

# MICROCHIP STUDIO

AVR PROGRAMMING

Κυριακή Τσαντικίδου  
e-mail: k DOT tsantikidou AT upatras DOT gr

# Διάρθρωση Παρουσίασης

1. Μετατροπέας Αναλογικού σε Ψηφιακό (ADC)
2. Δεύτερη Εργαστηριακή Άσκηση
3. Βιβλιογραφία

# Μετατροπές Αναλογικού σε Ψηφιακό

Ενότητα 1

# Ορισμός και Τρόποι Λειτουργίας

- ▶ Ο Μετατροπέας Αναλογικού σε Ψηφιακό (Analog to Digital Converter – ADC) είναι ένα σύστημα το οποίο μετατρέπει αναλογικά σήματα που έρχονται ως είσοδοι στα κατάλληλα PINs, όπως ήχος, ρεύμα, φως, κτλ., σε ψηφιακά σήματα.
- ▶ Οι τιμές των αναλογικών σημάτων μπορούν να διαβαστούν από τον μικροελεγκτή και να αξιοποιηθούν κατάλληλα.
- ▶ Οι βασικοί τρόποι λειτουργίας του ADC είναι δύο:
  - ▶ Το Free-Running Mode στο οποίο ο ADC συνεχώς δέχεται τιμές και τις μετατρέπει σε ψηφιακά σήματα.
  - ▶ Η προκαθορισμένη λειτουργία του, το Single Conversion Mode, που εκτελεί μόνο μία μετατροπή όποτε ζητηθεί στον κώδικα και μετά σταματά.

# Ενεργοποίηση Μετατροπέα

- ▶ Πρώτα το resolution ορίζεται σε 10-bit ή 8-bit → Resolution Selection bit (RESSEL) στον Control A register (ADCn.CTRLA).
- ▶ Έπειτα, το Free-Running Mode ενεργοποιείται → '1' στο Free-Running bit (FREERUN) στο ADCn.CTRLA. Στην περίπτωση που ζητείται να εκτελεστεί η προκαθορισμένη λειτουργία του ADC (single-conversion mode), δηλαδή να εκτελεστεί μία μόνο μετατροπή όποτε ζητηθεί, αυτό το ψηφίο (bit) δεν ενεργοποιείται και παραμένει '0'.
- ▶ Ορίζεται το ψηφίο (bit) με το οποίο θα συνδεθεί ο ADC → MUXPOS bit field στον MUXPOS register (ADCn.MUXPOS).
- ▶ Ο ADC ενεργοποιείται → Γράφουμε '1' στο ENABLE bit στο ADCn.CTRLA.
- ▶ Δίνεται η επιλογή της ενεργοποίησης του Debug Mode ώστε ο ADC να μην σταματά ποτέ να τρέχει και να λαμβάνει τιμές.

# Λειτουργία Σύγκρισης (Window Comparator Mode)

- ▶ Πρώτα εισάγεται το κατώφλι (threshold) στον καταχωρητή ADCn.WINLT και/ή ADCn.WINHT.
- ▶ Ενεργοποιείται η λειτουργία δημιουργίας διακοπών (interrupts) → Window Comparator Interrupt Enable bit (WCOMP) στον Interrupt Control register (ADCn.INTCTRL).
- ▶ Ορίζεται το επιθυμητό Mode (στην περίπτωση μας θέλουμε διακοπές (interrupts), όταν  $RESULT < THRESHOLD$ ) → WINCM bit field στον ADCn.CTRL.

# Εκκίνηση Μετατροπής (Conversion)

- ▶ Για την εκκίνηση της λειτουργίας του ADC, πρέπει να γραφτεί '1' στο Start Conversion bit (STCONV) στον 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.

# Λειτουργία Παραδείγματος

- ▶ Στο παράδειγμα, όταν ο ADC θα διαβάσει μετρήσεις που είναι κάτω από μία τιμή (για παράδειγμα  $RES=10$ ) θα ενεργοποιείται μια διακοπή (interrupt).
- ▶ Στη διακοπή (interrupt) θα ανάβει ένα LED για  $T=5ms$  και μετά θα επιστρέφουμε στην αρχική ροή και θα περιμένουμε ξανά την μέτρηση του ADC.



# Υλοποίηση Τέταρτου Παραδείγματος: ADC

```
#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.

# Δεύτερη Εργαστηριακή Άσκηση

Ενότητα 2

# Έξυπνη Οικιακή Συσκευή, με Κίνηση στο Χώρο

- ▶ Σκοπός αυτής της εργαστηριακής άσκησης είναι η προσομοίωση της λειτουργίας 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:
  - ▶ Αν η τιμή είναι κάτω του επιτρεπτού, η συσκευή θα πρέπει να σταματήσει και να κινηθεί αριστερά.
  - ▶ Η συσκευή θα επιλέγει να κινηθεί δεξιά αν ο δεξιός αισθητήρας δείχνει ότι δεν υπάρχει τοίχος στα δεξιά της συσκευής, δηλαδή η τιμή είναι πάνω του επιτρεπτού.
- ▶ Όταν φτάσει στην γωνία από την οποία ξεκίνησε, η συσκευή πρέπει να σταματήσει.
- ▶ Αν πατηθεί ένας διακόπτης (switch), η συσκευή πρέπει να εκτελέσει στροφή 180 μοιρών και να γυρίσει πίσω στην θέση της, ακολουθώντας την πορεία που έχει κάνει μέχρι τώρα.

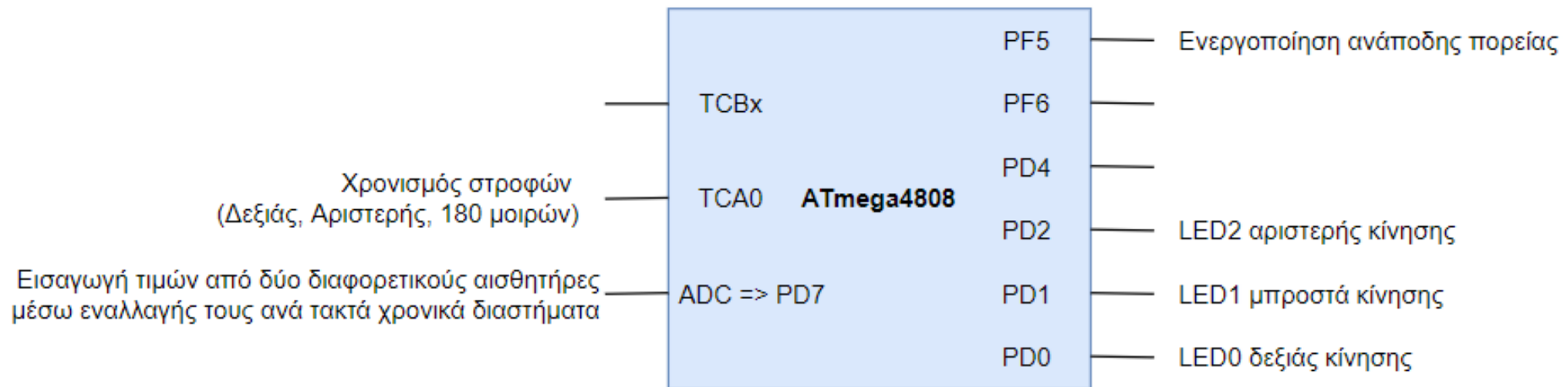
# Παραδοχές (1/2)

- ▶ Η κίνηση θα προσομοιωθεί με ένα LED (όταν κινείται ευθεία το LED1 είναι ανοιχτό αλλιώς κλείνει).
- ▶ Υπάρχει μόνο ένας ADC διαθέσιμος στην πλακέτα, επομένως για να χρησιμοποιηθεί σε δύο διαφορετικές λειτουργίες θα πρέπει να δέχεται τιμές από κάθε αισθητήρα εναλλάξ ανά τακτά χρονικά διαστήματα.
  - ▶ Αρχικά θα δέχεται συνεχόμενα τιμές από τον πλαϊνό αισθητήρα (Free-Running Mode), και μόλις περάσουν  $T1=3s$  (χρήση χρονιστή/timer) τότε θα δεχτεί μία και μόνο τιμή από τον μπροστά αισθητήρα (Single Conversion).
  - ▶ Μόλις περάσουν  $T2= 2s$  (χρήση χρονιστή/timer), ο ADC θα επιστρέψει στην πρώτη του λειτουργία (Free-Running Mode).
  - ▶ Αυτές οι διαδικασίες θα πρέπει συνεχώς να εκτελούνται μέχρι την ολοκλήρωση του περιγράμματος του δωματίου.
- ▶ Η δεξιά και αριστερή στροφή θα προσομοιωθούν με δύο διαφορετικά LED (LED0 και LED2 αντίστοιχα) τα οποία είναι αναμμένα μέχρι ένας χρονιστής/μετρητής φτάσει μια συγκεκριμένη τιμή.

## Παραδοχές (2/2)

- ▶ Όταν πατηθεί ένας διακόπτης (Switch 5) θα ενεργοποιείται η ανάποδη πορεία.
  - ▶ Αυτό θα προσομοιωθεί με τα τρία LEDs (LED 0, 1, 2) ταυτόχρονα ενεργοποιημένα για ένα χρονικό διάστημα (χρήση χρονιστή/timer).
  - ▶ Έπειτα, θα εκτελεί την ίδια ακριβώς διαδικασία αλλά ανάποδα.
  - ▶ Καθώς προχωράει μπροστά, όταν ADC δεχθεί τιμή μικρότερη του κατωφλίου, η συσκευή θα στρίψει δεξιά (LED0). Ο ADC συνεχίζει να δέχεται τιμές από τον μπροστά αισθητήρα με λειτουργία Single Conversion.
  - ▶ Όταν καταλάβει ότι δεν υπάρχει τοίχος αριστερά, δηλαδή ο ADC δεχτεί τιμή μεγαλύτερου του κατωφλίου που έχει οριστεί από τον αριστερό αισθητήρα, η συσκευή κινείται αριστερά (LED2). Έπομένως, για την αριστερή στροφή αυτή τη φορά ο ADC θα βρίσκεται σε λειτουργία Free-Running Mode.
  - ▶ Τέλος, όταν καταλάβει ότι επέστρεψε στην αρχική της θέση, τερματίζει.

# Απεικόνιση ATmega 4808



# Ερωτήματα

- ▶ **Ερώτημα 1:** Υλοποιήστε τον κώδικα της κίνησης της οικιακής συσκευής, όταν το δωμάτιο είναι τετράγωνο (επομένως δεν χρειάζεται ο δεξιός αισθητήρας και η δεύτερη λειτουργία του ADC). Ωστόσο, οι λειτουργίες με τους χρονισμούς θα χρειαστούν για την προσομοίωση των στροφών.
- ▶ **Ερώτημα 2:** Υλοποιήστε τον κώδικα της κίνησης για ένα τυχαίο δωμάτιο που περιέχει και δύο αμβλείες γωνίες, εδώ εισάγεται και η δεύτερη λειτουργία του ADC.
- ▶ **Ερώτημα 3:** Υλοποιήστε την ανάποδη λειτουργία.

# Βιβλιογραφία



# Βιβλιογραφία

- ▶ Microchip, Επίσημη ιστοσελίδα: <https://www.microchip.com/mplab/microchip-studio>
- ▶ Microchip, “AVR-IoT Wx Hardware User Guide”:  
<https://ww1.microchip.com/downloads/en/DeviceDoc/AVR-IoT-Wx-Hardware-User-Guide-DS50002805C.pdf>
- ▶ Microchip, “ATmega4808/4809 Data Sheet”:  
<https://ww1.microchip.com/downloads/en/DeviceDoc/ATmega4808-09-DataSheet-DS40002173B.pdf>
- ▶ Microchip, “Advanced Software Framework (ASF)”:  
<https://www.microchip.com/mplab/avr-support/advanced-software-framework>
- ▶ Microchip, “MPLAB® XC Compilers”:  
<https://www.microchip.com/en-us/development-tools-tools-and-software/mplab-xc-compilers>