

ΕΡΓΑΣΤΗΡΙΟ ΠΡΟΗΓΜΕΝΩΝ ΜΙΚΡΟΕΠΕΞΕΡΓΑΣΤΩΝ

2Η ΑΣΚΗΣΗ

07/04/2022

ΧΑΡΑΛΑΜΠΟΣ ΠΑΠΠΑΣ - 1053359

ΓΕΩΡΓΙΟΣ ΚΟΝΤΟΓΙΑΝΝΗΣ - 1070908



* Για το 1ο ερωτημα της ασκησης, προκειμενου να προσομοιωθει η κινηση του τετραγωνου, θετουμε το PIN2 ανεμενο για την ευθεια κινηση και τον PIN0 για την αριστερη στροφη. Ολες οι αλλες περιπτωσεις στο συγκεκριμεο παραδειγμα δεν μας αποσχολουν καθως η συσκευη στριβει μονο αριστερα.
* Με τη μεταβλητη “turns”, κραταμε τον αριθμο στροφων που εχει κανει, και μολις αυτο φτασει στο τεσσερα (4 στροφες κατα μεγιστο μπορει να κανει), θα ειναι ουσιαστικα στην θεση εκκινησης, οποτε και το προγραμμα σταματαει.
* Καθε φορα που πυροδοτειται το interrupt του ADC, το LED της ευθειας σβηνει, αναβει το LED της στροφης. Η στροφη αυτη προσομοιωνειται με μια μικρη καθυστερηση. Στη συνεχεια, αναβει ξανα το LED της ευθειας.Τελος, η μεταβλητη turns μειωνεται κατα 1.









* Η λογικη του προγραμματος ειναι ιδια με πριν, με τη διαφορα οτι τα LED0, LED1 υποδηλωνουν την αριστερα και δεξια στροφη αντιστοιχα και το LED2 την ευθεια πορεια. Επισης, εχουν δημιουργηθει δυο συναρτησεις “turnLeft” & “turnRight”, οι οποιες προσομοιωνουν τις στροφες, αναβοσβηνοντας τα αντιστοιχα LED καθε φορα με τη βοήθεια ενός timer, οπως στην ασκηση 1. Επίσης, η μεταβλητή turns τώρα ονομάζεται cnt.
* Έχει οριστεί ένα άνω και κάτω όριο (WINLT=5, WINHT=20) που υποδηλώνουν την ενεργοποίση ανίχνευσης μπροστινού και δεξιού εμποδίου, αντίστοιχα. Δηλαδή, εισάγωντας στον καταχωρητή RES τιμή μικρότερη ίση του 5, η συσκευή ανιχνεύει σε free-running mode μπροστινά εμπόδια, και για RES μεγαλύτερο ίσο του 20, ελέγχθει μια φορά δεξιά με τη βοήθεια της συνάρτησης single(). Η συνάρτηση single() αλλάζει τον ADC0 σε mode 0, δειγματοληπτεί την απόσταση μέσω του RES και εάν ξεπερνάει κάποιο όριο ενεργοποιέιται το ISR(ADC0\_WCOMP\_vect) το οποίο αποφασίζει αν θα στρίψει δεξιά ή όχι. Αν ναί, καλείται η συνάρτηση turnRight(), γίνονται οι αντίστοιχες ενέργειες, μηδενίζει τον RES, ελέγχει άν έχει διανύσει και τις επτά (7) πλευρές (ελέγχει αν cnt >= 7, αν ναί κάνει το flag still=0) και ξανα επιστρέφει στο free-running mode. Επιστρέφει στη main και ελέγχει το flag still. Άν είναι «1», η παραπάνω διαδικασία συνεχίζεται. Στη περίπτωση όπου το RES <= WINLT πυροδοτείται το ISR(ADC0\_WCOMP\_vect) εκτελείται η αριστερά στροφή με τη βοήθεια της turnLeft(), ελέγχει το still και η διαδικασία επαναλαμβάνεται. Τελος, οταν το cnt φτασει τη τιμη ‘7’, θετουμε τη μεταβλητη still = 0 , διακοπτεται το προγραμμα και η συσκευη βρισκεται στην αρχικη της θεση.
* Η μεταβλητή reverse=0 μας ενημερώνει ότι η συσκεύη χαράζει κανονική πορεία και όλες οι παραπάνω λειτουργίες εκτελούνται όπως αναλύθηκαν υπό τη συνθήκη οτι if(!reverse)=true. Η λειτουργια της επαναφορας θεσης, εκτελειται με την πυροδότηση του ISR(PORTF\_PORT\_vect) μετά την ενεργοποιηση του switch στο PIN 5, θέτοντας το reverse=1, ενημερώνοντας έτσι το σύστημα if(!reverse)=false ότι θέλουμε αντίστροφη πορεία. Με το τρόπο αυτό, στη πυροδότηση του ISR(ADC0\_WCOMP\_vect) εκτελείται κάθε φορά ένα ξεχωριστό τμήμα κώδικα το οποίο κάνει τις καθρεπτικές λειτουργίες απο πρίν, βασιζόμενο στη μέχρι τότε τιμή του cnt, μέχρις ώτου cnt=0.