



4^η ΕΡΓΑΣΤΗΡΙΑΚΗ ΑΣΚΗΣΗ
ΓΙΑ ΤΟ ΜΑΘΗΜΑ "Εργαστήριο Μικροϋπολογιστών"
3^η Εργ. Άσκ. στον Μικροελεγκτή AVR – Χρονιστές, ADC
(υλοποίηση στο εκπαιδευτικό σύστημα easyAVR6)

Εξέταση – Επίδειξη: Τετάρτη 9/12/2020.
Προθεσμία για παράδοση Έκθεσης: Κυριακή 15/12/2020

Χρονιστής Timer1

Ένα πλεονέκτημα των σύγχρονων μικροελεγκτών είναι η ενσωμάτωση στην ίδια ψηφίδα χρήσιμων περιφερειακών συσκευών, με σύνδεση στο σύστημα διακοπών του Μικροελεγκτή. Για παράδειγμα, οι χρονιστές είναι καταχωρητές που μπορούν να προγραμματιστούν να προκαλέσουν διακοπή στον Μικροελεγκτή μετά την πάροδο συγκεκριμένου χρόνου.

Ο Μικροελεγκτής AVR ATmega16 διαθέτει 2 χρονιστές. Τον 8-ψήφιο και απλούστερο TCNT0 (διεύθυνση \$32) και τον 16-ψήφιο TCNT1 (TCNT1H, διεύθυνση \$2D και TCNT1L, διεύθυνση \$2C). Στους χρονιστές αυτούς μπορεί να τοποθετηθεί μια αρχική τιμή η οποία αυξάνεται από το Μικροελεγκτή με επιλεγμένη συχνότητα. Όταν ένας χρονιστής υπερχειλίζει, μπορεί να δημιουργήσει κατάλληλο σήμα διακοπής εφόσον τροποποιηθεί το περιεχόμενο του καταχωρητή TIMSK (διεύθυνση \$39) σύμφωνα με το παρακάτω σχήμα. Γράφοντας 1 στο ψηφίο TOIE0 επιτρέπονται διακοπές υπερχειλίσσης του Χρονιστή TCNT0 και αντίστοιχα με 1 στο ψηφίο TOIE1 επιτρέπονται διακοπές υπερχειλίσσης του Χρονιστή TCNT1 (εφόσον επιτραπούν και γενικά οι διακοπές με την εντολή sei).

TIMSK:

| | | | | | | | |
|--|--|--|--|--|--|--|--|
| | | | | | | | |
|--|--|--|--|--|--|--|--|

| | | |
|-------|--|-------|
| TOIE1 | | TOIE0 |
|-------|--|-------|

Για παράδειγμα, με τις παρακάτω εντολές επιτρέπεται η διακοπή υπερχειλίσσης του μετρητή TCNT1.

ldi r24, (1 << TOIE1) ; ενεργοποίηση διακοπής υπερχειλίσσης του μετρητή TCNT1
out TIMSK, r24 ; για τον timer1

Η επιλογή συχνότητας αύξησης του κάθε χρονιστή γίνεται διαφορετικά. Για το Χρονιστή TCNT1 χρησιμοποιούμε τον καταχωρητή TCCR1B (διεύθυνση \$2E) σύμφωνα με το παρακάτω σχήμα και τον παρακάτω πίνακα.

TCCR1B:

| | | | | | | | |
|--|--|--|--|--|--|--|--|
| | | | | | | | |
|--|--|--|--|--|--|--|--|

| | | |
|------|------|------|
| CS12 | CS11 | CS10 |
|------|------|------|

| CS12 | CS11 | CS10 | Περιγραφή σήματος εισόδου χρονιστή |
|------|------|------|--|
| 0 | 0 | 0 | Κανένα. Χρονιστής σταματημένος |
| 0 | 0 | 1 | CLK |
| 0 | 1 | 0 | CLK/8 |
| 0 | 1 | 1 | CLK/64 |
| 1 | 0 | 0 | CLK/256 |
| 1 | 0 | 1 | CLK/1024 |
| 1 | 1 | 0 | Κατερχόμενη ακμή εξωτερικού σήματος ακροδέκτη T1 |

| | | | |
|---|---|---|---|
| 1 | 1 | 1 | Ανερχόμενη ακμή εξωτερικού σήματος ακροδέκτη T1 |
|---|---|---|---|

Για παράδειγμα, με τις παρακάτω εντολές επιλέγεται συχνότητα αύξησης του Χρονιστή TCNT1 ίση με τη 1/1024 της συχνότητας ρολογιού του Μικροελεγκτή (δηλαδή, κάθε 1024 κύκλους ρολογιού αυξάνεται κατά 1 ο Χρονιστής TCNT1).

```
ldi r24 ,(1<<CS12) | (0<<CS11) | (1<<CS10) ; CK/1024
out TCCR1B ,r24
```

Στην αναπτυξιακή πλακέτας EasyAVR6 με συχνότητα ρολογιού 8MHz, η επιλογή αυτή ισοδυναμεί με συχνότητα αύξησης του TCNT1 ίση με 8MHz/1024=7812.5Hz. Αν με αυτές τις επιλογές θέλουμε ο TCNT1 να δημιουργήσει σήμα διακοπής υπερχειλίσσης μετά από για παράδειγμα 5sec, πρέπει να τον κάνουμε να μετρήσει $5 \times 7812.5 = 39062.5$ κύκλους. Επειδή η υπερχειλίση γίνεται όταν μετρήσει 65536 κύκλους (16 ψηφία), θα πρέπει η αρχική τιμή που θα του δοθεί πριν αρχίσει να μετράει προς τα πάνω να είναι $65536 - 39062.5 = 26473.5 = 0x6769$. Αυτό γίνεται με τον παρακάτω κώδικα:

```
ldi r24,0x67 ; αρχικοποίηση του TCNT1
out TCNT1H ,r24 ; για υπερχειλίση μετά από 5 sec
ldi r24 ,0x69
out TCNT1L ,r24
```

Για μέγιστη ασφάλεια, τα δύο τμήματα του μετρητή TCNT1 TCNT1H και TCNT1L πρέπει να διαβάζονται αδιαίρετα, χωρίς να μεσολαβήσει για παράδειγμα κάποια άλλη διακοπή. Για το λόγο αυτό ο Μικροελεγκτής κάνει μια ειδική διαδικασία. Όταν μια τιμή γράφεται στον TCNT1H αυτή τοποθετείται στον προσωρινό καταχωρητή TEMP. Στη συνέχεια, όταν γραφεί τιμή στον TCNT1L η τιμή που υπάρχει στον TEMP συνδυάζεται με αυτή και τα 16 ψηφία γράφονται ταυτόχρονα σε όλο το μήκος του TCNT1. Κατά την ανάγνωση, όταν διαβάζεται μια τιμή από τον TCNT1L αυτή τοποθετείται στον επιλεγμένο καταχωρητή του Μικροελεγκτή και ταυτόχρονα η τιμή του TCNT1H μεταφέρεται στον καταχωρητή TEMP. Όταν στην συνέχεια διαβαστεί και ο TCNT1H, μεταφέρεται στον επιλεγμένο καταχωρητή η τιμή που έχει τοποθετηθεί στον TEMP. Με βάση αυτή τη διαδικασία κατά την εγγραφή πρέπει πάντα να γράφεται πρώτα η τιμή στον TCNT1H και μετά στον TCNT1L ενώ κατά την ανάγνωση πρέπει πάντα να διαβάζεται πρώτα ο TCNT1L και μετά ο TCNT1H. Σε πολύπλοκες εφαρμογές πριν την πρόσβαση στον TCNT1 μπορούν να απενεργοποιηθούν και οι διακοπές (αν και δεν συνηθίζεται). Τέλος, η θέση μνήμης του διανύσματος διακοπής του Χρονιστή TCNT1 φαίνεται στον παρακάτω κώδικα.

```
.org 0x10
rjmp ISR_TIMER1_OVF ; ρουτίνα εξυπηρέτησης της διακοπής υπερχειλίσσης του timer1
```

Μετατροπείας αναλογικής σε ψηφιακή μορφή ADC.

Βασικό περιφερειακό των μικροελεγκτών είναι ο μετατροπείας από Αναλογική σε Ψηφιακή (ADC) μορφή για την εισαγωγή, τον έλεγχο και την επεξεργασία αναλογικών ποσοτήτων από τον εξωτερικό φυσικό κόσμο, όπως θερμοκρασία, πίεση, βάρος. Γενικά, επιτρέπει τη μετατροπή σε ψηφιακή μορφή διαφορών δυναμικού που παράγονται από αισθητήρες. Η λειτουργία ενός ADC περιγράφεται ως εξής:

- Θεωρούμε αναλογική τάση εισόδου V_{in} και μία τάση αναφοράς V_{REF} βάση της οποίας γίνεται η μετατροπή.
- Επίσης θεωρούμε ότι για να γίνει σωστά η μετατροπή πρέπει $V_{in} < V_{REF}$
- Για ADC ανάλυσης n-bit, χωρίζεται η δυναμική περιοχή ($0 - V_{REF}$) σε 2^n ίσα διαστήματα και αντιστοιχείται το καθένα σε ένα δυαδικό αριθμό των n-bit ανάλογα με την τάξη του. Το κάθε διάστημα Q που αντιπροσωπεύει την ανάλυση του ADC είναι $Q = V_{REF} / 2^n$. Αν υποθέσουμε ότι $V_{REF} = 2,56V$ και $n=10$ τότε $Q=2,5mV$.
- Γενικά η ψηφιακή έξοδος X του ADC είναι $X = \lceil V_{in} \cdot 2^n / V_{REF} \rceil = \lceil V_{in} / Q \rceil$ όπου με το σύμβολο $\lceil \cdot \rceil$ δηλώνεται το στρογγυλευμένο ακέραιο μέρος.

Ο Μικροελεγκτής AVR ATmega16 διαθέτει έναν ADC ανάλυσης 10 bit, που βασίζεται στη μέθοδο των διαδοχικών προσεγγίσεων. Ο ADC είναι συνδεδεμένος με ένα πολυπλέκτη 8 εισόδων που επιτρέπει την απλή μετατροπή των

τάσεων από τα pins της θύρας PORTA. Επιτρέπει επίσης και μετατροπή από διαφορικές εισόδους (16 συνδυασμοί των 8 εισόδων), που συνδυάζονται και με ένα στάδιο προγραμματιζόμενης (x1, x10, x200) ενίσχυσης.

Περισσότερες πληροφορίες σχετικά με την λειτουργία των καταχωρητών αλλά και γενικότερα τον ADC μπορείτε να βρείτε στις σελίδες 133-137 του βιβλίου Συστήματα Μικροϋπολογιστών ii ή και στο [ATmega16](#).

Καταχωρητές

Καταχωρητής επιλογής πολυπλέκτη (ADMUX)

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|---------------|--------------|--------------|--------------|-------------|-------------|-------------|-------------|-------------|--------------|
| | REFS1 | REFS0 | ADLAR | MUX4 | MUX3 | MUX2 | MUX1 | MUX0 | ADMUX |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | |
| Initial Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |

REFS1:0: Χρησιμοποιείται για την επιλογή του V_{REF} σύμφωνα με τον επόμενο πίνακα:

Table 83. Voltage Reference Selections for ADC

| REFS1 | REFS0 | Voltage Reference Selection |
|-------|-------|--|
| 0 | 0 | AREF, Internal Vref turned off |
| 0 | 1 | AVCC with external capacitor at AREF pin |
| 1 | 0 | Reserved |
| 1 | 1 | Internal 2.56V Voltage Reference with external capacitor at AREF pin |

MUX4:0: Χρησιμοποιείται για την επιλογή των εισόδων που θα οδηγηθούν από τον πολυπλέκτη στον ADC καθώς και για την ρύθμιση του κέρδους. Στον παρακάτω πίνακα φαίνονται μόνο οι επιλογές που έχουν κέρδος 1 και επιτρέπουν την απλή μετατροπή των τάσεων από τα pins της θύρας PORTA.

| MUX4..0 | Single Ended Input |
|---------|--------------------|
| 00000 | ADC0 |
| 00001 | ADC1 |
| 00010 | ADC2 |
| 00011 | ADC3 |
| 00100 | ADC4 |
| 00101 | ADC5 |
| 00110 | ADC6 |
| 00111 | ADC7 |

Καταχωρητής Ελέγχου και Κατάστασης του ADC (ADCSRA)

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|---------------|-------------|-------------|--------------|-------------|-------------|--------------|--------------|--------------|---------------|
| | ADEN | ADSC | ADATE | ADIF | ADIE | ADPS2 | ADPS1 | ADPS0 | ADCSRA |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | |
| Initial Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |

ADEN: ADC Enable

Αν το bit είναι 1 ο ADC τίθεται σε κατάσταση λειτουργίας.

ADSC: ADC Start Conversion

Αν το bit γίνει 1 τότε ξεκινάει μετατροπή. Μόλις η μετατροπή ολοκληρωθεί το bit γίνεται 0.

ADIF: ADC Interrupt Flag

Γίνεται 1 όταν ολοκληρωθεί μια μετατροπή και ενημερωθούν οι καταχωρητές δεδομένων. Αν η τιμή του ADIE καθώς και του I-bit στον SREG είναι 1 τότε θα προκληθεί διακοπή ολοκλήρωσης μετατροπής του ADC.

ADIE: ADC Interrupt Enable

Αν η τιμή του ADIE καθώς και του I-bit στον SREG είναι 1 τότε θα προκληθεί διακοπή ολοκλήρωσης μετατροπής του ADC.

ADPS2:0: ADC Prescaler Select Bits

Τα bit αυτά θέτουν ένα παράγοντα διαίρεσης μεταξύ του κεντρικού ρολογιού και της συχνότητας λειτουργίας του ADC όπως φαίνεται στον επόμενο πίνακα:

Table 85. ADC Prescaler Selections

| ADPS2 | ADPS1 | ADPS0 | Division Factor |
|-------|-------|-------|-----------------|
| 0 | 0 | 0 | 2 |
| 0 | 0 | 1 | 2 |
| 0 | 1 | 0 | 4 |
| 0 | 1 | 1 | 8 |
| 1 | 0 | 0 | 16 |
| 1 | 0 | 1 | 32 |
| 1 | 1 | 0 | 64 |
| 1 | 1 | 1 | 128 |

Καταχωρητές δεδομένων του ADC – ADCL και ADCH

| | | | | | | | | | |
|-----|------|------|------|------|------|------|------|------|------|
| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | |
| | – | – | – | – | – | – | ADC9 | ADC8 | ADCH |
| | ADC7 | ADC6 | ADC5 | ADC4 | ADC3 | ADC2 | ADC1 | ADC0 | ADCL |
| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |

Τα δεδομένα μόλις ολοκληρωθεί μια μετατροπή αποθηκεύονται σε αυτούς τους δυο καταχωρητές. Πρώτα πρέπει να διαβαστεί ο ADCL και μετά ο ADCH.

Ρουτίνα αρχικοποίησης του ADC:

```
; Routine: usart_init
; Description:
; This routine initializes the
; ADC as shown below.
; ----- INITIALIZATIONS -----
;
; Vref: Vcc (5V for easyAVR6)
; Selected pin is A0
; ADC Interrupts are Enabled
; Prescaler is set as CK/128 = 62.5kHz
; -----
; parameters: None.
; return value: None.
; registers affected: r24
; routines called: None
ADC_init:
```

```

ldi r24, (1<<REFS0) ; Vref: Vcc
out ADMUX, r24      ; MUX4:0 = 00000 for A0.
; ADC is Enabled (ADEN=1)
; ADC Interrupts are Enabled (ADIE=1)
; Set Prescaler CK/128 = 62.5Khz (ADPS2:0=111)
ldi r24, (1<<ADEN) | (1<<ADIE) | (1<<ADPS2) | (1<<ADPS1) | (1<<ADPS0)
out ADCSRA, r24
ret

```

Στην αναπτυξιακή πλακέτα easyAVR6 με την χρήση jumper στις υποδοχές που φαίνονται στην εικόνα μπορείτε να συνδέσετε οποιοδήποτε από τα A0-A7 με ένα ποτενσιόμετρο που λειτουργεί σαν διαιρέτης της τάσης των 5V. Αυτό σας δίνει την δυνατότητα να ελέγξετε τη λειτουργία του προγράμματος σας με διαφορετικές τάσεις στο εύρος από 0 έως 5V. Σε περίπτωση που σας χρειαστεί η τάση δίνεται από τον τύπο:

$$V_{IN} = \frac{ADC}{1024} V_{REF} \text{ όπου}$$

V_{IN} η τάση στο pin A0

ADC η τιμή που διαβάζεται από τον ADC (αριθμός 10bit από 0-1023)

V_{REF} η τάση αναφοράς που με την δεδομένη ρουτίνα αρχικοποίησης έχει οριστεί σαν $V_{CC}=5V$.



Το ποτενσιόμετρο του συστήματος EasyAvr6

Στο πλαίσιο της άσκησης θα μελετηθούν δύο τρόποι χειρισμού του ADC. Ο πρώτος τρόπος θα είναι με την χρήση της διακοπής ολοκλήρωσης της μετατροπής του ADC. Η διακοπή αυτή μεταφέρει τον έλεγχο στην διεύθυνση **0x1C**, αν είναι ενεργοποιημένη η αντίστοιχη διακοπή (από το bit ADIE του ADCSRA) καθώς και οι γενικές διακοπές. Για να ξεκινήσει μια μετατροπή αρκεί να γραφεί 1 στο bit ADSC του καταχωρητή ADCSRA. Ο δεύτερος τρόπος είναι το πρόγραμμα να αναμένει να ολοκληρωθεί η μετατροπή (polling). Η αναμονή αυτή γίνεται ελέγχοντας το bit ADSC του ADCSRA το οποίο γίνεται 0 μόλις ολοκληρωθεί η μετατροπή.

Τα ζητούμενα της 4ης εργαστηριακής άσκησης

Καλείστε να τροποποιήσετε την Άσκηση 3 έτσι ώστε το σύστημα ηλεκτρονικής κλειδαριάς που είχατε σχεδιάσει να προορίζεται για την είσοδο σε έναν χώρο όπου υπάρχει αυξημένος κίνδυνος ύπαρξης μονοξειδίου του άνθρακα (CO):

- 1) Ο αισθητήρας CO είναι συνδεδεμένος στην θύρα A0. Καθ' όλη την διάρκεια πρέπει να διαβάζεται η τιμή του αισθητήρα ανά 100 ms (μικρές αποκλίσεις είναι αποδεκτές) και να εμφανίζεται μια ένδειξη του επιπέδου του αερίου στα LED PB0-PB6. Αν οποιαδήποτε στιγμή η συγκέντρωση του CO ξεπεράσει τα 70ppm να τυπώνεται στην LCD το μήνυμα GAS DETECTED και να αναβοσβήνουν τα αντίστοιχα LED στα PB0-PB6 αναλόγως το επίπεδο του αερίου. Το επίπεδο θα πρέπει να συνεχίζει να διαβάζεται (και να εμφανίζεται στα LED) και αν επανέλθει σε επίπεδο χαμηλότερο των 70ppm τα LEDs να σταματούν να αναβοσβήνουν και να τυπώνεται στην LCD το μήνυμα CLEAR.
- 2) Αν πατηθεί ο κωδικός της ομάδας σας στο πληκτρολόγιο να τυπώνεται στην LCD το μήνυμα WELCOME και ανάβει το PB7 σταθερά για 4 δευτερόλεπτα. Αν ο συναγερμός ήταν ανοιχτός να κλείνει για τα 4 αυτά δευτερόλεπτα (προκειμένου η ειδική ομάδα που εισήλθε να φτιάξει το πρόβλημα). Έπειτα να σβήνει το PB7 και το WELCOME και να συνεχίζει η λειτουργία του ερωτήματος 1. Σε περίπτωση λάθος κωδικού να αναβοσβήνει το PB7 (χρόνος ~0.5 sec αναμμένο και ~0.5 sec σβησμένο για συνολικά 4 δευτερόλεπτα) μαζί με την κανονική λειτουργία τους ερωτήματος 1.

Το σύστημα να υλοποιηθεί σε assembly και σε C με τις εξής διαφοροποιήσεις:

Ζήτημα 5.1: Υλοποίηση σε Assembly. Στην assembly η ανάγνωση των δεδομένων του ADC πρέπει να γίνεται μέσα στην ρουτίνα εξυπηρέτησης της διακοπής ολοκλήρωσης μετατροπής του ADC.

Ζήτημα 5.2: Υλοποίηση σε C. Στην C επιλέξτε εσείς εάν θα χρησιμοποιήσετε την μέθοδο polling η την μέθοδο της διακοπής προκειμένου να διαβάσετε τον ADC. Η υλοποίηση της LCD σε C είναι προαιρετική (απλά θα παραληφθούν τα μηνύματα στην LCD).

Είναι καλή πρακτική να διατηρείτε τις ρουτίνες εξυπηρέτησης διακοπών όσο το δυνατόν μικρότερες σε χρονική διάρκεια. Παρόλα αυτά αν το πρόγραμμα σας είναι λειτουργικό δεν θα υπάρξει αρνητική επίπτωση στην βαθμολογία. Για την μέτρηση της συγκέντρωσης του CO θεωρείστε ότι χρησιμοποιείται αισθητήρας του τύπου ULPSM-CO-968-001 (<http://mycourses.ntua.gr/courses/ECE1175/document/ULPSM-CO-968-001.pdf>) ο οποίος βρίσκεται σε σταθερές συνθήκες θερμοκρασίας (20°C) και υγρασίας (40%). Δίνεται ότι $V_{gas0}=0.1V$ και **sensitivity code 129nA/ppm**. Για να προσομοιώσετε την έξοδο που θα είχε ο αισθητήρας CO μπορείτε να χρησιμοποιήσετε το slider για τον ADC (δεξιά πάνω στην πλακέτα) το οποίο παράγει μια σταθερή τάση στο εύρος 0-4V, έχει δηλαδή την ίδια λειτουργία με ένα ποτενσιόμετρο.

Περιορισμοί προγράμματος απομακρυσμένης πρόσβασης:

- 1) Για να κάνετε reset στον ATMEGA16 χρησιμοποιείτε το push button RS που βρίσκεται αριστερά του PD7. Το Reset που βρίσκεται δίπλα από το Disconnect κάνει reset σε όλο το σύστημα σε περίπτωση που έχει κολλήσει.
- 2) Για να κάνετε Upload ένα αρχείο πρέπει να έχετε πατήσει πρώτα το connect και να φαίνεται ότι είναι Connected.
- 3) Για να ενεργοποιήσετε την ανάγνωση ενός LED πρέπει να κάνετε δεξί κλικ στο συγκεκριμένο LED. Για να την απενεργοποιήσετε κάνετε αριστερό κλικ.
- 4) Τα Push Buttons λειτουργούν με αριστερό κλικ (πατημένο = Λογικό 1) και αν θέλετε μπορείτε να κάνετε δεξί κλικ ώστε να μείνει πατημένο το συγκεκριμένο Push Button. Επανέρχεται με αριστερό κλικ.
- 5) Μπορείτε να χρησιμοποιήσετε τα LED της PORTB αλλά κανένα από τα Push Button ή μπορείτε να έχετε μονάχα 1 Push Button ενεργοποιημένο κάθε χρονική στιγμή και κανένα από τα LED.
- 6) Δεν μπορείτε για οποιαδήποτε PORT να χρησιμοποιείται και το LED και το Push Button.

- 7) Δεν μπορείτε να χρησιμοποιήσετε το keypad αν δεν το διαβάξει ο ATMEGA16 με την ρουτίνα `scan_keypad_rising_edge_sim` αλλιώς το κουμπί που πατήσατε θα παραμείνει κόκκινο και θα πρέπει να κάνετε reset στο σύστημα. Για να ενεργοποιήσετε την λειτουργία του πληκτρολογίου χρησιμοποιήστε τον διακόπτη ακριβώς από πάνω (1=ενεργοποιημένο)
- 8) Για να ενεργοποιήσετε την αποκωδικοποίηση των σημάτων της PORTD σαν να είναι συνδεδεμένη η οθόνη LCD χρησιμοποιήστε τον διακόπτη πάνω από την LCD (1=ενεργοποιημένη). Για να λειτουργήσει σωστά σιγουρευτείτε ότι ο ATMEGA16 έχει εκτελέσει την ρουτίνα `lcd_init` ενώ η LCD είναι ενεργοποιημένη.
- 9) Η τιμή του slider για τον ADC (δεξιά πάνω στην πλακέτα) αποστέλλεται όταν αφήσετε το πλήκτρο του ποντικιού. Η τιμή δεν αντιπροσωπεύει κάτι.