



**ΠΑΝΕΠΙΣΤΗΜΙΟ
ΔΥΤΙΚΗΣ ΜΑΚΕΔΟΝΙΑΣ**

Ενσωματωμένα Συστήματα

Ενότητα: ΕΡΓΑΣΤΗΡΙΑΚΗ ΑΣΚΗΣΗ: CIRC-06

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-06.....	4
4.Προγραμματισμός του κυκλώματος CIRC-06.....	5
5.Παραμετροποίηση του κυκλώματος CIRC-06.....	5
5.1Ρύθμιση της ταχύτητας της μελωδίας.....	5
5.2Έλεγχος τονικότητας κάθε νότας.....	5
5.3Σύνθεση μιας δικιάς σας μελωδίας.....	5

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 πιεζοηλεκτρικού στοιχείου.

2. Παραδοτέα

- Ένα αρχείο zip με τα project **CIRC_06**, **CIRC_06_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: 20 λεπτά.

Το Arduino μπορεί να χρησιμοποιηθεί και για τον έλεγχο συσκευών που παράγουν ήχο, παρόλο που ο ήχος είναι αναλογικό φαινόμενο. Για να το πετύχουμε αυτό θα χρησιμοποιήσουμε ένα πιεζοηλεκτρικό στοιχείο. Το πιεζοηλεκτρικό στοιχείο παράγει έναν ήχο κάθε φορά που το διαπερνά ένας παλμός ρεύματος. Αν πάλλουμε το πιεζοηλεκτρικό στοιχείο στην κατάλληλη συχνότητα οι παραγόμενοι ήχοι θα παράγουν μουσικές νότες (π.χ. 440Hz για να παραχθεί η νότα λα).

3. Κατασκευή του κυκλώματος CIRC-06

Για την εκπόνηση του κυκλώματος CIRC-05 απαιτούνται τα εξής μέρη:

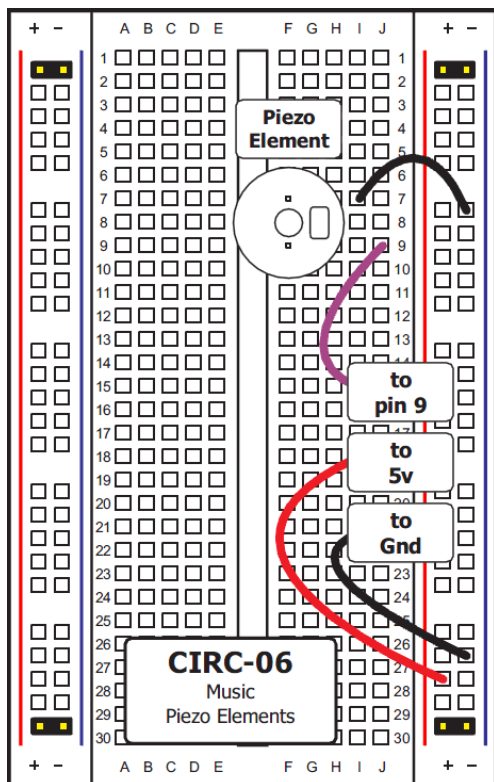
Στοιχείο	Περιγραφή και ποσότητα
	1x Πιεζοηλεκτρικό στοιχείο
	4x Καλώδια

Τα φυλλάδια δεδομένων (datasheets) των υλικών βρίσκονται παρακάτω:

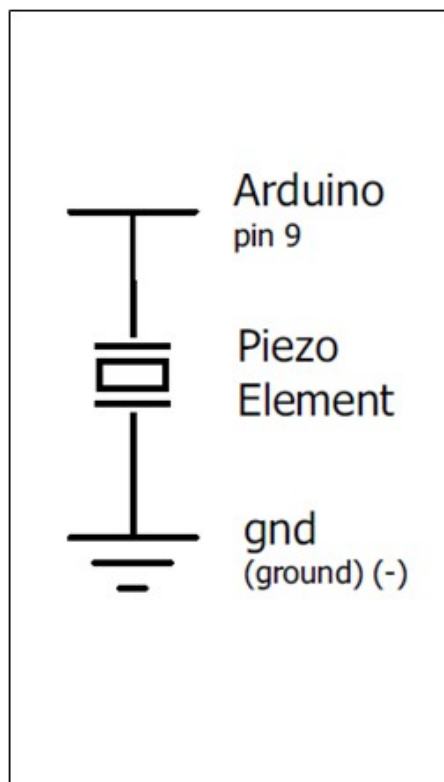
- [Πιεζοηλεκτρικό στοιχείο](#)

Κατασκευάστε το κύκλωμα, σύμφωνα με τις παρακάτω σχηματικές αναπαραστάσεις:

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



Σχήμα 1



Σχήμα 2

Συνδέστε το **θετικό ακροδέκτη** του πιεζοηλεκτρικού στοιχείου στο **pin 9** του **Arduino**, και τον **αρνητικό ακροδέκτη** στη **γείωση**.

4. Προγραμματισμός του κυκλώματος CIRC-06

Αντιγράψτε τον παρακάτω κώδικα στο προγραμματιστικό περιβάλλον του Arduino.

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

Κώδικας προγραμματισμού του κυκλώματος

```

/* Melody
 * (cleft) 2005 D. Cuartielles for K3
 *
 * This example uses a piezo speaker to play melodies. It sends
 * a square wave of the appropriate frequency to the piezo, generating
 * the corresponding tone.
 *
 * The calculation of the tones is made following the mathematical
 * operation:
 *
 * timeHigh = period / 2 = 1 / (2 * toneFrequency)
 *
 * where the different tones are described as in the table:
 *
 *note frequency period timeHigh
 * c 261 Hz 3830 1915
 * d 294 Hz 3400 1700
 * e 329 Hz 3038 1519
 * f 349 Hz 2864 1432
 * g 392 Hz 2550 1275
 * a 440 Hz 2272 1136

```

Κώδικας προγραμματισμού του κυκλώματος

```
* b    493 Hz    2028  1014
* C    523 Hz    1912  956
*
* http://www.arduino.cc/en/Tutorial/Melody
*/

int speakerPin = 9;

int length = 15; // the number of notes
char notes[] = "ccggaagffeeddc "; // a space represents a rest
int beats[] = { 1, 1, 1, 1, 1, 1, 2, 1, 1, 1, 1, 1, 2, 4 };
int tempo = 300;

void playTone(int tone, int duration) {
  for (long i = 0; i < duration * 1000L; i += tone * 2) {
    digitalWrite(speakerPin, HIGH);
    delayMicroseconds(tone);
    digitalWrite(speakerPin, LOW);
    delayMicroseconds(tone);
  }
}

void playNote(char note, int duration) {
  char names[] = { 'c', 'd', 'e', 'f', 'g', 'a', 'b', 'C' };
  int tones[] = { 1915, 1700, 1519, 1432, 1275, 1136, 1014, 956 };

  // play the tone corresponding to the note name
  for (int i = 0; i < 8; i++) {
    if (names[i] == note) {
      playTone(tones[i], duration);
    }
  }
}

void setup() {
  pinMode(speakerPin, OUTPUT);
}

void loop() {
  for (int i = 0; i < length; i++){
    if (notes[i] == ' '){
      delay(beats[i] * tempo); // rest
    } else {
      playNote(notes[i], beats[i] * tempo);
    }

    // pause between notes
    delay(tempo / 2);
  }
}
```

Αποθηκεύστε το ως CIRC_05 και στη συνέχεια φορτώστε το πρόγραμμα στην πλακέτα. **Αν το Arduino σβήνει**, ελέγξτε μήπως έχετε τοποθετήσει ανάποδα τον καταχωρητή ολίσθησης.

5. Παραμετροποίηση του κυκλώματος CIRC-06

5.1 Ρύθμιση της ταχύτητας της μελωδίας.

Η ταχύτητα της μελωδίας καθορίζεται από την τιμή της μεταβλητής `tempo`. Μεγαλύτερες τιμές, δίνουν γρηγορότερη εκτέλεση της μελωδίας, ενώ μικρότερες πιο αργή. Δοκιμάστε να αλλάξετε την τιμή της στο παρακάτω τμήμα του κώδικα:

```
int tempo = 300;
```

Στη συνέχεια, αποθηκεύστε και φορτώστε το πρόγραμμα στο Arduino ώστε να διαπιστώσετε την αλλαγή στη λειτουργία του κυκλώματος.

5.2 Έλεγχος τονικότητας κάθε νότας

Μπορείτε επίσης να αλλάξετε την τονικότητα κάθε νότας, αλλάζοντας τις τιμές του πίνακα `tones[]`.

```
int tones[ ] = { 1915, 1700, 1519, 1432, 1275, 1136, 1014, 956 };
```

Κάθε τιμή του πίνακα είναι η αντίστοιχη νότα στον πίνακα `names[]`.

5.3 Σύνθεση μιας δικιάς σας μελωδίας

Το πρόγραμμα είναι ρυθμισμένο να παίζει το παιδικό τραγούδι "Twinkle Twinkle Little Star", αλλά η αλλαγή του είναι απλή υπόθεση.

Αλλάξτε το παρακάτω τμήμα του κώδικα:

```
int length = 15; // the number of notes  
char notes[ ] = "ccggaagffeeddc "; // a space represents a rest  
int beats[ ] = { 1, 1, 1, 1, 1, 1, 2, 1, 1, 1, 1, 1, 1, 2, 4 };
```

με αυτό:

```
int length = 13; // the number of notes  
char notes[ ] = "ccdcfecdcgfc "; // a space represents a rest  
int beats[ ] = { 1, 1, 1, 1, 1, 2, 1, 1, 1, 1, 1, 2, 4 };
```

Αφού το δοκιμάσετε, προσπαθήστε να **συνθέσετε τη δικιά σας μελωδία** και παραδώστε τον κώδικα.

Αποθηκεύστε το ως `CIRC_06_c1` και φορτώστε το στο Arduino.