

ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

ΡΟΜΠΟΤΙΚΗ 1: ΑΝΑΛΥΣΗ ΕΛΕΓΧΟΣ ΕΡΓΑΣΤΗΡΙΟ

[ΡΟΗ Σ] 7ο ΕΞΑΜΗΝΟ 2019-2020

ΟΜΑΔΑ ΕΡΓΑΣΙΑΣ:

Γ2

ΠΡΑΓΜΑΤΟΠΟΙΗΣΗ ΕΡΓΑΣΤΗΡΙΟΥ: Τετάρτη 06/01 16.00-17.30

Ασημάκη Γεωργία-Γρηγορία,
Γιαννιός Γεώργιος-Ταξιάρχης,
Κρασιάς Δημήτριος,
Ντόκου Μυρσίνη,
Χατζηαντωνίου Παυλίνα,
Σαλπέα Ναταλία,

A.M.:03116197
A.M.:03116156
A.M.:03116030
A.M.:03116179
A.M.:03116186
A.M.:03116083

1η Εργαστηριακή Άσκηση Ρομποτικό Κύτταρο

Στόχος Εργαστηρίου-Περίληψη

Στόχο της 1ης Εργαστηριακής Άσκησης αποτελούσε η ταξινόμηση αντικειμένων με βάση χαρακτηριστικές ιδιότητες, με απώτερο σκοπό την εναπόθεσή τους σε διαφορετικά σημεία του workspace. Η ταξινόμηση αυτή έγινε με την βοήθεια τριών αισθητήρων (Επαγωγικός, Οπτικός, Χωρητικός) ενώ η εναπόθεση με τη χρήση ενός ρομποτικού βραχίονα.

Θεωρητικό Μέρος

A. Αισθητήρες

Ενδιαφέρον παρουσιάζει η λειτουργία των αισθητήρων που χρησιμοποιήσαμε. Ο **Επαγωγικός αισθητήρας** λοιπόν παρουσίαζε δύο καταστάσεις (Μπροστά από την ενεργό επιφάνεια 1. υπάρχει μεταλλικό αντικείμενο 2. δεν υπάρχει ή υπάρχει και δεν είναι μεταλλικό). Όταν τροφοδοτηθεί με ρεύμα ο επαγωγικός αισθητήρας, τότε δημιουργείται εναλλασσόμενο μαγνητικό πεδίο, μπροστά από την ενεργό επιφάνειά του. Αν εισέλθει μέσα στο μαγνητικό αυτό πεδίο, κάποιο μεταλλικό αντικείμενο (σίδηρο, αλουμίνιο, χαλκός κτλ), τότε προκαλείται ισχυρή απόσβεση της ταλάντωσης, αντιστρέφεται η έξοδος του κυκλώματος σκανδάλης και αλλάζει η κατάσταση της εξόδου του αισθητήρα.

Ο **Οπτικός αισθητήρας** από την άλλη εντόπιζε αν υπάρχει αντικείμενο σε εύρος 10cm. Αποτελούνταν από δύο μέρη. Πομπός ο οποίος εξέπεμπε Η/Μ κύμα και ο δέκτης ο οποίος με τη βοήθεια ευαίσθητων στοιχείων, λάμβανε το κύμα. Αν στον δέκτη έφτανε πίσω το κύμα, σημαίνει ότι πραγματοποιήθηκε ανάκλαση, άρα υπήρχε αντικείμενο.

Ο **Χωρητικός αισθητήρας** εντόπιζε από αν το αντικείμενο ήταν φορτισμένο. Ο βραχίονας για να διαπιστώσει αν το αντικείμενο είναι φορτισμένο, το τοποθετούσε πρώτα από την μια μεριά και αν δεν ανίχνευε φόρτιση το ανέστρεφε ώστε να αλλάξει η πολικότητα. Ο χωρητικός αισθητήρας προσέγγισης (Capacitive transducers), βασίζει την αρχή λειτουργίας του, στην αλλαγή της χωρητικότητας πυκνωτή σε ένα κύκλωμα ταλάντωσης RC, όταν πλησιάσει την ενεργό επιφάνεια του αισθητήρα ένα οποιοδήποτε αντικείμενο.

Μπροστά από την ενεργό επιφάνεια του ταλαντωτή δημιουργείται ένα ανομοιογενές ηλεκτρικό πεδίο. Όταν εισέλθει στο χώρο του ηλεκτρικού πεδίου κάποιο αντικείμενο, μεταβάλλεται η χωρητικότητα του ταλαντωτή και παράγεται ημιτονοειδές σήμα υψηλής συχνότητας. Το σήμα αυτό εισέρχεται στην επόμενη βαθμίδα, όπου απομονώνεται και ανορθώνεται. Το ανορθωμένο σήμα διεγείρει το κύκλωμα σκανδαλισμού (Schmitt - Triggering) το οποίο στη συνέχεια ενεργοποιεί το κύκλωμα ενίσχυσης.

B. Συστήματα Συντεταγμένων

Με τη βοήθεια όλων αυτών των αισθητήρων πραγματοποιήθηκε η ταξινόμηση των αντικειμένων.

Η τοποθέτηση πραγματοποιήθηκε με τη βοήθεια του βραχίονα, ο οποίος λάμβανε εντολές από τον Η/Υ. Για την επίτευξη της ζητούμενης θέσης, μπορούσαν να χρησιμοποιηθούν τέσσερα συστήματα:

Παγκόσμιο: Πρόκειται για σύστημα μέσω του οποίου στέλνουμε διαφορικές εντολές μετατόπισης του τελικού στοιχείου στους 3 άξονες

Αξονικό: Με τη βοήθεια αυτού του συστήματος στέλνουμε εντολές στο χώρο των αρθρώσεων

Του Εργαλείου: Είναι ίδιο σχεδόν με το Παγκόσμιο Σύστημα, με τη μόνη διαφορά ότι δεν βρίσκεται κάπου σταθερό στον χώρο, αλλά εφαπτόμενο στο τελικό στοιχείο δράσης.

Τα δυο πρώτα ήταν χρήσιμα όταν επιθυμούσαμε μεγάλες μετατοπίσεις στο χώρο, ενώ το άλλο όταν έπρεπε να γίνουν μικρές ρυθμίσεις. Να σημειωθεί ότι μέσω του Tool Orientation κάναμε μεταβολή σε προσανατολισμό (όχι μετατόπιση).

Γ. Κίνηση μεταξύ θέσεων

Η κίνηση μεταξύ δύο θέσεων μπορεί να γίνει μέσω:

Ark: Πηγαίνουμε από μια θέση (αρχική) σε μια θέση (τελική) ακολουθώντας τροχιά τόξου

Line: Πηγαίνουμε από μια θέση σε μία άλλη μέσω του ευθύγραμμου τμήματος που ενώνει τις θέσεις. Το ευθύγραμμο λοιπόν τμήμα χωρίζεται σε στόχους και πηγαίνουμε από στόχο σε στόχο (κάνοντας ανάστροφη κινηματική).

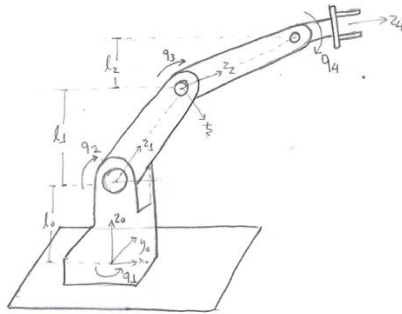
- Μειονέκτημα: Υψηλή Υπολογιστική Πολυπλοκότητα
- Πλεονέκτημα: Ξέρω αν υπάρχει εμπόδιο

point-to-point: Έχω μια αρχική και μια τελική θέση και λύνω μόνο για αυτές τις δύο το ανάστροφο

1. Μειονέκτημα: Δεν γνωρίζω την τροχιά που ακολουθεί ο end-effector
2. Πλεονέκτημα: Χαμηλή Υπολογιστική Πολυπλοκότητα

Κινηματική Ανάλυση

Εικόνα του βραχίονα με τοποθετημένα πλαίσια:



Ο πίνακας Denavit-Hartenberg προκύπτει από το παραπάνω σχήμα ως εξής:

Σύνδεσμος i	θ_i	d_i	α_i	a_i
1	q_1	l_0	-90	0
2	$q_2 - 90$	0	0	l_1
3	q_3	0	0	l_2
4	q_4	0	0	0

Υπολογίζουμε τους παρακάτω διαδοχικούς μετασχηματισμούς συντεταγμένων με βάση τη μήτρα D-H

$$A_1^0 = \begin{bmatrix} c1 & 0 & -s1 & 0 \\ s1 & 0 & c1 & 0 \\ 0 & -1 & 0 & l0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_2^1 = \begin{bmatrix} s2 & c2 & 0 & s2l1 \\ -c2 & s2 & 0 & -c2l1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_3^2 = \begin{bmatrix} c3 & -s3 & 0 & c3l2 \\ s3 & c3 & 0 & s3l2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_4^3 = \begin{bmatrix} c4 & -s4 & 0 & 0 \\ s4 & c4 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

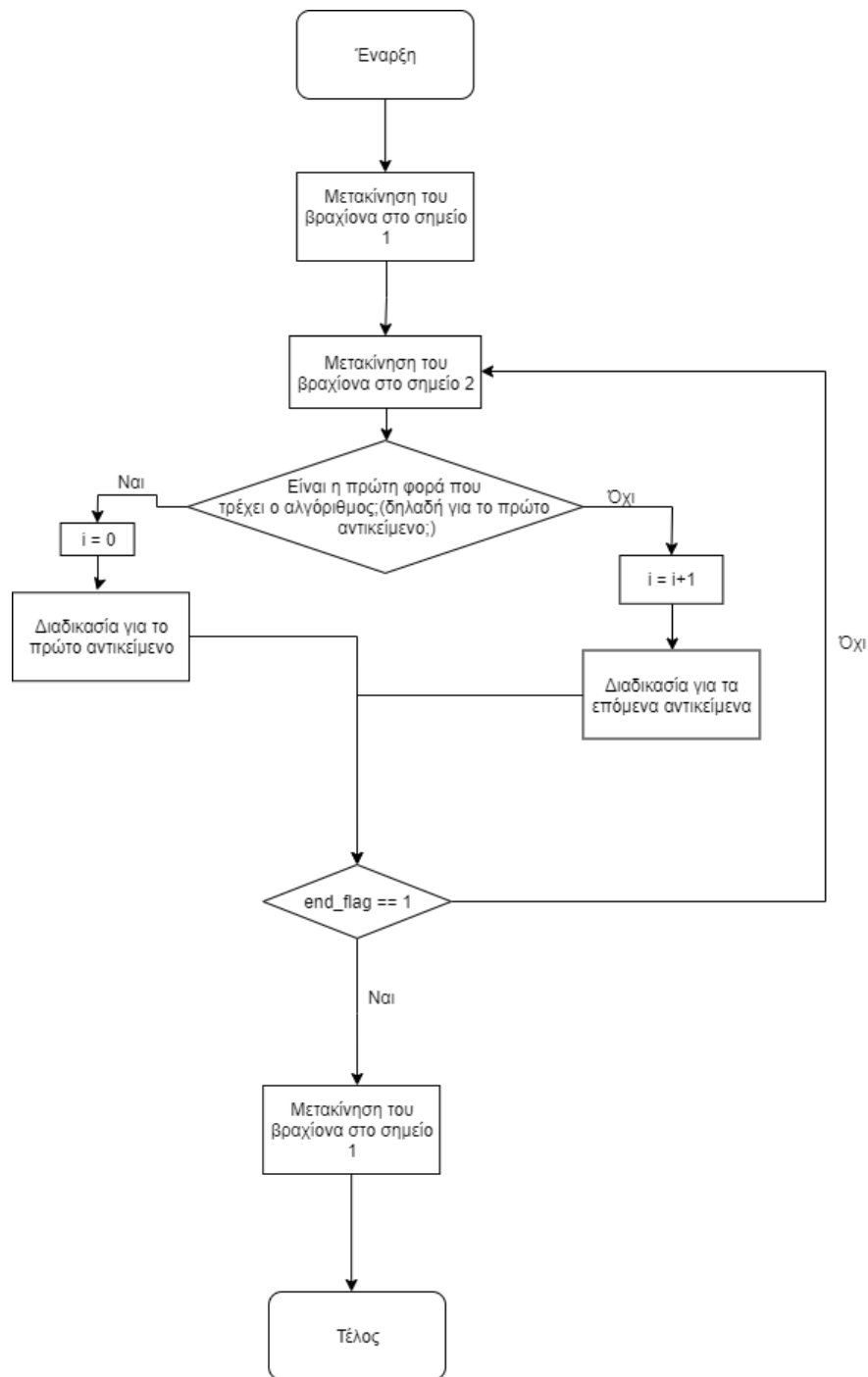
Για το ευθύ κινηματικό μοντέλο από Matlab θα έχουμε:

$$T = A_1^0 A_2^1 A_3^2 A_4^3 = \begin{bmatrix} s_{24} * c_1 & c_{24} * c_1 & -s_1 & l_1 * s_2 * c_1 \\ s_{24} * s_1 & c_{24} * s_1 & c_1 & l_1 * s_2 * s_1 \\ c_{24} & -s_{24} & 0 & l_0 + l_1 * c_2 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

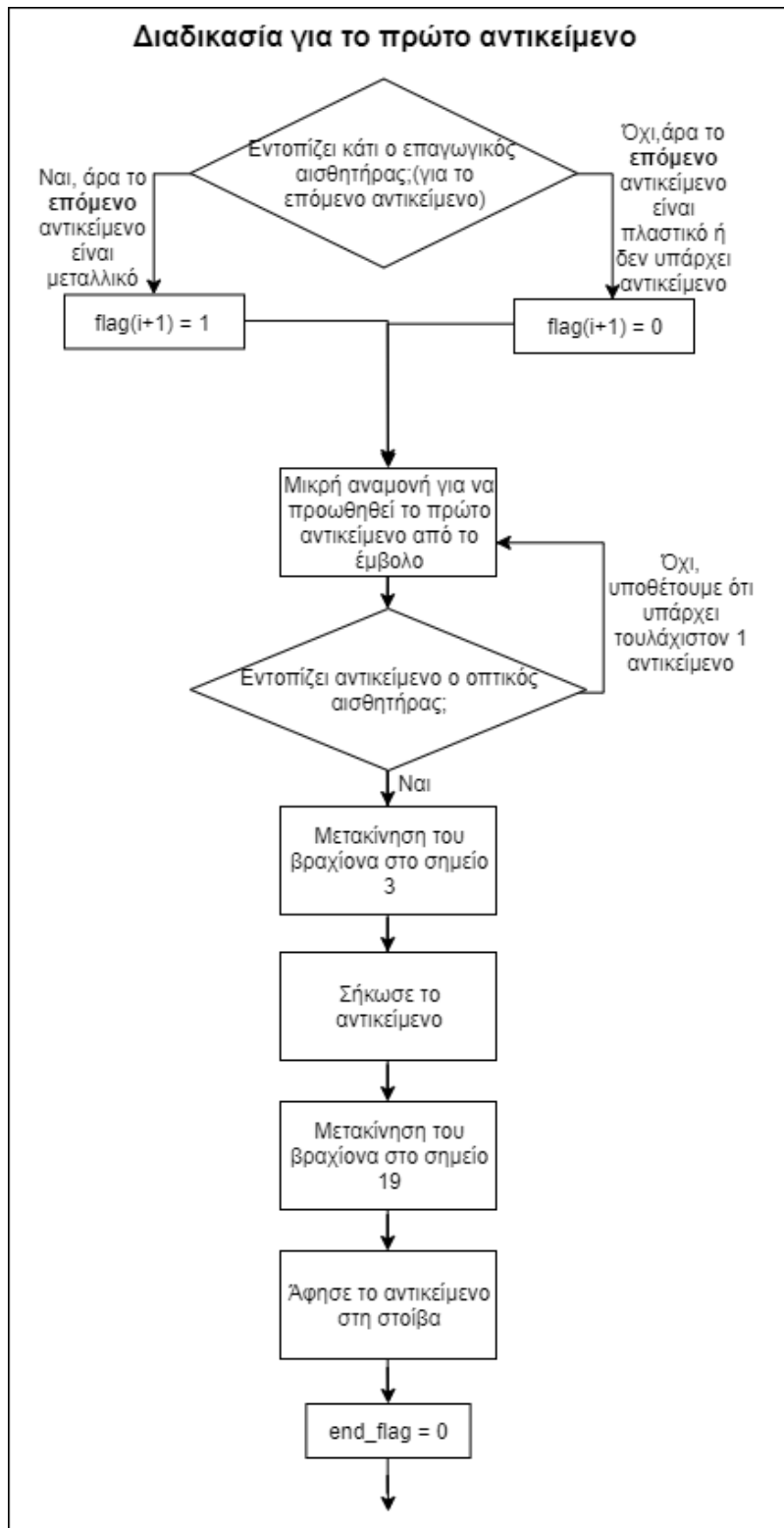
Πειραματική Διάταξη

Η πειραματική διάταξη της 1ης Εργαστηριακής Άσκησης, αποτελούνταν από ένα **ρομποτικό βραχίονα** 5 βαθμών ελευθερίας (5 περιστροφικές αρθρώσεις: 1 αρθρωση στην βάση, 3 αρθρώσεις με παράλληλους άξονες πάνω στον βραχίονα και 1 για την περιστροφή του end-effector). Τα διάφορα λοιπόν rotations στις αρθρώσεις, πραγματοποιούνταν με τη βοήθεια **Σερβοκινητήρων** οι οποίοι δέχονταν εντολές από τον **Η/Υ**. Να σημειωθεί ότι σε αντίθεση με τους DC κινητήρες, οι σερβοκινητήρες που διέθεταν κλειστό κύκλωμα, προσέγγιζαν την ζητούμενη θέση χωρίς την απαίτηση εξωτερικού ελεγκτή (για έλεγχο θέσης ταχύτητας κλπ). Η πειραματική διάταξη πλασιωνόταν και από τρία έμβολα πνευματικά, τα οποία λειτουργούσαν με πίεση αέρα.

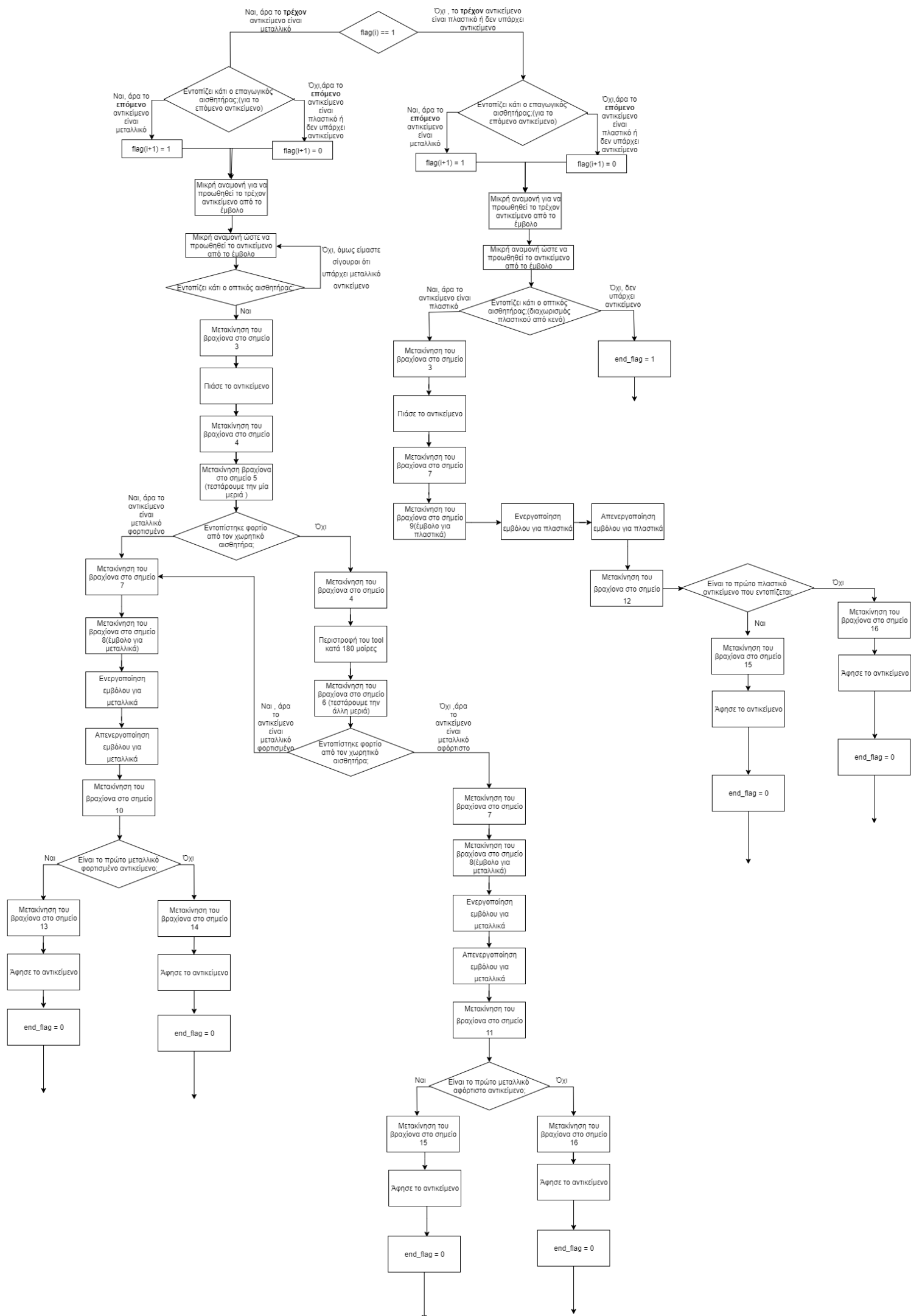
Διάγραμμα Ροής



Όπου:

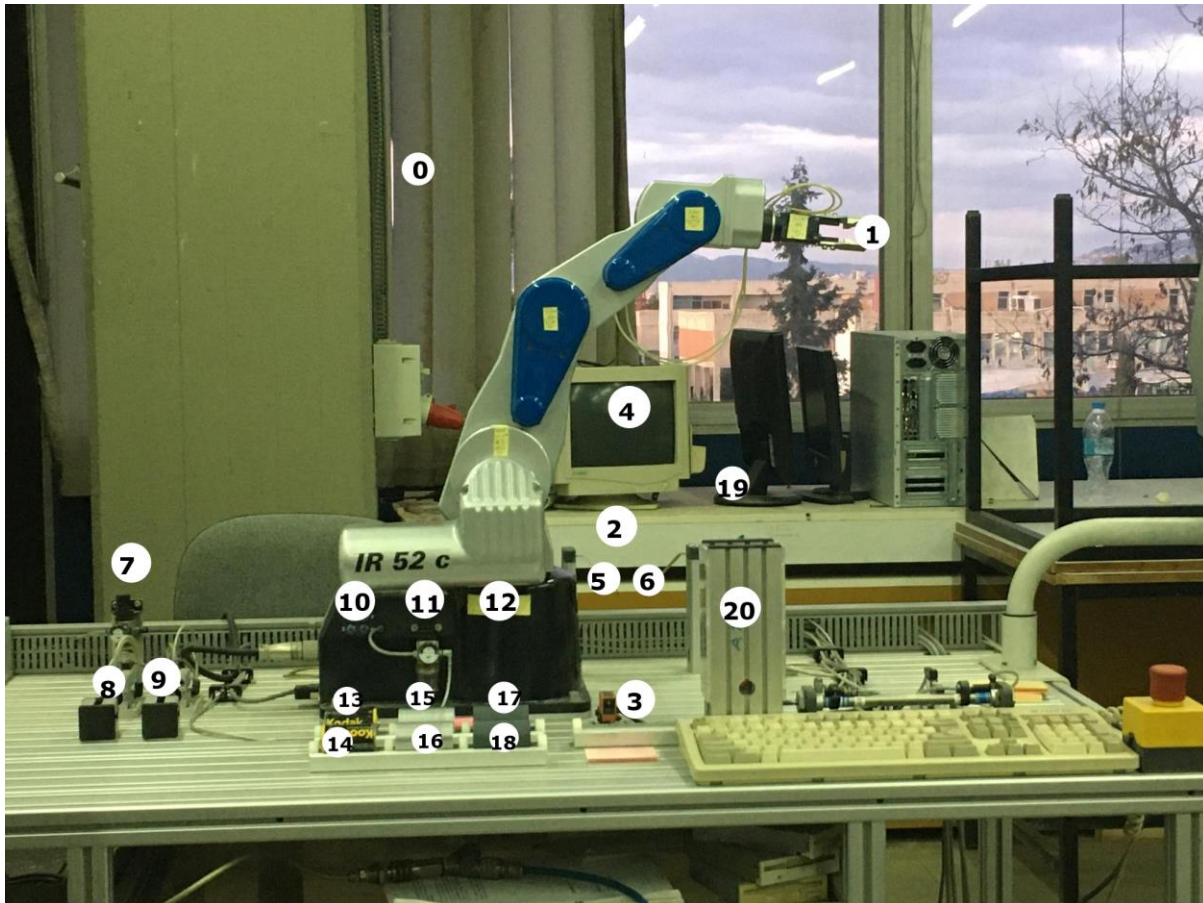


Διαδικασία για επόμενα αντικείμενα



Ψευδοκώδικας

Ο ψευδοκώδικας γράφτηκε με βάση το παρακάτω σχήμα:



Σημειώσεις: Το σημείο 0 είναι η θέση του ρομποτικού βραχίονα πριν τρέξει το πρόγραμμα.
Το σημείο 2 βρίσκεται πάνω από το σημείο 3.
Το σημείο 4 βρίσκεται πάνω από τα σημεία 5,6.

Βασικές εντολές που χρησιμοποιούμε στον ψευδοκώδικα:

- move_arm(position, mode): Κινεί τον βραχίονα στο point που θέλουμε(position), με το mode που θέλουμε(PTP:Point To Point, Line)
- gripper(mode): Χειρίζεται την λαβή του βραχίονα(open, close)
- embolus(mode): Χειρίζεται το έμβολο που σπρώχνει τα αντικείμενα(push, return)
- put_label(type): Βάζει ετικέτα στα αντικείμενα(plastic, metal)
- charged: Παίρνει τις τιμές 1,0 ανάλογα με το αν είναι φορτισμένο ή όχι το αντικείμενο
- read(sensor): Διαβάζει την τιμή των αισθητήρων και την αποθηκεύει(visual, inductive, voltage)
- check(position): Ελέγχει αν υπάρχει αποθηκευτικός χώρος στα points 13, 15, 17, αλλιώς τοποθετούμε τα αντικείμενα στα points 14, 16, 18 αντίστοιχα
- next_object: Μεταβλητή για το είδος του επόμενου αντικειμένου που βλέπει ο επαγωγικός αισθητήρας
- current_object: Μεταβλητή για το είδος του αντικειμένου που έχουμε τώρα

Ο ψευδοκώδικας:

#Initialisation

```
move_arm(point_1, PTP)
check(point_13) = 0
check(point_15) = 0
check(point_17) = 0
next_object = read(inductive)
```

#Put first object back in the top of the stack(because the inductive sensor can check the second object)

```
embolus(push)
embolus(return)
move_arm(point_2, PTP)
gripper(open)
move_arm(point_3, Line)
gripper(close)
move_arm(point_2, Line)
move_arm(point_19, PTP)
move_arm(point_20, Line)
gripper(open)
move_arm(point_19, Line)
```

#Preparing for the loop

```
current_object = next_object
move_arm(point_1, PTP)
embolus(push)
embolus(return)
next_object = read(inductive) #read(inductive) returns 1 if metal, else 0
```

#Loop

```
while (read(visual) == 1) { ***Go to end for explanation
    move_arm(point_2, PTP)
    move_arm(point_3, Line)
    gripper(close)
    move_arm(point_2, Line)
    #If object is metal, check if charged
    if (current_object == 1) { #If object is metal
        move_arm(point_4, PTP)
        move_arm(point_5, Line)
        if (read(voltage) == 0) { #read(voltage) returns 1 if charged, else 0
            move_arm(point_4, PTP)
            move_arm(point_6, PTP)
            if (read(voltage) == 1) {
                charged = 1
            }
            else {
                charged = 0
            }
        }
        else {
            charged = 1
        }
        move_arm(point_4, PTP)
    }
}
```

#Put metal or plastic label

```
move_arm(point_1, PTP)
move_arm(point_7, PTP)
if (current_object == 1) {
    move_arm(point_8, Line)
    put_label(metal)
}
else {
    move_arm(point_9, Line)
    put_label(plastic)
}
move_arm(point_7, Line)
```

#Store

```
if ((current_object == 1) && (charged == 1)) {
    move_arm(point_10, PTP)
    if (check(point_13) == 0) {
        move_arm(point_13, Line)
        gripper(open)
        check(point_13) = 1
    }
    else {
        move_arm(point_14, Line)
        gripper(open)
    }
    move_arm(point_10, Line)
}
else if ((current == 1) && (charged == 0)) {
    move_arm(point_11, PTP)
    if (check(point_15) == 0) {
        move_arm(point_15, Line)
        gripper(open)
        check(point_15) = 1
    }
    else {
        move_arm(point_16, Line)
        gripper(open)
    }
    move_arm(point_11, Line)
}
else {
    move_arm(point_12, PTP)
    if (check(point_17) == 0) {
        move_arm(point_17, Line)
        gripper(open)
        check(point_17) = 1
    }
    else {
        move_arm(point_18, Line)
        gripper(open)
    }
    move_arm(point_12, Line)
}
move_arm(point_1, PTP)
```

```
    current_object = next_object
    embolus(push)
    embolus(return)
    next_object = read(inductive)
} #While ends here
gripper(close)
```

***If `read(visual)==1` then we have an object, else it is air. The inductive sensor can only identify metal/non-metal, so it can't see any difference in plastic and air.