



**ΠΑΝΕΠΙΣΤΗΜΙΟ  
ΔΥΤΙΚΗΣ ΜΑΚΕΔΟΝΙΑΣ**

---

## **Ενσωματωμένα Συστήματα**

**Ενότητα:** ΕΡΓΑΣΤΗΡΙΑΚΗ ΑΣΚΗΣΗ: CIRC-01

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



ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ

## Περιεχόμενα

1.Σκοπός της άσκησης.....	4
2.Παραδοτέα.....	4
3.Κατασκευή του κυκλώματος CIRC-01.....	5
4.Επαλήθευση παραμέτρων στο προγραμματιστικό περιβάλλον του Arduino.....	5
5.Προγραμματισμός του κυκλώματος CIRC-01.....	6
6.Παραμετροποίηση του κυκλώματος CIRC-01.....	6
6.1Αλλαγή του pin σύνδεσης του LED.....	6
6.2Αλλαγή της χρονικής διάρκειας on/off του LED.....	7
6.3Αλλαγή της φωτεινότητας του LED.....	7
6.4Ομαλή μετάβαση μεταξύ των καταστάσεων on/off του LED (Fading).....	7

## 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 LED.

## 2. Παραδοτέα

- Ένα αρχείο zip με τα project **CIRC\_01**, **CIRC\_01\_d1**, **CIRC\_01\_d2**, **CIRC\_01\_d3**, **CIRC\_01\_d4** 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: 15 λεπτά.**




Τα LEDs (δίοδοι εκπομπής φωτός) χρησιμοποιούνται σε πάρα πολλές εφαρμογές. Αυτός είναι και ο λόγος που έχουν συμπεριληφθεί στο SparkFun Inventor's Kit. Σκοπός της άσκησης είναι να κάνουμε ένα LED να ενεργοποιείται και να απενεργοποιείται κατ' επανάληψη.

---

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

### 3. Κατασκευή του κυκλώματος CIRC-01

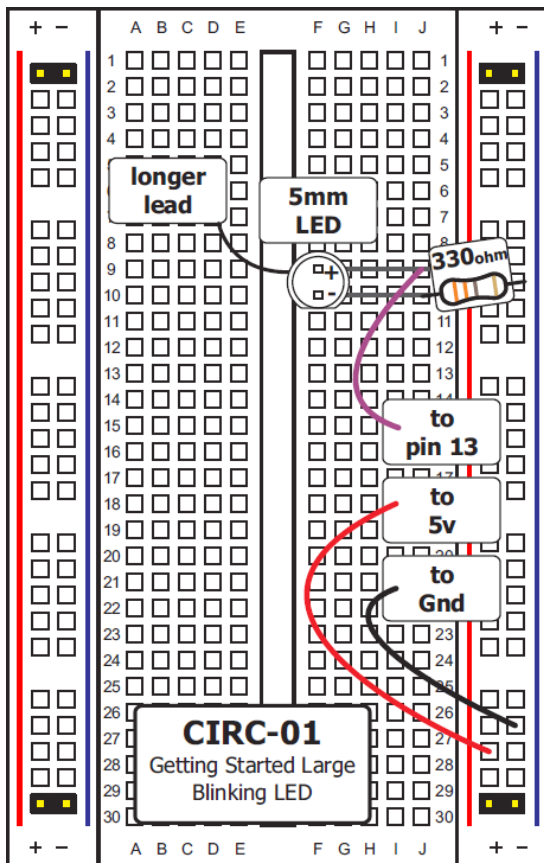
Για την εκπόνηση του κυκλώματος CIRC-01 απαιτούνται τα εξής μέρη:

Στοιχείο	Περιγραφή και ποσότητα
	1x 5mm κίτρινο LED
	1x Αντίσταση 330 Ohm (Πορτοκαλί – Πορτοκαλί – Καφέ )
	3x Καλώδια

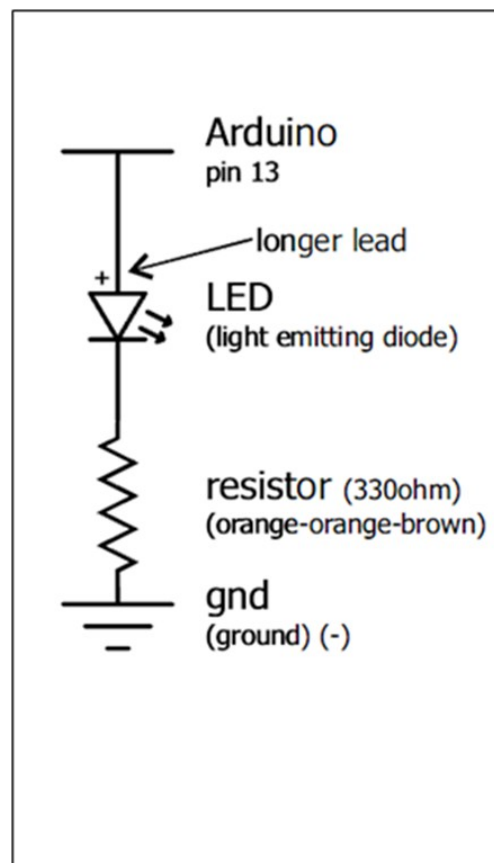
Τα φυλλάδια δεδομένων (datasheets) των υλικών βρίσκονται παρακάτω:

- [LED](#)
- [Αντίσταση 330 Ohm](#)

Κατασκευάστε το κύκλωμα, σύμφωνα με την παρακάτω σχηματική αναπαράσταση:



Σχήμα 1



Σχήμα 2

Συνδέστε το θετικό ακροδέκτη του LED στο 13ο pin του Arduino και μια αντίσταση στον αρνητικό ακροδέκτη, που να πηγαίνει στη γείωση.

**Επισημαίνεται ότι ο θετικός ακροδέκτης του LED είναι αυτός με το μεγαλύτερο μήκος.**

Τέλος συνδέστε, στις κατάλληλες θέσεις του breadboard, ένα καλώδιο στο pin τροφοδοσίας 5V του Arduino και ένα καλώδιο σε ένα από τα τρία pin γείωσης (Gnd) που βρίσκονται πάνω στο Arduino.

## 4. Επαλήθευση παραμέτρων στο προγραμματιστικό περιβάλλον του Arduino

Συνδέστε το Arduino Uno και ανοίξτε το προγραμματιστικό περιβάλλον του.

**Επιβεβαιώστε ότι η επιλεγμένη σειριακή θύρα είναι η θύρα στην οποία είναι συνδεδεμένο το Arduino.**

Πηγαίνετε στις **Ιδιότητες Υπολογιστή>Διαχείριση Συσκευών>Θύρες(COM & LPT)** και συγκρίνετε την αναγραφόμενη θύρα με αυτή που είναι επιλεγμένη στο μενού **Εργαλεία > Σειριακή θύρα**, του προγραμματιστικού περιβάλλοντος του Arduino.

**Βεβαιωθείτε ότι η επιλεγμένη πλακέτα είναι το Arduino Uno**

Πηγαίνετε στο μενού **Εργαλεία > Πλακέτα** και επιλέξτε το Arduino Uno.

**Βεβαιωθείτε ότι ο επιλεγμένος προγραμματιστής είναι ο AVRISP mkII**

Πηγαίνετε στο μενού **Εργαλεία > Προγραμματιστής** και επιλέξτε τον AVRISP mkII.

## 5. Προγραμματισμός του κυκλώματος CIRC-01

Αντιγράψτε τον παρακάτω κώδικα στο προγραμματιστικό περιβάλλον του Arduino.


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


### Κώδικας προγραμματισμού του κυκλώματος

```
/*  
 *Blink  
 *Turns on an LED on for one second, then off for one second, repeatedly  
 *The circuit:  
 * LED connected from digital pin 13 to ground  
 * Note: On most Arduino boards, there is already an LED on the board  
 * connected to pin 13, so you don't need any extra components for this  
 example  
  
 *Created 1 June 2005  
 *By David Cuartielles  
 *http://arduino.cc/en/Tutorial/Blink  
 *based on an original by H. Barragan for the Wiring i/o board  
 */  
  
int ledPin = 13; // LED connected to digital pin 13  
  
// The setup() method runs once, when the sketch starts  
  
void setup(){  
 // initialize the digital pin as an output:  
 pinMode(ledPin, OUTPUT);  
}
```

## Κώδικας προγραμματισμού του κυκλώματος

```
// the loop() method runs over and over again,  
// as long as the Arduino has power  
  
void loop()  
{  
  digitalWrite(ledPin, HIGH); // set the LED on  
  delay(1000); // wait for a second  
  digitalWrite(ledPin, LOW); // set the LED off  
  delay(1000); // wait for a second  
}
```

Πατήστε  **Αποθήκευση (Ctrl+s)**. Θα σας ζητηθεί να δώσετε όνομα φακέλου (*Sketch Folder*). Δώστε “**CIRC-01**” και πατήστε “**Αποθήκευση**”. Το πρόγραμμα θα δημιουργήσει τον φάκελο **CIRC-01** μέσα στον οποίο θα περιέχεται το αρχείο **CIRC-01.ino** με τον κώδικα που γράψατε παραπάνω.

Στη συνέχεια φορτώστε το πρόγραμμα στην πλακέτα πατώντας  **Φόρτωση (Ctrl+u)**.

Μόλις ολοκληρωθεί η διαδικασία φόρτωσης το πρόγραμμα θα εκτελεστεί στην πλακέτα, κάνοντας το LED του κυκλώματος και της πλακέτας να αναβοσβήνει.

**Σε περίπτωση που δε συμβαίνει τίποτα στο κύκλωμα, ελέγξτε την πολικότητα του LED, καθώς και την ορθότητα των παραμέτρων που περιγράφηκαν στην ενότητα 4.**

## 6. Παραμετροποίηση του κυκλώματος CIRC-01

### 6.1 Αλλαγή του pin σύνδεσης του LED

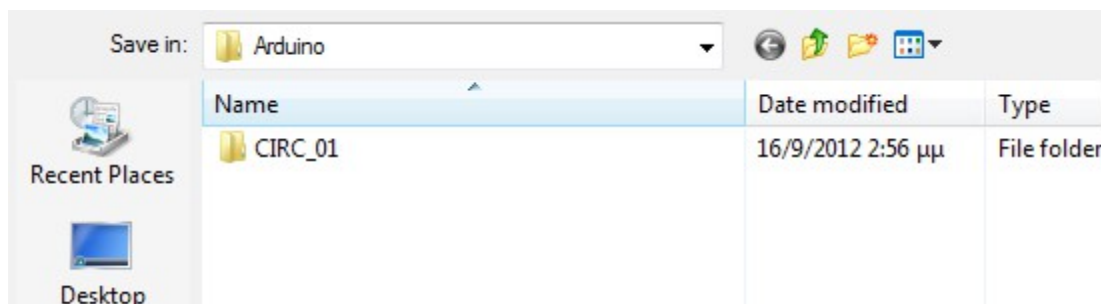
Το LED είναι συνδεδεμένο στο pin 13, αλλά μπορούμε να το συνδέσουμε σε οποιοδήποτε από τα pins του Arduino. Για αλλάξετε το pin αφαιρέστε το καλώδιο από το pin 13 και συνδέστε το σε ένα άλλο pin της επιλογής σας (από 0 ως 13 για τα ψηφιακά ή A0 ως A5 για τα αναλογικά pins).

Ταυτόχρονα αλλάξτε τη γραμμή 15 του κώδικα:

```
int ledPin = 13; // LED connected to digital pin 13
```

Δίνοντας τον αλλαγμένο αριθμό pin (από 0 ως 13 για τα ψηφιακά ή 14 ως 19 για τα αναλογικά pins).

Πατήστε Αποθήκευση ως (Ctrl+Shift+s), και μεταβείτε έξω από το φάκελο του project CIRC\_01 αν είστε μέσα σ αυτόν (θα πρέπει να βλέπετε το φάκελο CIRC\_01).



Δώστε όνομα φακέλου CIRC\_01\_d1 για να αποθηκεύσετε το πρόγραμμα.

Στη συνέχεια, φορτώστε το στην πλακέτα (Ctrl+u).

Επιβεβαιώστε την ορθή λειτουργία του κυκλώματος.

## 6.2 Αλλαγή της χρονικής διάρκειας on/off του LED

Το LED είναι ρυθμισμένο να αναβοσβήνει ανά 1sec (1000ms), κάτι το οποίο μπορούμε να αλλάξουμε.

Αλλάξτε τις τιμές “1000” στο παρακάτω τμήμα του κώδικα:

```
digitalWrite(ledPin, HIGH); // set the LED on
delay(1000); // wait for a second
digitalWrite(ledPin, LOW); // set the LED off
delay(1000); // wait for a second
```

Πατήστε Αποθήκευση ως (Ctrl+Shift+s), μεταβείτε έξω από το φάκελο του ήδη αποθηκευμένου project, αν είστε μέσα σ αυτόν, (όμοια με πριν θα πρέπει να βλέπετε τους φακέλους CIRC\_01, CIRC\_01\_d1) και δώστε όνομα φακέλου **CIRC\_01\_d2** για να αποθηκεύσετε το πρόγραμμα.

Στη συνέχεια, φορτώστε το στην πλακέτα (Ctrl+u).

Επιβεβαιώστε ότι ο χρόνος on/off έχει μεταβληθεί.

## 6.3 Αλλαγή της φωτεινότητας του LED

Εκτός από τον ψηφιακό (on/off) έλεγχο, το Arduino μπορεί να παρέχει έλεγχο που μοιάζει με τον αναλογικό (ενδιάμεσες καταστάσεις) σε ορισμένα pins.

Αλλάξτε την τιμή του pin σε 9.

```
int ledPin = 9;
```

**Προσαρμόστε κατάλληλα και το καλώδιο στο κύκλωμα.**

Στη συνέχεια αντικαταστήστε τον κώδικα μέσα στο loop() με τον παρακάτω κώδικα:

```
analogWrite(ledPin, 20);
```

Όπου 20 μπορείτε να δώσετε οποιαδήποτε τιμή από 0 ως 255.

Πατήστε Αποθήκευση ως (Ctrl+Shift+s), μεταβείτε έξω από το φάκελο του ήδη αποθηκευμένου project (CIRC-01-d2) και δώστε όνομα φακέλου **CIRC\_01\_d3** για να αποθηκεύσετε το πρόγραμμα.

Στη συνέχεια, φορτώστε το στην πλακέτα (Ctrl+u).

Επιβεβαιώστε την ορθή λειτουργία του κυκλώματος.

## 6.4 Ομαλή μετάβαση μεταξύ των καταστάσεων on/off του LED (Fading)



Προσαρμόζοντας τον κώδικα του προηγούμενου ερωτήματος σε δύο διαδοχικές επαναλήψεις από 0 ως 255 η πρώτη, και από 254 ως 1 η δεύτερη, μπορούμε να δώσουμε όλες τις ενδιάμεσες τιμές φωτεινότητας στο LED, δημιουργώντας έτσι, ένα αποτέλεσμα ομαλής μετάβασης μεταξύ των καταστάσεων on/off.

Αντικαταστήστε τον κώδικα μέσα στο loop() με τον παρακάτω κώδικα:

```
for(int i=0; i<=255; i++)
{
  analogWrite(ledPin, i);
  delay(10);
}
```

```
for(int i=254; i>=1; i--)
{
  analogWrite(ledPin, i);
  delay(10);
}
```

Πατήστε Αποθήκευση ως (**Ctrl+Shift+s**), μεταβείτε έξω από το φάκελο του ήδη αποθηκευμένου project (CIRC-01-d3) και δώστε όνομα φακέλου **CIRC\_01\_d4** για να αποθηκεύσετε το πρόγραμμα.

Στη συνέχεια, φορτώστε το στην πλακέτα (**Ctrl+u**).

Επιβεβαιώστε την ορθή λειτουργία του κυκλώματος.

Δείτε επίσης τον κώδικα στο **Αρχείο>Παραδείγματα>01.Basics>Fade** που περιέχει μια διαφορετική υλοποίηση της ίδιας λειτουργίας.