.

**Másolás (copy).** A kijelölt rész a másolás hatására egy másik (általunk kiválasztott) helyen ismételtelen megjelenik.

**Mozgatás (move).** A kijelölt rész az adott helyről átkerül az általunk kiválasztott új helyre.

Az **Edit** menü az MPLAB szövegszerkesztőjéhez kapcsolódik. Az MPLAB szövegszerkesztője lényegében egy átlagos, Windows alapú szövegszerkesztő, de olyan részekkel lett kiegészítve, amelyek a programok megrrását megkönnyítik.

<b>Undo</b>	<b>Ctrl+Z</b>	Utolsó szerkesztő művelet visszavonása.
<b>Redo</b>		Utolsó szerkesztő művelet megismétlése
<b>Cut</b>	<b>Ctrl+X</b>	Kijelölt rész kivágása és a vágólapra másolása.
<b>Copy</b>	<b>Ctrl+C</b>	Kijelölt rész vágólapra másolása.
<b>Paste</b>	<b>Ctrl+V</b>	A kurzorral megjelölt helyre másoljuk a vágólap tartalmát.
<b>Delete</b>	<b>Del</b>	Kijelölt rész törlése.
<b>Select All</b>	<b>Ctrl+A</b>	A teljes szöveg kijelölése.
<b>Find...</b>	<b>Ctrl+F</b>	Keresés.
<b>Find Next.</b>	<b>F3</b>	Keresés a következőre.
<b>Replace...</b>	<b>Ctrl+H</b>	Keresés és helyettesítés.
<b>Go To...</b>	<b>Ctrl+G</b>	Ugrás adott sorszámú sorra.
<b>Match Brace</b>	<b>Ctrl+M</b>	Ha a kezdő zárójelre állunk a kurzorral, akkor a parancs segítségével megkereshetjük a hozzá tartozó befejező zárójelre.
<b>Properties...</b>		A szöveg jellemzőit: nagyságot, szint, karakter típust, tabulátor hosszát stb. állíthatunk be itt.

Fontos megjegyezni, hogy a program egyes jellemzőinek megjelenítése színes szöveggel történik, így jól megkülönböztethetők az utasítások, azok paraméterei, megjegyzések, illetve az assembler direktívák.

### VIEW MENÜ

- ✓ **Project**
- Output**
- Hardware Stack**
- Program Memory**
- File Registers**
- EEPROM**
- Watch**
- Special Function Reg**

**WATCH** ablak

Ezek közül külön megemlíthetjük a Watch (néző) ablakot. A program regisztereinek futás közbeni megfigyelésére az Add SFR vagy az Add Symbol gombok segítségével lehet a mikrovezérlő SFR, illetve felhasználói regisztereit kiválasztani,

<b>Close</b>	
<b>Find ...</b>	<b>Ctrl+F</b>
<b>Find Next</b>	<b>F3</b>
<b>Delete</b>	
<b>Save Watch ...</b>	
<b>Load Watch ...</b>	
<b>Rename Watch ...</b>	
<b>Output To File ...</b>	
<b>Print ...</b>	<b>Ctrl+P</b>
<b>Refresh</b>	
<b>Properties ...</b>	

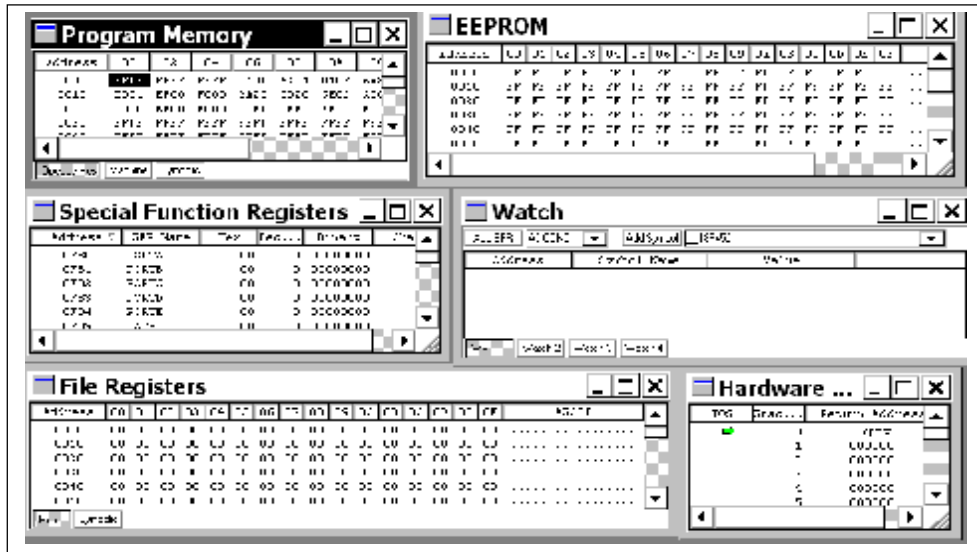
és tartalmukat megjeleníteni, illetve módosítani. A watch ablakra jobb egérgombbal kattintva egy menü jelenik meg, aminek segítségével a watch ablakok menthetők, átnevezhetők, törölhetők, illetve betölthetők. Az ablakban lévő regiszterek tartalma fájlba menthető, illetve kinyomtatható. Egy projekthez egyidőben maximum négy watch ablak is választható.

A Properties... menüben állíthatjuk be, hogy a regiszter értékét milyen formában (decimális, hexa, bináris, ASCII) kívánjuk megjeleníteni.

**FONTOS!**

**A WATCH ablakban elhelyezett regiszterek tartalmának vizsgálatával tudjuk a programunk helyes működését ellenőrizni.**

A következő ábrán összefoglaltuk a megjeleníthető adatsoportok ablakait.

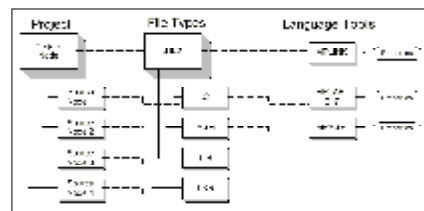


**PROJECT MENÜ**

Két fogalmat, a projekt, illetve munkakörnyezet fogalmát már ismertettük. A leggyakoribb esetekben egy munkakörnyezethez egy projekt tartozik, de több projekt is használható egy munkakörnyezetben. Ezek alapján lehetséges:

- egy assembly fájl használata (Quickbuild),
- egy projekt használata egy munkakörnyezetben – ez a MPLAB IDE v5.xx programnál használt kialakítás,
- több projekt használata egy munkakörnyezetben.

Fontos tudnunk, hogy az MPLAB 6.xx verzióban először mindig az eszközt, vagyis a mikrokontrollert választjuk ki a **Configure>Select Device** menüpontban.



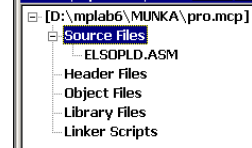
Az MPLAB-ben a projekt egy projektobjektum (ez a node, azaz csomópont) egy vagy több forrásobjektumból (node-ból) épül fel. Ezek a források általában MPASM assembler, illetve magas szintű C vagy BASIC forrásfájlok, előre lefordított könyvtárfájlok, más tárgykódok (object fájlok), illetve az ezek összekapcsolását végző linker script vezérlő szövegfájlok (hasonlók a DOS batch fájllaihoz).

Általában ugyanabban a könyvtárban helyezük el a projektet, mint a forrásfájlokat. Linkeléskor az összelinkelni kívánt ún. object fájlokból hozzuk létre a végleges lefordított tárgykódot, vagyis .HEX fájl ilyen node-ok építik fel. Egy node

New...	
Open...	
Close	
<b>Set Active Project</b>	
Quickbuild (no .asm file)	Alt+F10
Clean	
Build All	Ctrl+F10
Make	F10
Build Options...	
Find in Project Files...	
Save Project	
Add Files to Project...	
Remove File From Project	
Select Language Toolsuite...	
Set Language Tool Locations...	

egy forrásprogramot és egy fordítóprogramot (BASIC, C) vagy assembler programot is meghatároz, ezeket hívjuk nyelvi eszközöknek (**Language Tool** – nyelvi eszköz).

A **Project** menü **New, Open, Close** pontjainak jelentése nyilvánvaló: projekt létrehozása, meglévő megnyitása, illetve az aktív projekt bezárása.



**Set Active Project:** több megnyitott projekt esetén kiválaszthatjuk az éppen használtat.

**QUICKBUILD használata**

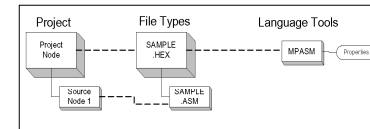
A legtöbb esetben a fejlesztés egyetlen assembler programot tartalmaz. Ilyenkor a projekt létrehozása helyett a gyorsabb megoldást biztosító **Quickbuild** fejlesztő módot használjuk.

Először kiválasztjuk a fejlesztéshez használt kontrollert: **Configure>Select Device**. Ellenőrizzük, hogy a Quickbuild mód aktív: (**Project>Set Active Project>None** (Quickbuild mode)). Vagy létrehozunk az assembler fájlt (**File>New**), vagy egy létezőt töltünk be (**File>Open**). Assembláláshoz kiválasztjuk a **Project>Quickbuild file.asm** menüpontot, ahol file.asm az aktív fájl neve.

Ezek után már lehet a programban hibát keresni, vagy a szimulátorban futtatni. A munka befejezésekor a munkakörnyezet megadott néven elmenthető: **File>Save Workspace**, illetve újbóli használatkor betölthető: **File>Open Workspace**.

**Egy projekt egy munkakörnyezetben**

Mi a továbbiakban (mivel ez a dőknek szól), csupán az MPLAB IDE projekt létrehozásának lépéseit írjuk leggyakrabban előforduló eset. Ilyenkor nem szükséges. Ezt az üzemmódot fülön állíthatjuk be: **Use one-to-one**. Először kiválasztjuk a kontrollert: **Configure>Select Device**.



leírás az MPLAB-bel ismerke- v5.xx programnál megszokott le. A gyakorlatban is ez a munkakörnyezet beállítása a **Configure>Settings>Projects project-workspace model**. Utána: fejlesztéshez használt

Létrehozunk a projektet, és beillesztjük a fájlokat: **Project>Add Files to Project...**

Létrehozunk a tárgykódot tartalmazó fájlt: **Project>Build All**. Ha csak a megváltozott fájlokat akarjuk újra fordítani, akkor a **Project>Make** menüpontot aktivizáljuk.

Foglaljuk össze a további menüpontok jelentését:

- Project>Clean** menüpontban a projekt létrehozása során keletkezett fájlokat törölhetjük ki.
- Project>Build All** menüpontot aktualizálva a projekt újrafordítása történik meg. (Ilyenkor a szimbólumokat tartalmazó .COD kiterjesztésű fájl létrehozási időpontját hasonlítja a projekt .INC és .ASM kiterjesztésű fájljaihoz. Ha a .COD régebbi, mint a hasonlított fájlok valamelyike [valószínűleg módosítva lett], akkor elvégzi az újrafordítást.)
- Project>Build Options...** menüpontra kattintva megadhatjuk a projektben szereplő fájlok útvonalát, azaz a könyvtárakat.
- Project>Save Project** a projekt mentése.
- Project>Add Files to Project...** a menüpont segítségével tudjuk a projektbe a szükséges (pl. :ASM) fájlokat beilleszteni.
- Project>Remove File From Project** a menüpont segítségével tudjuk a projektből fájlokat eltávolítani.
- Project>Select Language Toolsuits**, illetve a **Project>Set Language Tool Locations...** menüpontok segítségével új fordítót (pl. C, BASIC) illeszthetünk az MPLAB környezetébe.

<input checked="" type="checkbox"/>	Close open source files on project close
<input checked="" type="checkbox"/>	Clear output window before build
<input checked="" type="checkbox"/>	Save project before build
<input type="checkbox"/>	Save files before build
<input type="radio"/>	Yes
<input type="radio"/>	No
<input type="radio"/>	Prompt
<input checked="" type="checkbox"/>	Halt build on first failure
<input checked="" type="checkbox"/>	Use one-to-one project-workspace model

**AZ MPLAB SZIMULÁTORA**

A **Debugger>Select Tool>MPLAB\_SIM** kiválasztásakor aktivizált szimulátorprogram arra szolgál, hogy a PIC mikrokontrollerekre írt programot ellenőrzött módon lehessen végrehajtani. Ez azt jelenti, hogy a program végrehajtása közben:

- meg tudjuk vizsgálni és esetleg módosítani a kontrollert bármelyik regiszterének tartalmát (watch),
- adott helyen meg tudjuk állítani a program futását, azaz töréspontot tudunk elhelyezni (breakpoint),
- tudjuk a programot lépésenként futtatni (single step),
- a külső környezetből jövő (input) adatokat fájlból beolvasva szimulálni (stimulus),
- illetve kiküldött (output) adatokat fájlba eltárolni,
- a szimuláció során bekövetkező eseményeket a későbbi elemzés céljából gyűjteni (trace).

A szimulátor csupán egy a hibakeresést és tesztelést lehetővé tévő eszközök közül. A **Select Tool** aktivizálásával választhatjuk még az ICD2 debugert, illetve több emulátor típust.

A **Debugger>Clear All Memory** menüpont segítségével a szimulátor minden memóriáját töröljük (0-val töltjük

<b>Select Tool</b>	
<b>Clear All Memory</b>	
<b>Run</b>	F9
<b>Halt</b>	F5
<b>Step Into</b>	F7
<b>Step Over</b>	F8
<b>Reset</b>	
<b>Breakpoints...</b>	F2
<b>Stopwatch</b>	
<b>Stimulus</b>	
<b>Refresh PM</b>	
<b>Settings...</b>	

fel). A további menüpontok:

<b>Run</b>	<b>F9</b>	A program futtatása.
<b>Halt</b>	<b>F5</b>	A futó program megállítása, programszámláló megáll.
<b>Step Into</b>	<b>F7</b>	Lépésenkénti programvégrehajtás.
<b>Step Over</b>	<b>F8</b>	Lépésenkénti futtatás, a rutinokat egy lépésben hajtja végre.
<b>Reset</b>		Alaphelyzetbe állítás, programszámlálóba nulla kerül.

*Debugger>Breakpoints...* A felhasználónak a program belövése során igen nagy segítséget nyújt, ha a programot egy, még az indítás előtt kijelölt címeken meg lehet állítani, azaz a futását felfüggeszteni. Ekkor ugyanis a regiszterek és a programban használt változók tartalmainak vizsgálatával könnyen eldönthető, hogy a program eddig a pontig helyesen vagy hibásan működött. A teljes programot több, logikailag jól elhatárolt részre bontva, a hibásan működő programrész gyorsan meghatározható. Ilyen leállási feltétel – azaz **töréspont (breakpoint)** – a programban több is elhelyezhető.

Az is fontos lehet, hogy egy program szimulálása közben a futás során a használt regiszterekről, azok tartalmáról adatokat gyűjtsünk. Ez a nyomkövetés vagy más néven **tracing**.

Simulátor használatakor alkalmazhatunk a programunknál nyomkövetést. A **Trace** ablak *Debugger>Settings>Trace/Pins* menüjében engedélyezhetjük a nyomkövetést. A nyomkövetés eredménye a **Trace** ablakban látható, ahol az oszlopok feliratai a következőket jelentik:

- Line** - sorszám a nyomkövetés kezdetétől.
- Addr** - az utasítás programcíme
- Op** - az utasítás numerikus kódja
- Label** - az utasítás címkéje
- Instruction** - az utasítás visszafordított mnemonikja
- SA** - forráscím
- SD** - forrásadat
- DA** - célcím
- DD** - céladat
- Cycles** - az eltelt ciklusok száma.

**A lépésenkénti programvégrehajtás (Step Into, Step Over)** üzemmódban a felhasználói programnak mindig csak egy utasítása hajtódik végre, és ilyenkor mindig lehetőség van a regiszterek vagy a tártartalmak vizsgálatára, vagyis a végrehajtott utasítás hatásának, a program helyes működésének az ellenőrzésére.

A stopperórával (**Stowatch**) programrészletek időbeli végrehajtását követhetjük nyomon. Megadhatjuk a szimulált processzor órajelét (**Processor Frequency**). A **Zero** gombbal nullázhatjuk a stopper. Kijelzi a nullázás óta végrehajtott ciklusok számát (**Instruction Cycles** – a legtöbb esetben egy utasítás egy ciklus), illetve az eltelt időt (**Time**).

Választható, hogy a stopper RESET-kor törlődjön.

**A külvilág figyelembe vétele: a gerjesztések (stimulusok)**

A szimuláció során a legnehezebb feladat a külső környezet figyelembe vétele. A külvilággal való kapcsolat szimulációjának megvalósítására az MPLAB két lehetőséget is felkínál:

- a processzor lábaira adhatunk jelet (**Pin stimulus**),
- illetve a program futásakor a végrehajtott ciklusszám, illetve az utasításszámláló (PC) adott értékének elérésekor generált **trigger** feltétel teljesülése esetén egy adott regiszterbe megadott érték kerül (**File stimulus**).

**A lábakra adható gerjesztések: Pin stimulus**

Az **Add Row** jelű gombra kattintva adhatunk meg új lábgerjesztést. A **Load** és **Save** gombokkal menthetjük el, illetve tölthetjük be a kész pin stimulus állományt.

A külvilágot egy lábon kétféle módon szimulálhatjuk: **szinkron (clock) stimulus** alkalmazásakor egy I/O lábhoz (**Pin**) ismétlődő periodikus négyszögjel (**clock**) rendelünk: meg kell adni, hogy hány ciklusig legyen a láb állapota magas (**high cycles**), és hány ciklusig alacsony (**low cycles**) szintű. A ciklusszámot decimális számként kell megadni. Az **Invert** bejelölésével a jel ellentétes lesz (alacsonnyal kezd).

**Aszinkron stimulus** segítségével a szimulált controller lábaihoz gerjesztéseket rendelünk, és ezeket a képernyőn lévő nyomógomb (**Fire**) segítségével aktivizálhatjuk. A stimulushoz hozzá kell rendelnünk a lábat (**Pin**) és a gerjesztés típusát (**Action**):

- Low** – a gomb megnyomásakor alacsony szint lesz az adott lábon;
- High** – a gomb megnyomásakor magas szint lesz az adott lábon;
- Toggle** – a gomb megnyomásakor az adott lábon lévő szint ellentétére vált;
- Pulse** – a gomb megnyomásakor egy impulzus jelenik meg az adott lábon.

**Regisztertartalmak megadása futáskor: File stimulus**

**Fájl stimulus** esetén egyedi gerjesztéseket tartalmazó **szinkron stimulus** fájlok listáját használjuk, vagyis egyszerre több gerjesztés lehet aktív. Az **Input Files** területen lévő **Add**, **Remove**, **Edit**, **Save**, **Cancel** gombokkal kezelhetjük ezeket a fájlokat. A listában szereplő szinkron stimulus fájl megszerkesztése az **Edit** gombra kattintással lehetséges.

A gerjesztés aktivizálási feltételét a **Trigger On** oszlopban lehet megadni, ez lehet **Cycles** – vagyis a RESET utáni hányadik utasítás ciklus után aktivizálódjon a gerjesztés, vagy **PC** azaz az utasításszámláló egy adott értéke.

**Trig Value** oszlopba kell írni a ciklusszámot, vagy a PC értékét.

**Repeat** jelölése esetén a megadott ciklusszám határozza meg a ciklusos gerjesztés ismétlődését.

**Pin/Register** oszlopban egy legördülő listából választhatjuk ki a gerjesztést kapó lábat, illetve azt a regisztert amelybe értéket egy **register stimulus fájl**ból jutattunk.

**Value:** a beküldött érték: ez 0 vagy 1 láb (Pin) gerjesztés esetén. Register gerjesztés esetén vagy egy hexadecimális érték, illetve egy register stimulus fájl megadása.

**Register Stimulus:** amikor a szimulációban megadott címre érünk, megadott regiszterbe egy szövegfájlból adott értéket töltünk. Ha többször haladunk át az adott címen, akkor a fájlban szereplő egymás utáni értékek sorban egymás után betöltődnek a regiszterbe. Csak **Trigger On=PC** esetén használható!

**PICSTART PLUS PROGRAMOZÓ**

A Picstart Plus programozó segítségével tudjuk a fejlesztett programjainkat EPROM-os vagy OTP-s tokba írni, és a valódi hardverkörnyezetben kipróbálni. A programozó minden PIC controller írására képes, az újabb típusok megjelenésekor az MCHIP adja azt a .HEX



kiterjesztésű fájlt, amelyet egy PIC17C44-es tokba írva a programozó aktualizálható (upgrade).

A programozót a számítógép valamelyik soros vonalára kell csatlakoztatni. A programozás során az MPLAB belső memóriapufferének tartalmát írjuk a programozóban lévő tokba, a beállított konfigurációs bitekkel együtt.

A puffert vagy egy fájlból töltjük fel: (File>Import) vagy az assemblálás során keletkezik. A programozóval pufferbe beolvasott PIC tok tartalmát fájlba is menthetjük: (File>Export)

**Enable Programmer:** a programozó engedélyezése.

**Disable Programmer:** a programozó tiltása.

**Program:** programozás.

**Verify:** a beírt program ellenőrzése.

**Read:** a programozóba helyezett PIC tok olvasása.

**Blank Check All:** a tok ürességének (memória + konfigurációs bitek) vizsgálata.

**Blank Check OTP** : OTP tok ürességének vizsgálata.

**Reset:** a programozó alaphelyzetbe állítása.

**Reset Program Statistics:** a programozáskor gyűjtött adatok törlése.

**Erase Flash Device:** a flash eszköz törlése.

**About:** a programozóról...

**Settings:** programozandó memóriatartományok, kommunikáció.

## CONFIGURE MENÜ

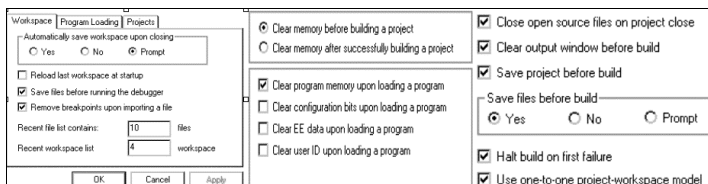
A munka kezdetekor ezt a menüpontot kell aktivizálni, és a *Select Device...* menüpontban kell a használt mikrovezérlőt kiválasztani.

A *Configuration Bits...* menüpontban lehet a konfigurációs biteket beállítani, ha a programban a CONFIG direktíva segítségével ezt nem tettük meg.

*External Memory* külső memóriát tartalmazó eszközök esetén állíthatók be a paraméterek.

*ID Memory...* a tokba írható egyedi azonosító ID értéke adható meg.

*Settings...* Munkaterület, program betöltési és projekt paraméterek adhatók meg itt.



## AZ ICD2 KEZELÉSE

Ha az ICD2-t használjuk, akkor a *Debugger>Tools* menüpontban az MPLAB ICD2 menüpontot kell választani. Természetesen először a debuggert soros vagy USB porton kell csatlakoztatni, majd soros kommunikáció esetén a megfelelő COM portot és adatátviteli sebességet (pl. 19200) is be kell állítani. Az USB használata a port installálását igényli. A *Debugger>Settings...* menüpontban az ICD2 paramétereit állíthatjuk be.

## WINDOW MENÜ

A WINDOW menüpontok Windows környezetben megszokottak.

## HELP MENÜ

Az egyik legnagyobb erőnye a MPLAB-nak az igen részletes és bőséges (igaz, angol nyelvű) segítség. Az itt található leírások hipertext jellegűek, azaz linkek vannak elhelyezve, amire kattintva eljuthatunk egy másik információhoz.

Lényegében minden információt tartalmaz a programról, külön szerepeltetve az MPLAB egyes komponenseit. Az egyes sűgök a jobb eligazodást segítő tematikus módon alfejezetekre vannak osztva.

## ÖSSZEFOGLALÁS

Foglaljuk össze, hogyan kell használnunk az MPLAB-ot programjaink fejlesztésére. Első lépésként kiválasztjuk a megfelelő PIC-típust. (*Configure > Select device*), ha még nincs projektünk, a **File > New** menüpontjával

létrehozhatunk egy új forrásfájlt. Másik, és valószínűleg jobb megoldás az, hogy a létrehozott új projektünkbe egy átnevezett .ASM fájlt illesztünk, mert azt könnyen módosíthatjuk a közös, szabványos részek meghagyásával. Ha szükséges, a projektbe INCLUDE fájlokat is beilleszthetünk.

Elmentés után a fájlokat a **Project > Quickbuild** menüpont segítségével lefordítjuk, az esetleges hibákat kijavítjuk. Ehhez az is segítséget nyújt, hogy a hibafájl megjelenítésekor rákattintva egérgombbal a kurzorral mutatott hibás sorra, a forrásprogram megfelelő sora jelenik meg a szerkesztőben.

A fájl futtatása a megfelelő debugger menüpont segítségével végezhető. Beállíthatók töréspontok, nyomkövetés, és gerjesztések (stimulusok) segítségével a külső környezetet is szimulálhatjuk. A **Watch** ablak felhasználásával a futás során fontos regisztereket is figyelemmel kísérhetjük.

**Fontos! A megfelelő működéshez a processzor típusát a forrásszövegben megadott típushoz azonos módon kötelező megadni!**

Egy ilyen összetett eszköz, mint az MPLAB kezelése, használatának elsajátítása, szorgalmas munkát igényel. A munkát jelentősen megkönnyíti az eszköztárak és a státuszor felhasználása. A leírás nem tért ki az egér használatára, de célszerű *mindkét* egérgombot különböző szituációkban kipróbálni, mert számos funkció valósítható meg az egér segítségével. (Például töréspontok, nyomkövetés beállítása a forrásnyelvi szövegben kurzorral az adott részre mutató és a jobb egérgombbal kattintva.)

## SZERSZÁMKÉSZLETEK (TOOLBARS)



A fejlesztői környezet használatát nagyon megkönnyíti a felső sorban elhelyezett, egérkattintással aktivizálható „szerszámok polc” (toolbar). Bár az egérkurzort az egyes ikonokra pozicionálva megjelenik az ikonhoz tartozó funkció neve, de a következőkben a könnyebb használat érdekében ezeket összefoglaljuk. Négy csoport van:

### Édit Toolbar Project Toolbar Debug Toolbar Program Toolbar

#### Édit Toolbar ikonjai:

New File – Új fájl ablak megnyitása

Open File – Egy létező fájl megnyitása az ablakban

Save File – Az aktuális fájl mentése

Cut – A kiválasztott szöveg kivágása és áthelyezése a vágólapra

Copy - A kiválasztott szöveg átmásolása a vágólapra

Paste - Szöveg bemásolása az aktuális kurzorpozícióra a vágólapról

Print – Az aktív ablakban lévő fájl nyomtatása

About – Információ az MPLAB IDE fejlesztői környezetről

#### Project Toolbar ikonjai:

New Project – Egy új projekt helyének és nevének megadása

Open Project – Létező projekt megnyitása

Save Workspace – Az aktuális projekt, és munkakörnyezetének mentése

Build Options – A projekt beállításainak megnézése, vagy megváltoztatása

Project Help – Projekttekkel kapcsolatos segítség

Make – Csak azokat a fájlokat dolgozza fel, amelyek megváltoztak

Build All – Minden fájl feldolgozása

#### Debug Toolbar ikonjai:

Run – Program futtatása

Halt – Program végrehajtásának megállítása

Step Into – Következő utasítás végrehajtása

Step Over – Szubrutin végrehajtása egy lépésben

Reset - MCLR reset végrehajtása

#### Program Toolbar ikonjai:

Program – Kódmemória programozása a *Programmer>Settings* menüben beállítottak szerint.

Read - Kódmemória kiolvasása a *Programmer>Settings* menüben beállítottak szerint.

Verify – Programozás helyességének az ellenőrzése a *Programmer>Settings*, menüben beállítottak szerint.

#### Megjegyzések:

A leírás az MPLAB V6.10 verzió alapján készült.

Egyes menüpontok kiválasztásakor újabb ikonok is megjelenhetnek.

