



**ΠΑΝΕΠΙΣΤΗΜΙΟ
ΔΥΤΙΚΗΣ ΜΑΚΕΔΟΝΙΑΣ**

Ενσωματωμένα Συστήματα

Ενότητα: ΕΡΓΑΣΤΗΡΙΑΚΗ ΑΣΚΗΣΗ: CIRC-04

Δρ. Μηνάς Δασυγένης

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



ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ

Περιεχόμενα

1.Σκοπός της άσκησης.....	4
2.Παραδοτέα.....	4
3.Κατασκευή του κυκλώματος CIRC-04.....	4
4.Προγραμματισμός του κυκλώματος CIRC-04.....	5
5.Παραμετροποίηση του κυκλώματος CIRC-04.....	6
5.1Προγραμματισμός του Servo χωρίς την χρήση των έτοιμων βιβλιοθηκών του Arduino.....	6
5.2Ενδιαφέρουσες εφαρμογές.....	7

1. Σκοπός της άσκησης

- Χρήση του αναπτυξιακού περιβάλλοντος του Arduino¹ για τη συγγραφή και τη μεταφόρτωση προγραμμάτων στην πλακέτα του Arduino Uno.
- Δημιουργία κυκλώματος ελέγχου ενός σερβοκινητήρα.

2. Παραδοτέα




- Ένα αρχείο zip με τα project **CIRC_04**, **CIRC_04_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: 30 λεπτά.

Σε εφαρμογές όπου χρειάζεται μεγαλύτερος έλεγχος και ακρίβεια στην κίνηση από αυτή που μας παρέχει ένα μοτέρ, χρησιμοποιούμε Servos. Στο εσωτερικό ενός servo, υπάρχει ένα μικρό κιβώτιο ταχυτήτων (για να κάνει πιο ισχυρή την κίνηση) και μερικά ηλεκτρονικά στοιχεία (μοτέρ, ποτενσιόμετρο κτλ). Ένα τυπικό servo μπορεί να κινείται μεταξύ των 0 και 180 μοιρών. Η κίνηση ελέγχεται μέσω ενός χρονισμένου παλμού, μεταξύ 1.25 ms (0 μοίρες) και 1.75 ms (180 μοίρες). Ο χρόνος ενδέχεται να διαφέρει μεταξύ διαφορετικών κατασκευαστών. Αν ο παλμός στέλνεται μεταξύ κάθε 25 - 50 ms το servo λειτουργεί ομαλά. Το Arduino έχει μια βιβλιοθήκη η οποία δίνει τη δυνατότητα να ελέγχετε 2 servos μαζί (συνδεδεμένα στα pins 9, 10) με μόνο μια γραμμή κώδικα.

3. Κατασκευή του κυκλώματος CIRC-04

Για την εκπόνηση του κυκλώματος CIRC-04 απαιτούνται τα εξής μέρη:

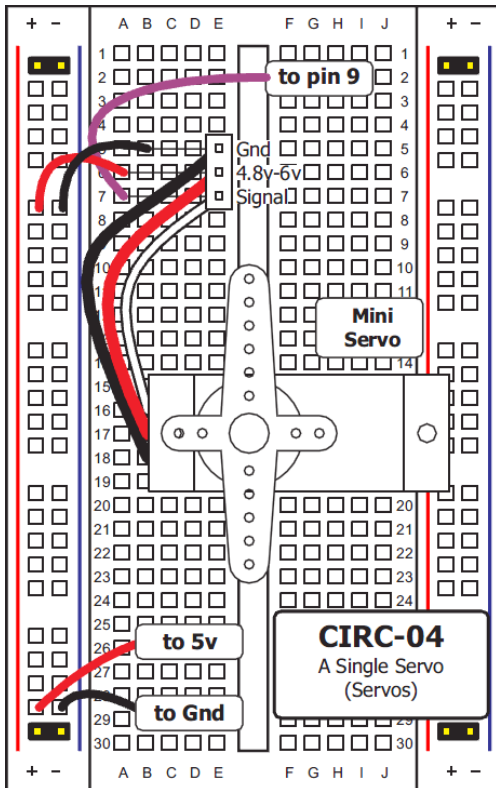
Στοιχείο	Περιγραφή και ποσότητα
	1x <u>Mini Servo</u>
	5x <u>Καλώδια</u>
	1x <u>3pin Header</u>

¹ Χρησιμοποιήθηκε υλικό από το [SparkFun Inventors Kit for Arduino](#)

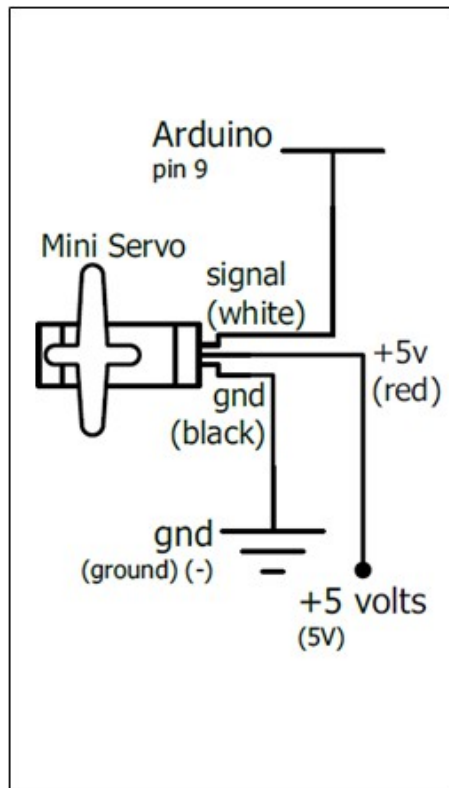
Τα φυλλάδια δεδομένων (datasheets) των υλικών βρίσκονται παρακάτω:

➤ [Mini Servo](#)

Κατασκευάστε το κύκλωμα, σύμφωνα με τις παρακάτω σχηματικές αναπαραστάσεις:



Σχήμα 1



Σχήμα 2

Συνδέστε το Servo με το 3pin Header και στη συνέχεια τοποθετήστε το στο breadboard όπως φαίνεται στο *Σχήμα 1*. Συνδέστε το άσπρο καλώδιο του Servo **με το pin 9** του Arduino, το **κόκκινο** καλώδιο στην **τάση +5 V**, ενώ το **μαύρο** στη γείωση. Τέλος συνδέστε τα +5V και Gnd στο Arduino.

Στο servo τοποθετήστε το εξάρτημα που έχει μια ακμή (“ένα δόντι”) μόνο για να μπορείτε να παρακολουθήσετε καλύτερα την μετακίνηση.

***** Προσοχή:** Ενδέχεται το header pin να μη μπορεί να σταθεροποιηθεί πάνω στο breadboard. Σε αυτή την περίπτωση, αφού προγραμματίσετε το arduino θα ασκείτε ελαφριά πίεση με το δάχτυλο στο συνδετήρα για να λειτουργεί. *******

4. Προγραμματισμός του κυκλώματος CIRC-04

Αντιγράψτε τον παρακάτω κώδικα στο προγραμματιστικό περιβάλλον του Arduino.

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

Κώδικας προγραμματισμού του κυκλώματος

```
// Sweep
// by BARRAGAN <http://barraganstudio.com>

#include <Servo.h>
Servo myservo; // create servo object to control a servo
               // a maximum of eight servo objects can be created

int pos = 0; // variable to store the servo position

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

void loop()
{
  for(pos = 0; pos < 180; pos += 1) // goes from 0 degrees to 180 degrees
  {
    // in steps of 1 degree
    myservo.write(pos); // tell servo to go to position in variable 'pos'
    delay(15); // waits 15ms for the servo to reach the position
  }
  for(pos = 180; pos>=1; pos--=1) // goes from 180 degrees to 0 degrees
  {
    myservo.write(pos); // tell servo to go to position in variable 'pos'
    delay(15); // waits 15ms for the servo to reach the position
  }
}
```

***** Προσοχή: Το servo δέχεται εντολές τοποθέτησης από 0 έως 180 μοίρες. Μη δοκιμάσετε να στείλετε τιμή μεγαλύτερη από 180 γιατί θα υπερθερμανθεί και μπορεί να καεί. *****

Αποθηκεύστε το ως CIRC_04 και στη συνέχεια φορτώστε το πρόγραμμα στην πλακέτα.

Αν το Servo δεν περιστρέφεται, ελέγξτε μήπως τυχόν το έχετε τοποθετήσει ανάποδα. Ελέγξτε επίσης αν έχετε συνδέσει σωστά την πηγή.

Αν το Servo ξεκινά να περιστρέφεται αλλά καταλήγει να κάνει σπασμωδικές κινήσεις, και υπάρχει ένα λαμπάκι πάνω στην πλακέτα arduino που αναβοσβήνει, τότε η τροφοδοσία του USB δεν είναι αρκετή και θα πρέπει να χρησιμοποιηθεί ένα εξωτερικό καλώδιο τροφοδοσίας.

5. Παραμετροποίηση του κυκλώματος CIRC-04

5.1 Προγραμματισμός του Servo χωρίς την χρήση των έτοιμων βιβλιοθηκών του Arduino

Μπορείτε να προγραμματίσετε το Servo χωρίς να χρησιμοποιήσετε τις έτοιμες βιβλιοθήκες του Arduino.

Αντιγράψτε τον παρακάτω κώδικα στο προγραμματιστικό περιβάλλον του Arduino:

```
int servoPin = 9;
void setup(){
  pinMode(servoPin,OUTPUT); }

void loop() {
  int pulseTime = 2100;  //(the number of microseconds
                          //to pause for (1500 90 degrees
                          // 900 0 degrees 2100 180 degrees)
  digitalWrite(servoPin, HIGH);
  delayMicroseconds(pulseTime);
  digitalWrite(servoPin, LOW);
  delay(25);
}
```

Τροποποιήστε τον (αν δε λειτουργεί κανονικά), ώστε να περιστρέφεται για 180ο.

Στη συνέχεια, αποθηκεύστε το ως CIRC_04_c1 και φορτώστε το πρόγραμμα στο Arduino.

5.2 Ενδιαφέρουσες εφαρμογές

Μπορείτε να δείτε μερικές ενδιαφέρουσες εφαρμογές με Servos στους παρακάτω συνδέσμους:

[Xmas Hit Counter](#)

[Open Source Robotic Arm](#)

[Servo Walker](#)