

Εργαστηριακή Άσκηση 3

MICROCHIP STUDIO

Διάρθρωση Παρουσίασης

- ▶ Σκοπός
- ▶ Περιγραφή και υλοποίηση παραδείγματος με Analog to Digital Converter (ADC)
- ▶ Επεξήγηση εργαστηριακής άσκησης

ΣΚΟΠΟΣ

- Ο σκοπός της άσκησης αυτής είναι η εξοικείωσή σας με:
- ▶ Εξοικείωση με τον Analog to Digital Converter (ADC).

Παράδειγμα

ΛΕΙΤΟΥΡΓΙΑ ΤΟΥ ADC

Περιγραφή Παραδείγματος

- ▶ Στο παράδειγμα αυτό θέλουμε να ενεργοποιήσουμε τον ADC να δέχεται από το PIN7 του PORTD μετρήσεις, οι οποίες αν είναι κάτω από μία τιμή (για παράδειγμα 10) θα ενεργοποιούν ένα interrupt.
- ▶ Στο interrupt θέλουμε να ανάψουμε ένα LED για 5ms και μετά να επιστρέψουμε στην αρχική μας ροή και να περιμένουμε ξανά την μέτρηση του ADC.

Activate ADC

- ▶ Πρώτα επιλέγουμε αν θέλουμε 10-bit ή 8-bit resolution → Resolution Selection bit (RESSEL) in Control A register (ADCn.CTRLA)
- ▶ Έπειτα, επιλέγουμε το Free-Running mode → Γράφουμε '1' στο Free-Running bit (FREERUN) in ADCn.CTRLA
- ▶ Επιλέγουμε με ποιο bit θα συνδεθεί ο ADC → MUXPOS bit field in MUXPOS register (ADCn.MUXPOS)
- ▶ Κάνουμε enable τον ADC → Γράφουμε '1' στο ENABLE bit in ADCn.CTRLA
- ▶ Μας δίνεται η επιλογή να ενεργοποιήσουμε το Debug Mode ώστε ο ADC να μην σταματά ποτέ να τρέχει και να λαμβάνει τιμές.

Window Comparator Mode

- ▶ Εισάγουμε το threshold στον καταχωρητή ADCn.WINLT και/ή ADCn.WINHT
- ▶ Κάνουμε enable τα interrupts → Window Comparator Interrupt Enable bit (WCOMP) in Interrupt Control register (ADCn.INTCTRL)
- ▶ Επιλέγουμε το Mode που θέλουμε (στην περίπτωσή μας θέλουμε interrupt όταν $RESULT < THRESHOLD$ → WINCM bit field in ADCn.CTRLA)

Εκκίνηση Conversion

- ▶ Για να ξεκινήσει την λειτουργία του ο ADC πρέπει να γράψουμε '1' στο Start Conversion bit (STCONV) in the Command register (ADCn.COMMAND)
- ▶ Οι τιμές καταγράφονται στον καταχωρητή RES (Result)

Συμβουλή: Ανατρέξτε στο ATmega4808 DataSheet και διαβάστε καλά τις σελίδες 394-421. Αναγράφονται αναλυτικά όλες οι πιθανές λειτουργίες που μπορείτε να χρησιμοποιήσετε για να κάνετε Analog to Digital Conversion. Στην εργαστηριακή άσκηση είστε ελεύθεροι να επιλέξετε όποιο mode σας βολεύει.

Υλοποίηση

```
#include <avr/io.h>
#include <util/delay.h>
#include <avr/interrupt.h>

int main(){
    PORTD.DIR |= PIN1_bm; //PIN is output
    //initialize the ADC for Free-Running mode
    ADC0.CTRLA |= ADC_RESSEL_10BIT_gc; //10-bit resolution
    ADC0.CTRLA |= ADC_FREERUN_bm; //Free-Running mode enabled
    ADC0.CTRLA |= ADC_ENABLE_bm; //Enable ADC
    ADC0.MUXPOS |= ADC_MUXPOS_AIN7_gc; //The bit
    //Enable Debug Mode
    ADC0.DBGCTRL |= ADC_DBGRUN_bm;
    //Window Comparator Mode
    ADC0.WINLT |= 10; //Set threshold
    ADC0.INTCTRL |= ADC_WCMP_bm; //Enable Interrupts for WCM
    ADC0.CTRLE |= ADC_WINCM0_bm; //Interrupt when RESULT < WINLT
    sei();
    ADC0.COMMAND |= ADC_STCONV_bm; //Start Conversion
    while(1){ }
}
```

```
ISR(ADC0_WCOMP_vect){
    int intflags = ADC0.INTFLAGS;
    ADC0.INTFLAGS = intflags;
    PORTD.OUTCLR= PIN1_bm; //LED is on
    _delay_ms(5);
    PORTD.OUT |= PIN1_bm; //LED is off
}
```

- ▶ Πάρτε τον κώδικα και κάντε Simulation στο Microchip Studio.
- ▶ Για το Simulation: Την τιμή που θα παράγει ο ADC μετρώντας την είσοδο που θέσαμε την γράφετε εσείς στον καταχωρητή RES (Result). Ανάλογα μετά εκτελεί τις λειτουργίες που έχουμε ορίσει.

Εργαστηριακή Άσκηση

ΕΞΥΠΝΗ ΟΙΚΙΑΚΗ ΣΥΣΚΕΥΗ ΠΟΥ ΚΙΝΕΙΤΑΙ ΣΤΟΝ ΧΩΡΟ

Περιγραφή

- ▶ Σε αυτή την εργαστηριακή άσκηση θα προσομοιώσουμε τη λειτουργία μιας έξυπνης οικιακής συσκευής που κινείται στον χώρο ενός άδειου δωματίου.
- ▶ Ξεκινάει από μία γωνία του δωματίου και ο σκοπός της είναι να σχεδιάσει το περίγραμμά του.
- ▶ Καθώς κινείται στον χώρο θα παίρνει τιμές από έναν αισθητήρα που μετράει την απόσταση της συσκευής από ένα εμπόδιο μπροστά της (εδώ θα προσομοιωθεί με ένα ποτενσιόμετρο που θα μετράτε από τον ADC).
- ▶ Αν η τιμή είναι κάτω του επιτρεπτού (ορίστε εσείς μια οποιαδήποτε τιμή μεταξύ 1-254) η συσκευή θα πρέπει να σταματήσει και να κινηθεί αριστερά.
- ▶ Η συσκευή θα επιλέγει να κινηθεί δεξιά αν ο δεξιός αισθητήρας μας δείχνει ότι δεν υπάρχει τοίχος στα δεξιά της συσκευής.
- ▶ Το δωμάτιο έχει συνολικά 8 γωνίες. Όταν φτάσει στην γωνία από την οποία ξεκίνησε θέλουμε να σταματήσει.
- ▶ Επίσης θέλουμε αν πατηθεί ένα switch να μπορεί να γυρίσει πίσω στην θέση της ακολουθώντας την πορεία που έχει κάνει μέχρι τώρα αλλά ανάποδα.

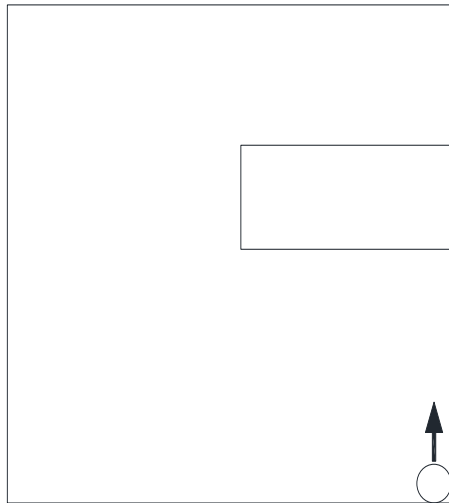
Παραδοχές

- ▶ Η κίνηση θα προσομοιωθεί με ένα LED (όταν κινείται ευθεία το LED είναι ανοιχτό αλλιώς κλείνει).
- ▶ Ο δεξιός αισθητήρας θα προσομοιωθεί με ένα switch στο οποίο θα επιτρέπεται το interrupt. Όταν πατηθεί το switch (υπάρχει '1' στο interrupt flag) θα ενεργοποιηθεί το interrupt και θα κινηθεί δεξιά.
- ▶ Η αριστερή και δεξιά κίνηση θα προσομοιωθούν με δύο διαφορετικά LED τα οποία είναι αναμμένα μέχρι ένας timer φτάσει μια συγκεκριμένη τιμή (βάλτε τιμές που σας βολεύουν και εξηγήστε πως βρήκατε τα values).
- ▶ Όταν πατηθεί ένα δεύτερο switch θα ενεργοποιείται η ανάποδη πορεία.

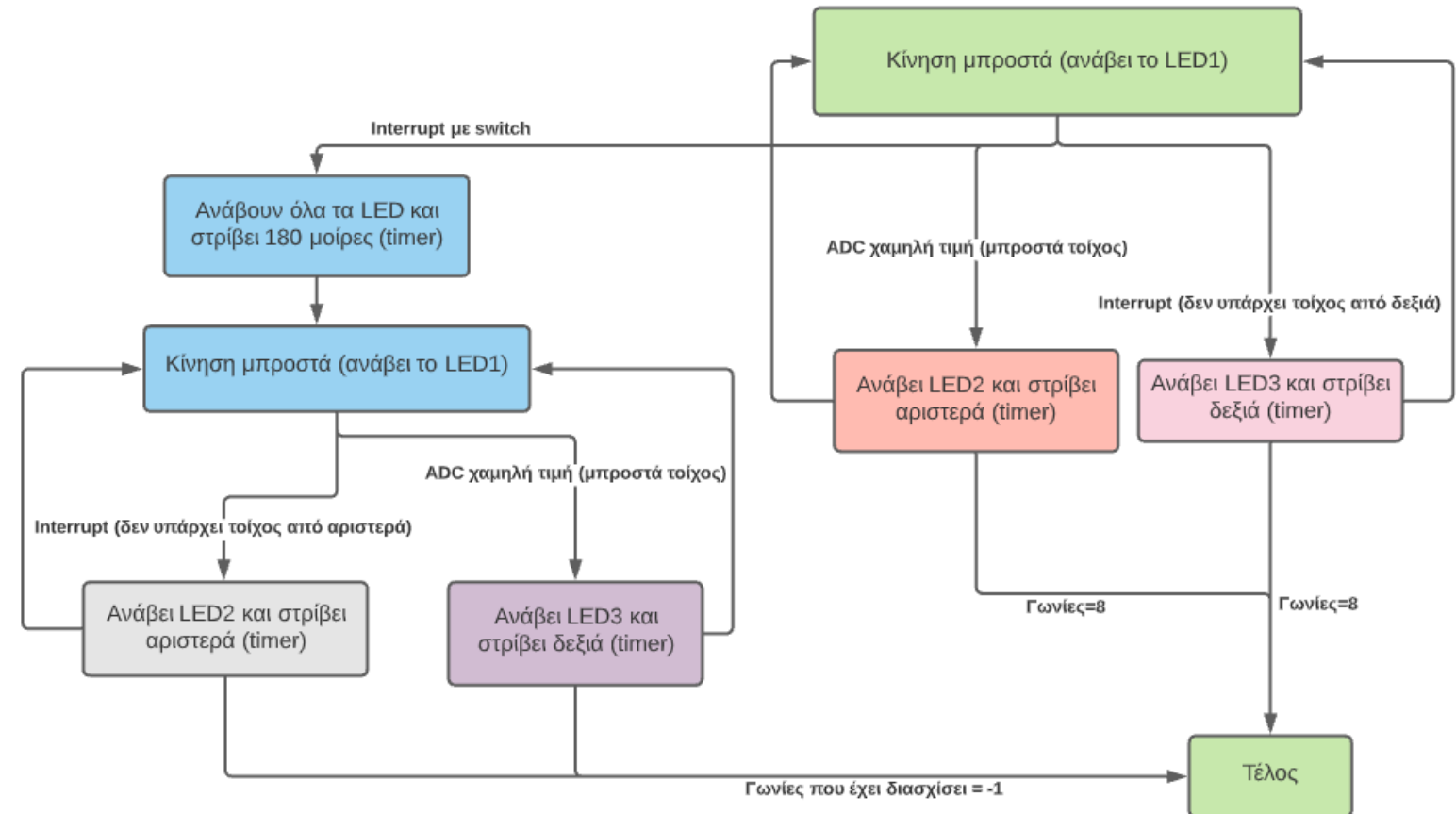
Ανάποδη πορεία

- ▶ Αρχικά, η συσκευή θέλουμε να γυρίσει 180 μοίρες. Αυτό θα προσομοιωθεί με τα τρία LEDs να είναι ταυτόχρονα ανοιχτά για ένα χρονικό διάστημα (χρήση timer).
- ▶ Έπειτα, θα εκτελεί την ίδια ακριβώς διαδικασία αλλά ανάποδα.
- ▶ Συγκεκριμένα
 - ▶ Καθώς προχωράει μπροστά και βρει τοίχο θέλουμε να στρίψει δεξιά (ανοιχτό το αντίστοιχο LED).
 - ▶ Όταν πατηθεί το switch που δηλώνει ότι δεν υπάρχει τοίχος πλέον από αριστερά, θέλουμε η συσκευή να στρίψει αριστερά (ανοιχτό το αντίστοιχο LED).
- ▶ Όταν καταλάβει ότι επέστρεψε στην αρχική της θέση, τερματίζει.

Διάγραμμα ροής και δωμάτιο



«Γωνίες που έχουν διασχιστεί
= -1»
Μετρήστε και την στροφή
των 180 ως μία κίνηση που
πρέπει να σημειωθεί.



Παραδοτέα

- ▶ Παραδίδεται αναλυτική αναφορά με τη λειτουργία του κώδικά σας.
- ▶ Επίσης, παραδίδεται τον κώδικά σας με αναλυτικά σχόλια.
- ▶ Bonus:
 - ▶ Μπορείτε να σκεφτείτε έναν τρόπο ώστε και χωρίς να ξέραμε τις γωνίες του δωματίου η συσκευή μας να καταλάβαινε πότε ολοκλήρωσε το περίγραμμα του δωματίου;
 - ▶ Κατά πόσο ο κώδικάς σας μπορεί να αξιοποιηθεί και σε τελείως άγνωστα δωμάτια; Δώστε κάποιο αντιπαράδειγμα.
 - ▶ Προσθέστε το στον κώδικα και εξηγήστε τον τρόπο σκέψη σας (Δεν χρειάζεται να αλλάξετε τον κώδικα της ανάποδης πορείας καθώς αυτό το ερώτημα αναφέρεται μόνο στην κανονική πορεία της συσκευής).