

**Μάθημα:** Εργαστήριο Ψηφιακών Ηλεκτρονικών Συστημάτων

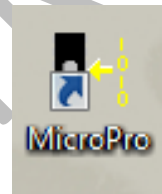
**Αντικείμενο:** Μέτρηση αναλογικού σήματος με PIC16F877

Αντιστοιχεί στην ΑΣΚΗΣΗ 4η: Μετατροπέας αναλογικού σε ψηφιακό(A/D Converter)

Υλικά που απαιτούνται:

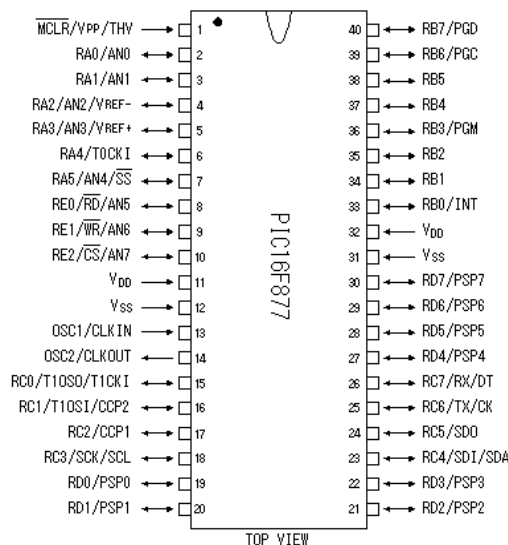
- Η εκπαιδευτική πινακίδα SE1001
- Ένας μ/σ PIC16F877
- Τροφοδοτικό 9-12 Volts
- Εγκατεστημένη γλώσσα προγραμματισμού mikroC για να γράψουμε τον κώδικα
- Προγραμματίστρια MICROCHIP PicStart Plus με εγκατεστημένο το περιβάλλον MPLAB (συνήθως με σειριακό καλώδιο με ακροδέκτη DB9) **ή** οποιαδήποτε άλλη USB προγραμματίστρια για μ/σ PIC με εγκατεστημένο το αντίστοιχο περιβάλλον προγραμματισμού που να είναι συμβατό με το λειτουργικό σύστημα του Η/Υ που χρησιμοποιούμε. Στην άσκηση αυτή θα χρησιμοποιήσουμε την **USB PIC Programmer** με στοιχεία **K149-BC** της **DIY Electronics** που φαίνεται στην φωτογραφία και εγκαθιστούμε τον αντίστοιχο driver για το λειτουργικό σύστημα .

Για τη μεταφορά του κώδικα hex εγκαθιστούμε το λογισμικό **MicroPro** που τρέχει σε λειτουργικό σύστημα



Αντικείμενο της άσκησης:

Στην εκπαιδευτική πινακίδα SE1001 η μεταβλητή αντίσταση R14 συνδέεται στον ακροδέκτη 5 (**RA3/AN3**) μεταφέροντας του μια μεταβαλλόμενη αναλογική τάση από 0 έως 5 Volts. Το αντικείμενο της άσκησης είναι να μετράμε με τον PIC αυτή την αναλογική τάση και μέσα από τη μετατροπή της από αναλογική σε ψηφιακή μορφή να την απεικονίζουμε στα LEDs που είναι συνδεδεμένα στους ακροδέκτες 40-33 (**RB7-RB0**).



Οι καταχωρητές που ρυθμίζουν την A/D λειτουργία του PIC16F877:

Ο PIC16F877 διαθέτει 8 αναλογικές εισόδους (AN0-AN7) οι οποίες αντιστοιχούν στους ακροδέκτες 2-5 και 7-10. Κάθε φορά όμως μπορούμε να μετράμε μόνο από έναν ακροδέκτη από τους παραπάνω. Το περιφερειακό A/D που είναι ενσωματωμένο στον PIC16F877 έχει τέσσερις καταχωρητές που ρυθμίζουν τη λειτουργία του, τους ADRESH, ADRESL, ADCON1 και ADCON0.

**Διακριτική ικανότητα (resolution)** : 10 bits (b9-b0) και η ψηφιακή λέξη απεικονίζεται στους καταχωρητές **ADRESH** και **ADRSEL**. Επειδή όμως οι καταχωρητές του μικροελεγκτή έχουν μήκος 8 bits στην άσκηση αυτή θα ρυθμίσουμε ώστε το ψηφιακό αποτέλεσμα να αποθηκεύεται όπως παρακάτω:

ADRESH							
b9	b8	b7	b6	b5	b4	b3	b2

ADRESL							
b1	b0						

Η ρύθμιση αυτή γίνεται θέτοντας το **ADFM** bit του καταχωρητή ADCON1 ίσο με 0. Θέτοντας επίσης τα **PCFG3-PCFG0** bits και αυτά 0 δηλώνουμε ότι όλοι οι ακροδέκτες 2-5 και 7-10 θα χρησιμοποιούνται σαν αναλογικές εισοδοί. Μπορούμε να ερευνήσουμε το datasheet του PIC16F877 για να δούμε με ποιους συνδυασμούς μπορούμε να δηλώσουμε κάποιους από τους ακροδέκτες 2-5 και 7-10 να λειτουργούν ως αναλογικές εισοδοί και κάποιοι ως ψηφιακές εισοδοί/έξοδοι.

#### ADCON1 REGISTER (ADDRESS 9Fh)

R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
ADFM	ADCS2	—	—	PCFG3	PCFG2	PCFG1	PCFG0
bit 7							bit 0

Επομένως η εντολή για τις παραπάνω ρυθμίσεις είναι η : **ADCON1 = 0b00000000;**

**Χρόνος μετατροπής (conversion time)** : με τον όρο **TAD** ορίζουμε το χρόνο μετατροπής για κάθε bit. Σύμφωνα με το τεχνικό εγχειρίδιο για το σύνολο των 10 bits απαιτείται ένας χρόνος μετατροπής 12x **TAD**. Ο χρόνος μετατροπής μπορεί να επιλεγεί μέσω του προγραμματισμού ορίζοντας κατάλληλα τα ADCS1 και ADCS0 bits του καταχωρητή ADCON0 καθώς και το ADCS2 bit ADCON1. Στην άσκηση αυτή οι μεταβολές του αναλογικού σήματος από τη χειροκίνητη μεταβλητή αντίσταση είναι πολύ αργές και επειδή στην εκπαιδευτική πινακίδα το κύκλωμα χρονισμού του PIC16F877 έχει κρύσταλλο 4 MHz ορίζουμε:

ADCS1=1 ADCS0=0 ADCS2=0 που καθορίζουν χρόνο μετατροπής **TAD = 32x TOSC**

όπου **TOSC=1/4MHz=0.25 μsec** άρα επιλέξαμε **TAD = 32x0.25 = 8 μsec**

Επομένως για να περιορίσουμε τα σφάλματα στη μετατροπή ο συνολικός χρόνος που απαιτείται είναι:

$$12 \times 8 \mu\text{sec (A/D)} + 16 \times 0.25 \mu\text{sec (για την εκτέλεση των εντολών)} = 20 \mu\text{sec}$$

που αντιστοιχεί σε συχνότητα μετατροπής **1/20 μsec=50 KHz**

άρα το μεταβαλλόμενο αναλογικό σήμα με βάση το θεώρημα δειγματοληψίας πρέπει να έχει μέγιστη συχνότητα μικρότερη από το μισό των 50 KHz.

#### ADCON0 REGISTER (ADDRESS 1Fh)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0
ADCS1	ADCS0	CHS2	CHS1	CHS0	GO/DONE	—	ADON
bit 7						bit 0	

**Επιλογή καναλιού μέτρησης (analog channel)** : για να επιλέξουμε τον ακροδέκτη αναλογικού σήματος (AN0-AN7) χρησιμοποιούμε τα bits CHS2, CHS1 και CHS0 πάλι του καταχωρητή ADCON0.

- για να σκανδαλίσουμε την A/D μετατροπή πρέπει επίσης να θέσουμε το bit ADON=1 επομένως η εντολή για τις παραπάνω ρυθμίσεις είναι η : **ADCON1 = 0b10011001;**
- Για όσους ακροδέκτες έχουμε δηλώσει ότι είναι αναλογικοί είσοδοι πρέπει να τους δηλώσουμε ως εισόδους με τον καταχωρητή TRIS άρα η εντολή είναι: **TRISA = 0xFF;**

**Ακρίβεια της μέτρησης (accuracy)**: καθορίζεται από το  $\Delta V_{ref}/1024$  (10 bit A/D)

Η  $\Delta V_{ref}$  μπορεί να οριστεί εξωτερικά με τους ακροδέκτες 4 και 5. Ωστόσο στην εκπαιδευτική πινακίδα SE1001 ο ακροδέκτης 5 χρησιμοποιείται σαν είσοδος του αναλογικού σήματος. Σύμφωνα με το τεχνικό εγχειρίδιο όταν δεν χρησιμοποιούμε εξωτερική τάση αναφοράς τότε σαν τάση αναφοράς ορίζεται η τάση τροφοδοσίας του PIC που στην περίπτωση της πινακίδας SE1001 είναι τα 5 Volts. Επομένως και το αναλογικό σήμα που συνδέουμε στον ακροδέκτη 5 δεν μπορεί να υπερβαίνει τα 5 Volts. Άρα η ακρίβεια της μέτρησης είναι  $5/1024=5$  mVolts. Στην άσκηση μας θα έχουμε και επιπλέον αποκλίσεις επειδή θα διαβάζουμε μόνο τα 8 πιο σημαντικά από τα 10 bits του ψηφιακού αποτελέσματος.

Τα βήματα για την υλοποίηση της άσκησης είναι τα παρακάτω:

- 1) Ακολουθούμε τις οδηγίες Άσκησης 1 και δημιουργούμε φάκελο με όνομα **Askisi4** μέσα στον οποίο θα αποθηκευτούν όλα τα αρχεία που δημιουργούνται από το περιβάλλον της MikroC μεταξύ των οποίων το αρχείο project (.mcpri), αρχείο κώδικα C (.c) και το αρχείο hex. Το όνομα των αρχείων μπορεί να είναι το **Askisi4**.
- 2) Στο παράθυρο κώδικα γράφουμε τις παρακάτω εντολές και συμπληρώνουμε τα σύντομα σχόλια για τη λειτουργία τους

```
unsigned int temp_res; //δήλωση μεταβλητής για αποτέλεσμα της A/D μετατροπής
void main() { // έναρξη κυρίως προγράμματος. Υπάρχει ήδη στον κώδικα

    ADCON1 = 0b00000000; // .....
    ADCON0 = 0b10011001; // .....

    TRISA = 0xFF; // .....
    TRISB = 0x00; // .....
    PORTB=0; // .....

    do { // .....

        temp_res = ADC_Read(3); // .....

        PORTB=temp_res; // .....
```

```

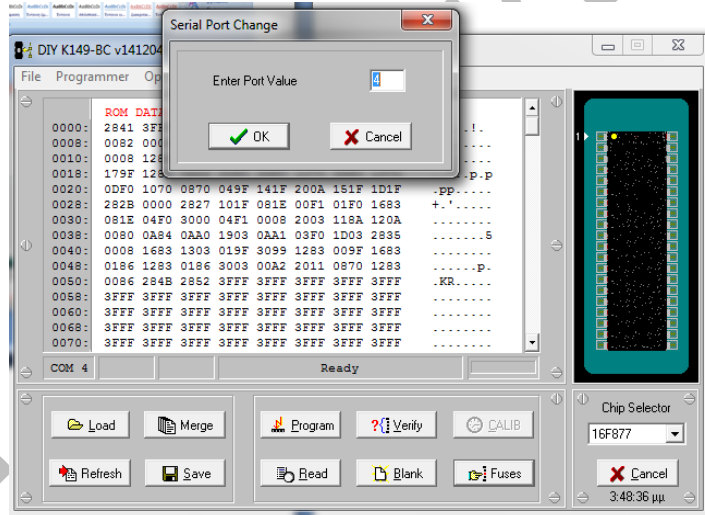
} while(1);           // .....
}                     //τέλος προγράμματος. Υπάρχει ήδη στον κώδικα

```

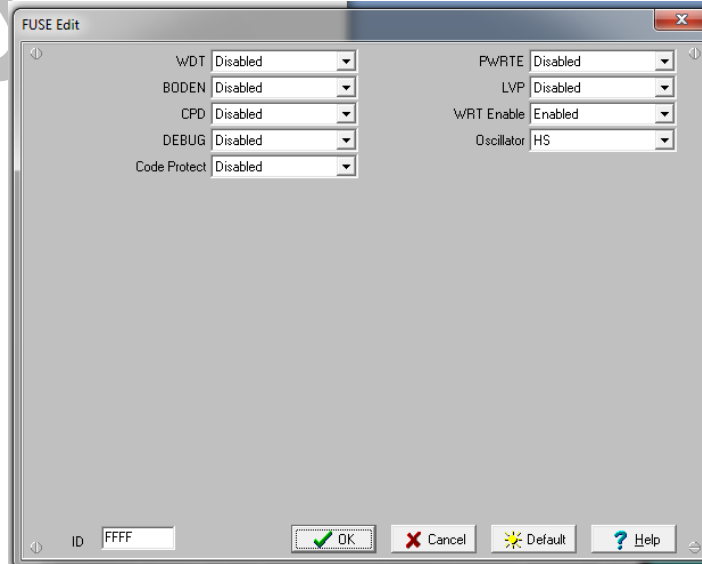
- 3) Κάνουμε συμβολομετάφραση (compilation) ώστε να διορθωθούν τα συντακτικά λάθη και να παραχθεί το αρχείο hex (οδηγίες στην Άσκηση 1)
- 4) Στη συνέχεια συνδέουμε σε μία θύρα USB την **USB PIC Programmer** με στοιχεία **K149-BC** της **DIY Electronics**. Περιμένουμε να ανιχνευθεί και να εγκατασταθεί 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** ).
- 5) 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**.



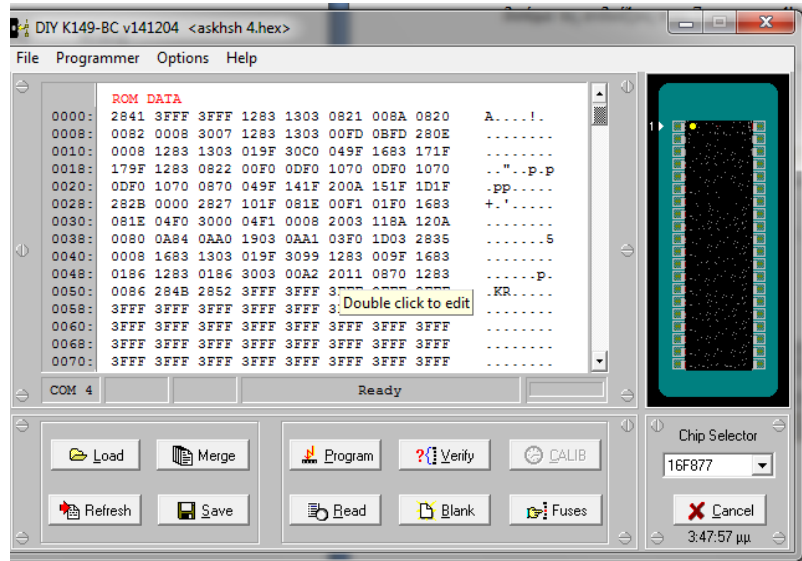
6) Από τη διαδρομή **File/Serial Port** δηλώνουμε τον αριθμό της **Com Port** που ανιχνεύτηκε η προγραμματίστρια.



7) Τοποθετούμε ένα ολοκληρωμένο PIC16F877 στην προγραμματίστρια και επιλέγουμε τον 16F877 στο πεδίο **Chip Selector** ενώ με το κουμπί **Load** φορτώνουμε το αρχείο **Askisi4.hex**. Με το κουμπί **Fuses** ρυθμίζουμε:



8) Στο παράθυρο ROM DATA βλέπουμε το αρχείο askisi4.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 H/Y στο chip.



9) Μεταφέρουμε το chip στην εκπαιδευτική πλακέτα SE1001 και την τροφοδοτούμε με τάση 9-12 Volts. Μεταβάλλουμε την αναλογική τάση και με ένα πολύμετρο μετράμε την αναλογική τάση στον ακροδέκτη AN3 και συμπληρώνουμε τον παρακάτω πίνακα:

Μέτρηση πολυμέτρου (Volts)	b9 RB7	b8 RB6	b7 RB5	b6 RB4	b5 RB3	b4 RB2	b3 RB1	b2 RB0	Υπολογιζόμενη τάση από τον A/D = $res * 5 / 1023$ με res η δεκαδική τιμή της ψηφιακής λέξης

Όνοματεπώνυμο : ..... Ημερομηνία: .....

**Βιβλιογραφία :**

- 1) « **Εισαγωγή στον προγραμματισμό μικροελεγκτών, FPGA και CPLD: Επιλεγμένες Εφαρμογές**» Σ. Μπουλταδάκης, Γ. Πατουλίδης και Ν. Ασημόπουλος, Εκδόσεις ΤΖΙΟΛΑ, Θεσσαλονίκη 2011, ISBN: 978-960-418-291-6
- 2) « **Υλικό και Λογισμικό Μετρήσεων: Παραδείγματα και Εφαρμογές**» Σ. Μπουλταδάκης και Ι. Καλόμοιρος, Εκδόσεις ΤΖΙΟΛΑ, Θεσσαλονίκη 2009, ISBN: 978-960-418-161-2

