



Συνοδευτικό Υλικό

Ar
duino
ino

4

Basics

LED που αναβοσβήνει και LED με κουμπί

Πρόγραμμα LED που αναβοσβήνει

και LED με κουμπί σε Arduino IDE



Πρόγραμμα LED που αναβοσβήνει και LED με κουμπί σε Arduino IDE

Υλοποίηση_A:

Το πρόγραμμα σε κώδικα C του Arduino (Υλοποίηση I. Μαλαμίδης):

```
void setup() {  
    pinMode(5, OUTPUT); // Όρισε το πιν 5 ως έξοδο (κόκκινο LED)  
    pinMode(6, OUTPUT); // Όρισε το πιν 6 ως έξοδο (πράσινο LED)  
    pinMode(2, INPUT); // Όρισε το πιν 2 ως είσοδο (διακόπτης)  
}  
  
void loop() {  
    if (digitalRead(2) == HIGH) { // Αν ο διακόπτης είναι πατημένος...  
        digitalWrite(5, HIGH); // Άναψε το κόκκινο LED  
    }  
    else { // ... αλλιώς ...  
        digitalWrite(5, LOW); // Σβήσε το κόκκινο LED  
    }  
    digitalWrite(6, HIGH); // Άναψε το πράσινο LED  
    delay(3000); // Περίμενε 3 δευτερόλεπτα εδώ  
    digitalWrite(6, LOW); // Σβήσε το πράσινο LED  
    delay(1000); // Περίμενε 3 δευτερόλεπτα εδώ  
}
```



Πρόγραμμα LED που αναβοσβήνει και LED με κουπί σε Arduino IDE

Υλοποίηση_B:

Το πρόγραμμα σε κώδικα C του Arduino (Υλοποίηση I. Μαλαμίδης):

```
unsigned long start_time; // Δημιούργησε μια μεταβλητή τύπου unsigned long  
integer με όνομα start_time
```

```
void setup() {
```

```
  pinMode(5, OUTPUT); // Όρισε το πιν 5 ως έξοδο (κόκκινο LED)
```

```
  pinMode(6, OUTPUT); // Όρισε το πιν 6 ως έξοδο (πράσινο LED)
```

```
  pinMode(2, INPUT); // Όρισε το πιν 2 ως είσοδο (διακόπτης)
```

```
}
```

```
void loop() {
```

```
  digitalWrite(6, HIGH); // Άναψε το πράσινο LED
```

```
  start_time=millis(); // Βάλε στην start_time την τιμή της millis()
```

```
  while (millis() - start_time < 3000) { // Εφόσον δεν πέρασαν 3 δευτ κάνε:
```

```
    if (digitalRead(2) == HIGH) { // Αν ο διακόπτης είναι πατημένος...
```

```
      digitalWrite(5, HIGH); // Άναψε το κόκκινο LED
```

```
    }
```

```
    else { // ... αλλιώς ...
```

```
      digitalWrite(5, LOW); // Σβήσε το κόκκινο LED
```

```
    }
```

```
  } // (τέλος της while)
```



Πρόγραμμα LED που αναβοσβήνει και LED με κουμπί σε Arduino IDE

```
digitalWrite(6, LOW); // Σβήσε το πράσινο LED
start_time=millis(); // Βάλε στην start_time την τιμή της millis()
while (millis() - start_time < 3000) { // Εφόσον δεν πέρασαν 3 δευτ κάνε:
  if (digitalRead(2) == HIGH) { // Αν ο διακόπτης είναι πατημένος...
    digitalWrite(5, HIGH); // Άναψε το κόκκινο LED
  }
  else { // ... αλλιώς ...
    digitalWrite(5, LOW); // Σβήσε το κόκκινο LED
  }
} // (τέλος της while)
} // (τέλος της loop)
```

Οι εντολές που χρησιμοποιούνται είναι:

- unsigned long start_time

Εδώ χρησιμοποιούμε μια ακέραια μεταβλητή τύπου unsigned long integer (32 bit).

Μπορεί να αποθηκεύσει αριθμό από 0 ως 4.294.967.295 (ο απλός long integer, πάλι 32 bit, μπορεί να πάρει τιμές από -2.147.483.648 έως 2.147.483.647).

Επειδή η τιμή της millis() πάει από 0 ως 4.294.967.295, χρειάζεται να έχουμε μια μεταβλητή αντίστοιχου τύπου για να χωράει να αποθηκεύσουμε την τιμή της.

- millis()

Είναι ένας αριθμός που αυξάνεται συνέχεια σε ένα εσωτερικό «χρονόμετρο» του Arduino. Ξεκινάει να μετράει από το 0 όταν δώσουμε ρεύμα στο Arduino και



Πρόγραμμα LED που αναβοσβήνει και LED με κουμπί σε Arduino IDE

αυξάνει την τιμή του κατά 1 κάθε χιλιοστό του δευτερολέπτου (κάθε 1 ms). Ξαναμηδενίζεται όταν φτάσει την τιμή 4.294.967.295 (σε περίπου 50 ημέρες) ή αν πατήσουμε το κουμπί reset.

- `while(... συνθήκη){ ... εντολές }`

Αυτή είναι μια δομή επανάληψης με συνθήκη, γνωστή σε όσους έχουν κάνει προγραμματισμό. Η συνθήκη εξετάζεται και εφόσον είναι αληθής εκτελούνται οι εντολές που υπάρχουν μέσα στα άγκιστρα. Στη συνέχεια η εκτέλεση ξαναπάει στην αρχή της `while` και ελέγχεται πάλι η συνθήκη. Αν δεν ισχύει η συνθήκη κάποια στιγμή, οι εντολές που υπάρχουν στις αγκύλες δεν εκτελούνται και το πρόγραμμα προχωράει παρακάτω, μετά το άγκιστρο τέλους.

- Η δομή ελέγχου `if(... συνθήκη) {... εντολές} else {... άλλες εντολές}` είναι γνωστή από πριν.

Πηγές

Εκπαιδευτικός Οδηγός Προγράμματος STEmpoweringYouth

Οδηγός Arduino για το μάθημα της Πληροφορικής (STEmpoweringYouth, σύνταξη Σ.Π. Λιωνής)

Κεντρικό Site Arduino (<https://www.arduino.cc/>)

Κεντρικό Site Fritzing (<http://fritzing.org/home/>)

Κεντρικό Site Processing (<https://processing.org/>)



Σημείωση

Οι λέξεις Arduino, Arduino UNO καθώς και λογότυπα ή άλλα διακριτικά γνωρίσματα που αναφέρονται στον παρόντα οδηγό ή απεικονίζονται στις εικόνες που αυτός εμπεριέχει είναι κατοχυρωμένα εμπορικά σήματα και διακριτικά γνωρίσματα που συνιστούν εμπορική ιδιοκτησία της Arduino AG. Το Ίδρυμα Vodafone, η Vodafone Παναφον Α.Ε.Ε.Τ ή η SciCo δεν έχουν σχέση με την εν λόγω εταιρεία. Το υλικό πνευματικής ιδιοκτησίας τρίτων σημειώνεται ρητά και διανέμεται με την αντίστοιχη άδεια που ορίζεται από τους όρους χρήσης αυτού.

