



**ΠΑΝΕΠΙΣΤΗΜΙΟ  
ΔΥΤΙΚΗΣ ΜΑΚΕΔΟΝΙΑΣ**

---

## **Ενσωματωμένα Συστήματα**

**Ενότητα:** ΕΡΓΑΣΤΗΡΙΑΚΗ ΑΣΚΗΣΗ: CIRC-13

Δρ. Μηνάς Δασυγένης

[mdasyg@ieee.org](mailto:mdasyg@ieee.org)

**Τμήμα Μηχανικών Πληροφορικής και Τηλεπικοινωνιών**

Εργαστήριο Ψηφιακών Συστημάτων και Αρχιτεκτονικής Υπολογιστών

<http://arch.ict.e.uowm.gr/mdasyg>

---

# Άδειες Χρήσης

- Το παρόν εκπαιδευτικό υλικό υπόκειται σε άδειες χρήσης Creative Commons.
- Για εκπαιδευτικό υλικό, όπως εικόνες, που υπόκειται σε άλλου τύπου άδειας χρήσης, η άδεια χρήσης αναφέρεται ρητώς.



# Χρηματοδότηση

- Το παρόν εκπαιδευτικό υλικό έχει αναπτυχθεί στα πλαίσια του εκπαιδευτικού έργου του διδάσκοντα.
- Το έργο «**Ανοικτά Ψηφιακά Μαθήματα του Πανεπιστημίου Δυτικής Μακεδονίας**» έχει χρηματοδοτήσει μόνο τη αναδιαμόρφωση του εκπαιδευτικού υλικού.
- Το έργο υλοποιείται στο πλαίσιο του Επιχειρησιακού Προγράμματος «Εκπαίδευση και Δια Βίου Μάθηση» και συγχρηματοδοτείται από την Ευρωπαϊκή Ένωση (Ευρωπαϊκό Κοινωνικό Ταμείο) και από εθνικούς πόρους.



Ευρωπαϊκή Ένωση  
Ευρωπαϊκό Κοινωνικό Ταμείο



ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ & ΘΡΗΣΚΕΥΜΑΤΩΝ, ΠΟΛΙΤΙΣΜΟΥ & ΑΘΛΗΤΙΣΜΟΥ  
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



## Περιεχόμενα

1.Σκοπός της άσκησης.....	4
2.Παραδοτέα.....	4
3.Κατασκευή του κυκλώματος CIRC-13.....	4
4.Προγραμματισμός του κυκλώματος CIRC-13.....	5
5.Παραμετροποίηση του κυκλώματος CIRC-13.....	6
5.1Βαθμονόμηση (Calibration).....	6
5.2Ενδιαφέρουσες εφαρμογές.....	6

## 1. Σκοπός της άσκησης

- Χρήση του αναπτυξιακού περιβάλλοντος του Arduino<sup>1</sup> για τη συγγραφή και τη μεταφόρτωση προγραμμάτων στην πλακέτα του Arduino Uno.
- Δημιουργία κυκλώματος ελέγχου ενός Servo με χρήση αισθητήρα κάμψης.

## 2. Παραδοτέα






- Ένα αρχείο zip με τα project **CIRC\_13**, **CIRC\_13\_c1** που θα δημιουργήσετε.
- Ένα screenshot από το arduino IDE που δείχνει ότι η μεταγλώττιση (compile) έγινε με επιτυχία και ταυτόχρονα εμφανίζει το μέγεθος του δυαδικού σχεδίου για κάθε κύκλωμα.
- Ένα video 5-10 sec επίδειξης του κυκλώματος μαζί με ηχητική περιγραφή για κάθε κύκλωμα.

**Χρόνος ολοκλήρωσης εργαστηρίου: 25 λεπτά.**

Στο παρόν εργαστήριο θα χρησιμοποιηθεί ένας αισθητήρας κάμψης σαν συσκευή εισόδου. Ο αισθητήρας κάμψης χρησιμοποιεί άνθρακα πάνω σε μία λωρίδα πλαστικό, για να λειτουργεί σαν ποτενσιόμετρο, μόνο που τώρα η αντίσταση δεν μεταβάλλεται γυρίζοντας κάποιον διακόπτη αλλά λυγίζοντας τον αισθητήρα. Χρησιμοποιείται ξανά διαιρέτης τάσης (όπως στα εργαστήρια 8 και 9) προκειμένου να ανιχνευθεί η αλλαγή στην αντίσταση από το Arduino. Ο αισθητήρας λυγίζει προς μια κατεύθυνση ( η “ριγέ” πλευρά να βρίσκεται στο εξωτερικό της καμπύλης ) και όσο περισσότερο λυγίζει, τόσο μεγαλώνει η αντίσταση. Η αντίσταση κυμαίνεται από 10K Ohm μέχρι 35K Ohm. Στο κύκλωμά μας θα χρησιμοποιήσουμε τον αισθητήρα κάμψης για να ελέγξουμε την κίνηση ενός Servo.

## 3. Κατασκευή του κυκλώματος CIRC-13

Για την εκπόνηση του κυκλώματος CIRC-13 απαιτούνται τα εξής μέρη:

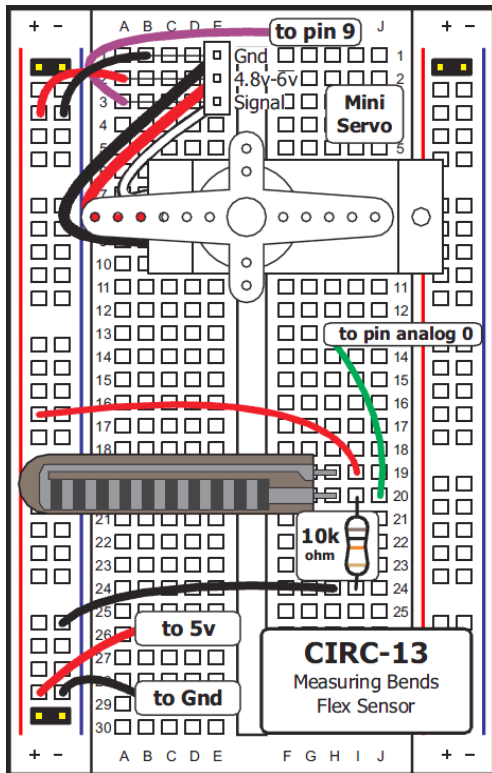
Στοιχείο	Περιγραφή και ποσότητα
	1x <u>Mini Servo</u>
	8x <u>Καλώδια</u>
	1x <u>Αντίσταση 10k Ohm</u> ( <u>Καφέ – Μαύρο – Καφέ</u> )
	1x <u>Αισθητήρας κάμψης</u>
	1x <u>3pin Header</u>

<sup>1</sup> Χρησιμοποιήθηκε υλικό από το [SparkFun Inventors Kit for Arduino](#)

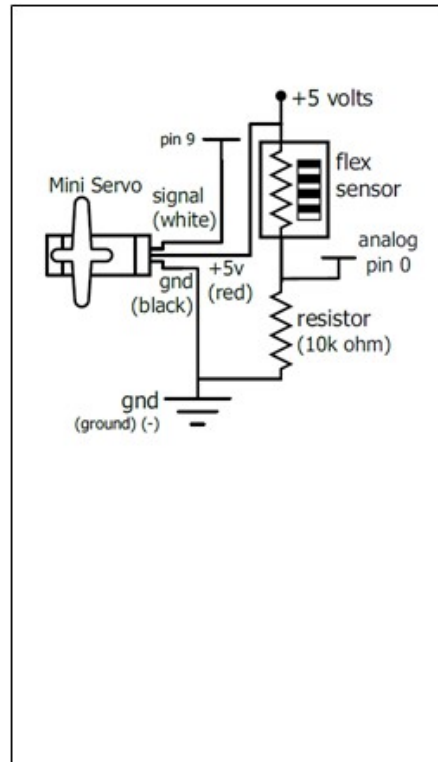
Τα φυλλάδια δεδομένων (datasheets) των υλικών βρίσκονται παρακάτω:

- [Αισθητήρας κάμψης](#)
- [Αντίσταση 10k Ohm](#)
- [Mini Servo](#)

Κατασκευάστε το κύκλωμα, σύμφωνα με τις παρακάτω σχηματικές αναπαραστάσεις:



Σχήμα 1



Σχήμα 2

Τοποθετήστε το 3pin Header όπως φαίνεται στο *Σχήμα 1*. Συνδέστε κάθε καλώδιο του Servo με το 3pin Header και στη συνέχεια με τρία καλώδια συνδέστε το άσπρο καλώδιο του Servo **με το pin 9** του Arduino, το **κόκκινο** καλώδιο στην **τάση +5V**, ενώ το **μαύρο** στη γείωση. Τοποθετήστε στη συνέχεια τον **αισθητήρα κάμψης** στο breadboard και συνδέστε **το ένα άκρο** του στα **+5V** και **το άλλο** σε μία **αντίσταση 10k Ohm** το άλλο άκρο της οποίας πηγαίνει στη **γείωση**, καθώς και με το **αναλογικό pin A0** του Arduino.

Τέλος συνδέστε τα +5V και Gnd στο Arduino.

**ΠΡΟΣΟΧΗ1:** Ο αισθητήρας κάμψης πρέπει να λυγίζει από τη μέση προς την άκρη, όχι στο σημείο που είναι οι συνδετήρες.

**ΠΡΟΣΟΧΗ2:** Ο αισθητήρας κάμψης πρέπει να λυγίζει ως 90 μοίρες όχι περισσότερο. Επίσης, να αποφεύγονται οι παραπάνω από μια κάμψεις.

## 4. Προγραμματισμός του κυκλώματος CIRC-13

Αντιγράψτε τον παρακάτω κώδικα στο προγραμματιστικό περιβάλλον του Arduino.

(Εναλλακτικά κατεβάστε τον από [εδώ](#))

Κώδικας προγραμματισμού του κυκλώματος

```
//Based on File>Examples>Servo>Knob
//Controlling a servo position using a potentiometer (variable resistor)
//by Michal Rinott <http://people.interaction-ivrea.it/m.rinott>

#include <Servo.h>

Servo myservo; //create servo object to control a servo

int potpin = 0; // analog pin used to connect the potentiometer
int val; // variable to read the value from the analog pin

void setup()
{
  Serial.begin(9600);
  myservo.attach(9); //attaches the servo on pin 9 to the servo object
}

void loop()

{
  val=analogRead(potpin);
  //reads the value of the potentiometer (value between 0 and 1023)
  Serial.println(val);
  val=map(val,50,300,0,179);
  //scale it to use it with the servo (value between 0 and 180)
  myservo.write(val);
  //sets the servo position according to the scaled value
  delay(15); //waits for the servo to get there
}
```

Αποθηκεύστε το ως CIRC\_13 και στη συνέχεια φορτώστε το πρόγραμμα στο Arduino.

Σε περίπτωση που το Servo δεν κινείται ελέγξτε μήπως έχετε συνδέσει κάτι λάθος.

Αν το Servo δεν κινείται όπως θα περιμένατε ίσως λυγίζετε τον αισθητήρα κάμψης προς τη λάθος πλευρά (*ενδέχεται να λειτουργεί μόνο προς μια κατεύθυνση*). Δοκιμάστε να τον λυγίσετε με τη “ριγέ” πλευρά να βρίσκεται στο εξωτερικό της καμπύλης.

Τέλος, αν το Servo κινείται μόνο μια φορά θα πρέπει να βαθμονομηθεί σωστά (*θα εξηγηθεί παρακάτω*).

## 5. Παραμετροποίηση του κυκλώματος CIRC-13

### 5.1 Βαθμονόμηση (Calibration)

Ανεξάρτητα απ' το αν το Servo κινείται σωστά ή όχι, πιθανότατα το εύρος τιμών που παίρνουμε από τον αισθητήρα είναι ανακριβές.

Για να προσαρμόσουμε το εύρος στις σωστές του τιμές θα ακολουθήσουμε την εξής διαδικασία:

1. Ανοίγουμε το μενού **Εργαλεία>Σειριακή οθόνη** (*Ctrl+Shift+M*).
2. Παρατηρούμε την τιμή που μας επιστρέφει ο αισθητήρας άκαμπτος.
3. Αλλάζουμε την τιμή **fromLow** στη συνάρτηση **map(value, fromLow, fromHigh, toLow, toHigh)** από **50** στην τιμή που βρήκαμε στο **βήμα 2**. Αν η τιμή που βρέθηκε στο **βήμα 2** είναι **50** δε χρειάζεται να προβούμε σε καμία αλλαγή.
4. Επιστρέφουμε στη σειριακή οθόνη.
5. Λυγίζουμε εντελώς τον αισθητήρα (90 μοίρες) και σημειώνουμε την τιμή που μας επιστρέφεται.
6. Αλλάζουμε την τιμή της **fromHigh** στη συνάρτηση **map(value, fromLow, fromHigh, toLow, toHigh)** από 300 στην τιμή που βρήκαμε στο βήμα 5. Αν η τιμή που βρέθηκε στο **βήμα 5** είναι **300** δε χρειάζεται να προβούμε σε καμία αλλαγή.

Αποθηκεύστε το πρόγραμμα ως CIRC\_13\_c1 και φορτώστε το στο Arduino.

Επιβεβαιώστε την ορθή λειτουργία του κυκλώματος.

### 5.2 Ενδιαφέρουσες εφαρμογές

Μπορείτε να δείτε μερικές ενδιαφέρουσες εφαρμογές του αισθητήρα κάμψης στους παρακάτω συνδέσμους:

- [One player Rock Paper Scissors Glove](#)
- [Electronic Plant Brace](#)