

Τομέας: Ηλεκτρολογίας – Ηλεκτρονικής

Εκπαιδευτικός: Μπουλταδάκης Στέλιος
Κασάμπαλης Στέλιος

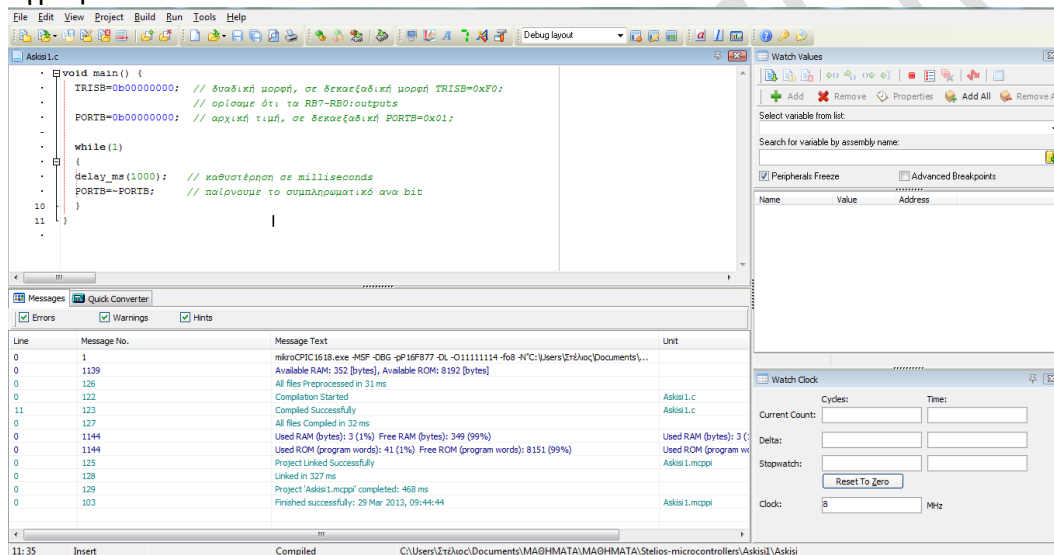
Μάθημα: Μηχαντρονική, Αισθητήρες-Ενεργοποιητές

Αντικείμενο: Οδηγίες προγραμματισμού μικροελεγκτών

Ολοκληρωμένο Προγραμματιστικό Περιβάλλον: εννοούμε τη γλώσσα προγραμματισμού που χρησιμοποιούμε για να γράψουμε τον πηγαίο κώδικα. Πλέον είναι γλώσσα ανώτερου επιπέδου που βασίζεται στη C++ όπως:

- Περιβάλλον MPLAB X IDE : <http://microchip.wikidot.com/tls0101:start>
- Περιβάλλον Mikro C : <http://www.mikroe.com/mikroc/pic/>
- Περιβάλλον της Wiring C πλατφόρμες Arduino <https://www.arduino.cc/> που χρησιμοποιούν μικροελεγκτές της ATMEL

Ο πηγαίος κώδικας είναι συνήθως ένα αρχείο με κατάληξη .c και γράφεται με το 'συντάκτη' στη γλώσσα ανωτέρου επιπέδου όπως φαίνεται στην παρακάτω φωτογραφία.

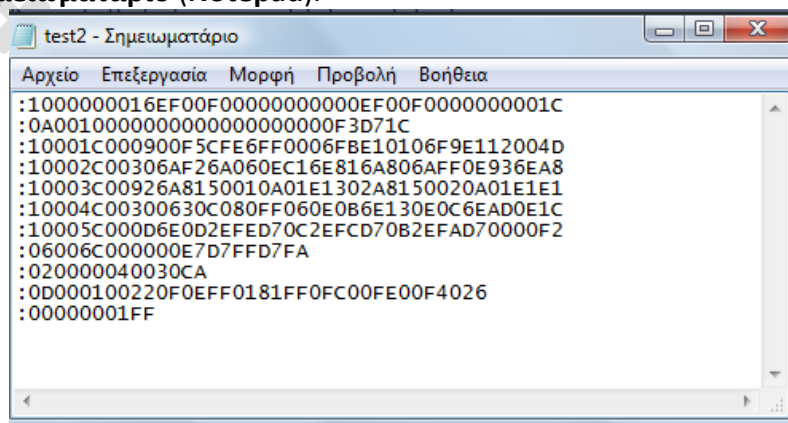


```
void main() {
    TRISB=0b00000000; // δεκαεξαδική μορφή, σε δεκαεξαδική μορφή TRISB=0xF0;
    // ορίσαμε ότι τα RB7-RB0 είναι inputs
    PORTB=0b00000000; // αρχική τιμή, σε δεκαεξαδική PORTB=0x01;

    while(1)
    {
        delay_ms(1000); // καθυστέρηση σε milliseconds
        PORTB=~PORTB; // παίρνουμε το συμπληρωματικό ανα bit
    }
}
```

Line	Message No.	Message Text	Unit
0	1	microC1618.exe -MSF -DBG -p16F877 -DL -O11111114 -f68 -Y\C:\Users\Στέλιος\Documents\...	
0	1139	Available RAM: 352 [bytes], Available ROM: 8192 [bytes]	
0	126	All Files Preprocessed in 31 ms	
0	122	Compilation Started	Askis1.c
11	123	Completed Successfully	Askis1.c
0	127	All Files Compiled in 32 ms	
0	1144	Used RAM (bytes): 3 (1%) Free RAM (bytes): 349 (99%)	Used RAM (bytes): 3 (1%)
0	1144	Used ROM (program words): 41 (1%) Free ROM (program words): 8151 (99%)	Used ROM (program words): 41 (1%)
0	125	Project Linked Successfully	Askis1.mcppi
0	128	Linked in 327 ms	
0	129	Project 'Askis1.mcppi' completed: 468 ms	
0	103	Finished successfully: 29 Mar 2013, 09:44:44	Askis1.mcppi

Στη συνέχεια μετά το **compilation** (συνήθως επιλέγοντας **Build All**) παράγεται από το περιβάλλον μεταξύ άλλων και ένα αρχείο με κατάληξη **.hex**. Τα περιεχόμενα αυτού του αρχείου είναι σε δεκαεξαδική μορφή και είναι εκείνα που θα 'φορτωθούν' στο ολοκληρωμένο του μικροελεγκτή. Για το λόγο αυτό είναι καλό για κάθε εφαρμογή που αναπτύσσουμε να δημιουργούμε εξ αρχής ένα ξεχωριστό φάκελο. Στην παρακάτω φωτογραφία φαίνεται η μορφή ενός αρχείου τύπου .hex χρησιμοποιώντας συνήθως την εφαρμογή **Σημειωματάριο (Notepad)**.

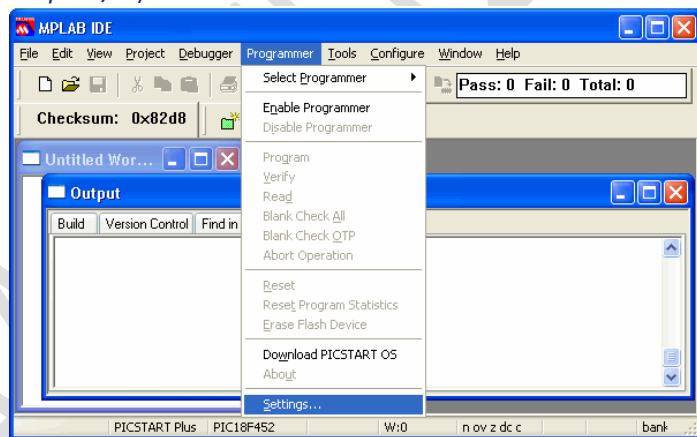


```
test2 - Σημειωματάριο
Αρχείο Επεξεργασία Μορφή Προβολή Βοήθεια
:100000016EF00F000000000000EF00F000000001C
:0A0010000000000000000000000000F3D71C
:10001C000900F5CFE6FF0006FBE10106F9E112004D
:10002C00306AF26A060EC16E816A806AFF0E936EA8
:10003C00926A8150010A01E1302A8150020A01E1E1
:10004C00300630C080FF060E0B6E130E0C6EAD0E1C
:10005C000D6E0D2EFED70C2EFCDD70B2EFAD70000F2
:06006C0000000E7D7FFD7FA
:020000040030CA
:0D000100220F0EFF0181FF0FC00FE00F4026
:00000001FF
```

Προγραμματισμός μικροελεγκτή: εννοούμε τη μεταφορά του αρχείου με κώδικα μηχανής (.HEX file) από τον προσωπικό Η/Υ στο ολοκληρωμένο κύκλωμα (IC) του μικροελεγκτή (στο σχολείο συνήθως είναι ο PIC16F877)

Προγραμματίστριες μονάδες για PIC: είναι οι ηλεκτρονικές πλακέτες στις οποίες τοποθετούμε τον μικροελεγκτή για να του μεταφέρουμε το αρχείο .hex από τον προσωπικό Η/Υ. Για τη μεταφορά ενεργοποιείται δεύτερο λογισμικό συνήθως μέσα από το αντίστοιχο κάθε φορά περιβάλλον προγραμματισμού. Τέτοιες πλακέτες είναι π.χ.:

- η **PICSTART Plus** που υποστηρίζεται από το περιβάλλον **MPLAB** που συνήθως συνδέεται με σειριακό καλώδιο στον Η/Υ. Στην περίπτωση που ο Η/Υ μας δεν διαθέτει σειριακό βύσμα 9 ακίδων τότε χρησιμοποιούμε ενδιάμεσους μετατροπείς USB to Serial και συνδέουμε την προγραμματίστρια σε μια θύρα USB του Η/Υ. (Δείτε στην ιστοσελίδα του σχολείου το φύλλο έργου: *Οδηγίες χρήσης της σειριακής διασύνδεσης ηλεκτρονικών συσκευών και μικροελεγκτών με Η/Υ.*)



Αυτή η πλακέτα υπάρχει στο σχολείο και τα βήματα που ακολουθούμε είναι τα παρακάτω: Τα βήματα για τη μεταφορά του αρχείου .hex είναι:

Βήμα 1^ο: Γράφουμε τον κώδικα σε γλώσσα C και συνήθως με την επιλογή Build All γίνεται έλεγχος σφαλμάτων και η παραγωγή του αρχείου .hex που παραμένει φορτωμένο και αυτό στη μνήμη RAM του Η/Υ.

εφόσον το αρχείο .hex έχει φτιαχτεί σε κάποια άλλη χρονική στιγμή τότε μέσω του IDE MPLAB της MICROCHIP από τη διαδρομή **File/ Import / Import to Memory** φορτώνουμε το αρχείο .hex στο περιβάλλον του MPLAB.

Βήμα 2^ο: Στη συνέχεια καθορίζουμε τα **Configuration bits** μέσα από το μενού **Configure** του MPLAB

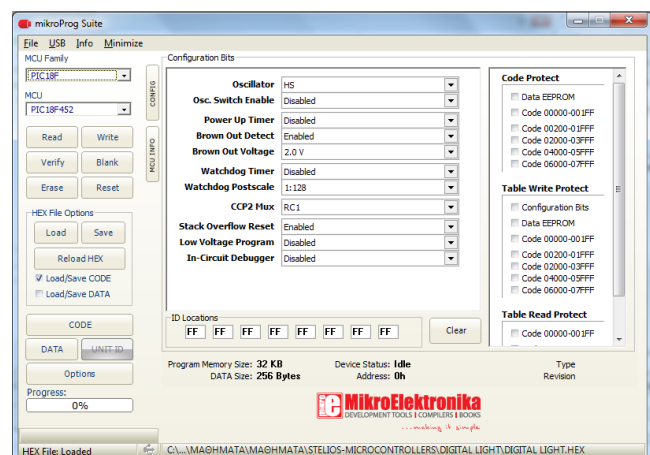
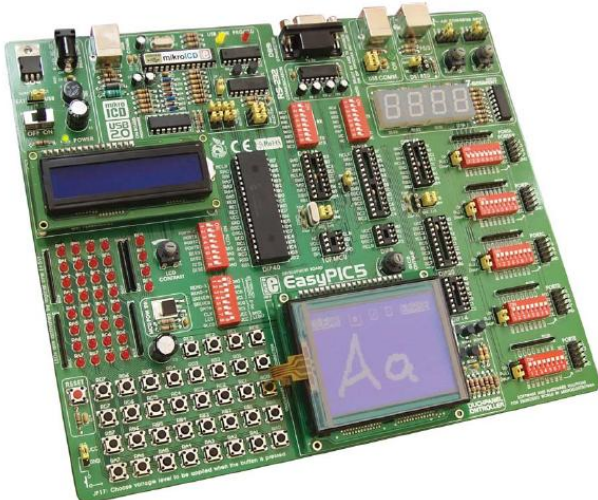
Βήμα 3^ο: η μεταφορά του αρχείου .hex γίνεται με την επιλογή του προγραμματιστή μέσα από τη διαδρομή **Programmer/Select Programmer/ PicStart Plus** και τις επιλογές **Enable Programmer** και **Program**.

Η παραπάνω πλακέτα υποστηρίζει πολύ μεγάλο αριθμό μικροελεγκτών PIC.

- η **USB-ICD2 Pro**: Εναλλακτικά και με πολύ μικρό κόστος μπορούμε να προμηθευτούμε την πλακέτα που υποστηρίζει πάλι μικροελεγκτές PIC ενδεχομένως όμως για μικρότερη γκάμα ολοκληρωμένων. Η πλακέτα αυτή υποστηρίζεται από πιο καινούργια προγραμματιστικά περιβάλλοντα για PIC όπως το **MPLAB ICD2 ver 8.9** και συνδέεται με τον Η/Υ μέσω καλωδίου UTP. Είναι απαραίτητη η εγκατάσταση των αντίστοιχων οδηγών που πρέπει να είναι συμβατοί και με το λειτουργικό σύστημα του Η/Υ που χρησιμοποιούμε.

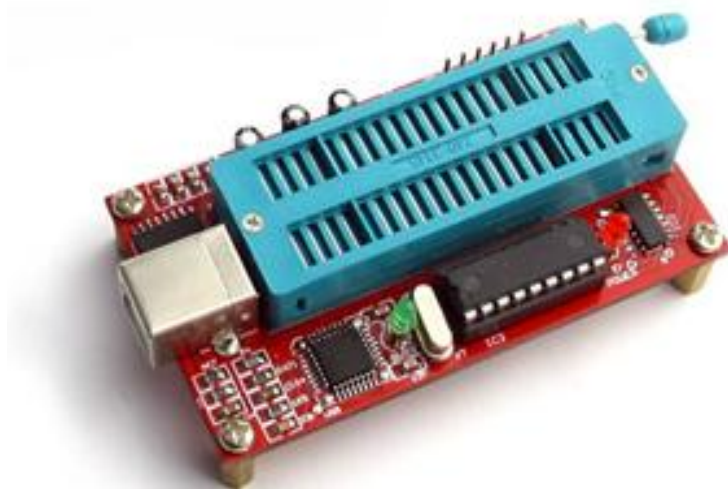


- η **EasyPic5 ή EasyPic6**: Είναι πλακέτα της εταιρείας Mikroelektronika (www.mikroe.com) με την οποία μπορούμε και να προγραμματίσουμε (να μεταφέρουμε δηλαδή το αρχείο .hex) στο ολοκληρωμένο του μικροελεγκτή PIC αλλά επίσης και να δοκιμάσουμε την εφαρμογή μας επειδή διαθέτει συνδεδεμένα παρά πολλά περιφερειακά κυκλώματα. Έτσι πριν μεταφέρουμε τον μικροελεγκτή στο δικό μας ηλεκτρονικό κύκλωμα μπορούμε να δοκιμάσουμε το πρόγραμμα της εφαρμογής μας απευθείας στην αναπτυξιακή αυτή πλακέτα. Συνεργάζεται με το Ολοκληρωμένο Προγραμματιστικό Περιβάλλον Mikro C Pro για την ανάπτυξη του κώδικα. Η παραγωγή του αρχείου .hex γίνεται με τοπάτημα του κουμπιού **Build All** ενώ η μεταφορά του αρχείου .hex στο ολοκληρωμένου του μικροελεγκτή γίνεται με το πάτημα του κουμπιού **Program**. Η μεταφορά γίνεται με την αυτόματη φόρτωση της εφαρμογής **mikroProg Suite**.



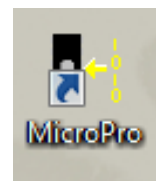
(για περισσότερες λεπτομέρειες ανατρέξτε στο Φύλλο Έργου: 'Οδηγίες βήμα προς βήμα για την Άσκηση 1 και τη διαχείριση Ψηφιακών Εξόδων'). Είναι απαραίτητη η εγκατάσταση των αντίστοιχων οδηγών που πρέπει να είναι συμβατοί και με το λειτουργικό σύστημα του Η/Υ που χρησιμοποιούμε. Η σύνδεση της αναπτυξιακής πλακέτας EasyPic5 γίνεται με αντίστοιχο καλώδιο σε θύρα USB του Η/Υ αφού πρώτα έχει εγκατασταθεί ο κατάλληλος οδηγός (driver) και έχει αναγνωρισθεί η πλακέτα από τη **Διαχείριση Συσκευών** του Η/Υ μας.

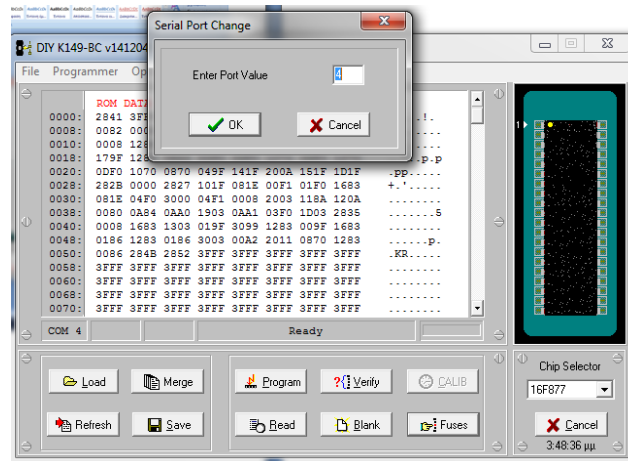
- η **USB PIC Programmer K149-BC** της **DIY Electronics** που φαίνεται στην φωτογραφία και για την οποία εγκαθιστούμε τον αντίστοιχο driver για το λειτουργικό σύστημα του Η/Υ μας.



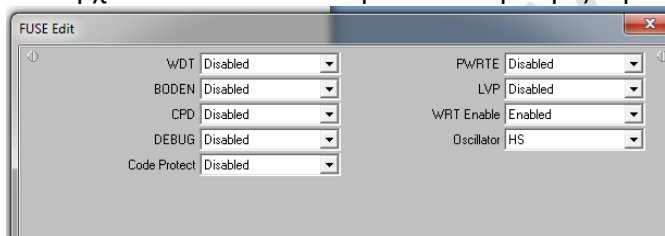
Οι οδηγίες για τη χρησιμοποίηση της συγκεκριμένης πλακέτας είναι οι παρακάτω:

- 1) Συνδέουμε σε μία θύρα USB την **USB PIC Programmer** με στοιχεία **K149-BC** της **DIY Electronics**. Περιμένουμε να ανιχνευθεί και να εγκατασταθεί ο driver και εντοπίζουμε σε ποια θύρα COM φαίνεται από το λειτουργικό σύστημα του Η/Υ μας (δεξιά κλικ **Η/Υ μου / Διαχείριση Συσκευών / Θύρες COM**).
- 2) Εγκαθιστούμε και τρέχουμε την εφαρμογή **MicroPro** που συνοδεύει την προγραμματίστρια φροντίζοντας η έκδοσή της να είναι συμβατή με το λειτουργικό σύστημα. Εμείς βρήκαμε έκδοση συμβατή έως τα **WINDOWS7 32 bits**.
- 3) Από τη διαδρομή **File/Serial Port** δηλώνουμε τον αριθμό της **Com Port** που ανιχνεύτηκε η προγραμματίστρια από τη **Διαχείριση Συσκευών** του Η/Υ μας.

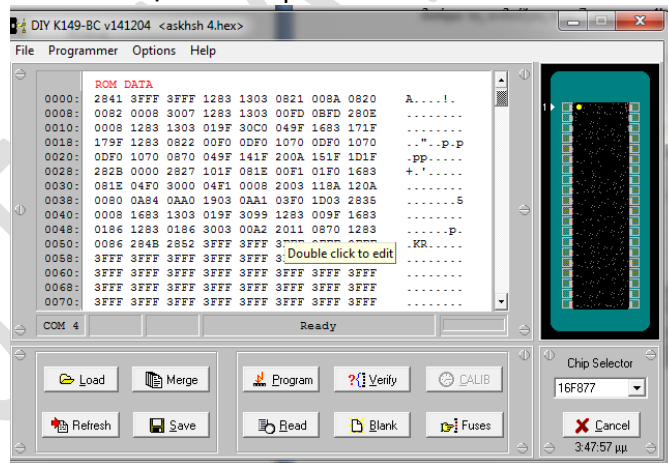




- 4) Τοποθετούμε ένα ολοκληρωμένο PIC16F877 στην προγραμματίστρα και επιλέγουμε τον 16F877 στο πεδίο **Chip Selector** ενώ με το κουμπί **Load** φορτώνουμε το αρχείο **.hex**. Με το κουμπί **Fuses** ρυθμίζουμε:



- 5) Στο παράθυρο ROM DATA βλέπουμε το αρχείο **.hex**. Με το κουμπί **Blank** επιλέγουμε **Erase Chip** για να διαγράψουμε τα προηγούμενα περιεχόμενα του chip και στη συνέχεια πατάμε το κουμπί **Program** για να μεταφερθεί ο κώδικας hex από τον Η/Υ στο chip.



Μετά τη μεταφορά του αρχείου **.hex** στον μικροελεγκτή, τοποθετούμε το chip του μικροελεγκτή στην εκπαιδευτική πλακέτα **SE1001** που βρίσκεται στο εργαστήριο του Ε.Κ., την τροφοδοτούμε με τάση 9-12 Volts οπότε βλέπουμε πως λειτουργεί ο μικροελεγκτής με βάση την εφαρμογή που γράψαμε.

