



Ar
duino

2

Basics

LED που αναβοσβήνει

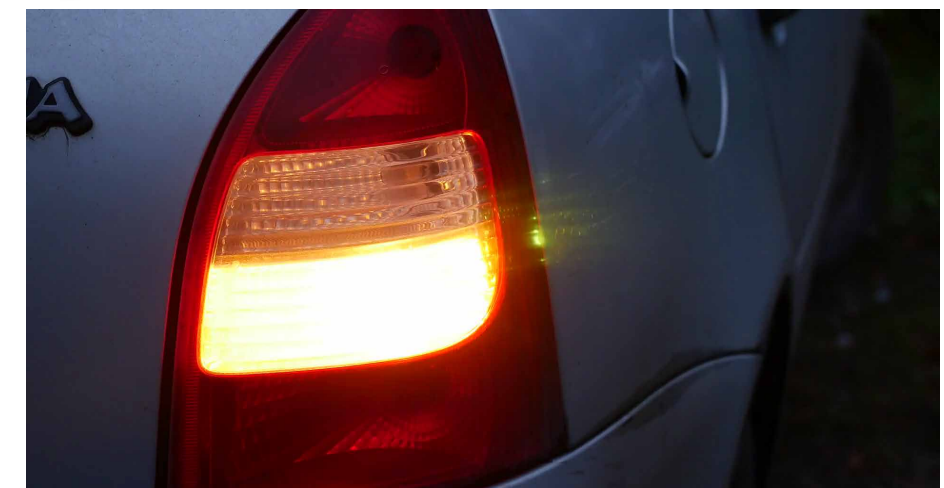
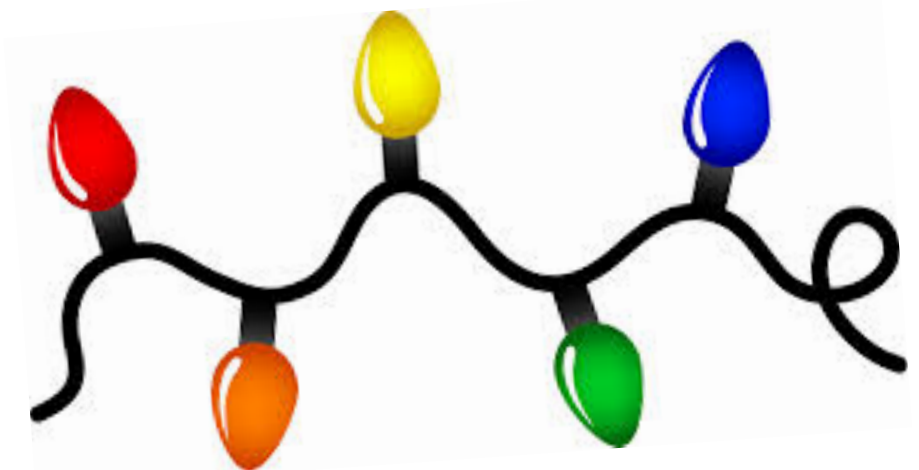


Δραστηριότητα 1: LED που αναβοσβήνει

Τώρα που μάθαμε τα βασικά για το Arduino, είμαστε έτοιμοι να ξεκινήσουμε τις κατασκευές μας!

Σε αυτή τη δραστηριότητα θα φτιάξουμε ένα κύκλωμα που κάνει το λαμπάκι LED να αναβοσβήνει αυτόματα μόνο του.

Πού μπορεί να είναι χρήσιμη μια τέτοια εφαρμογή;





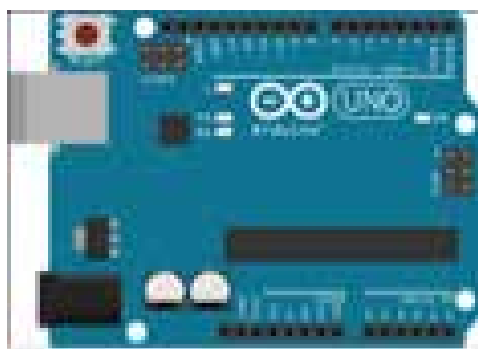
Θα χρειαστούμε:



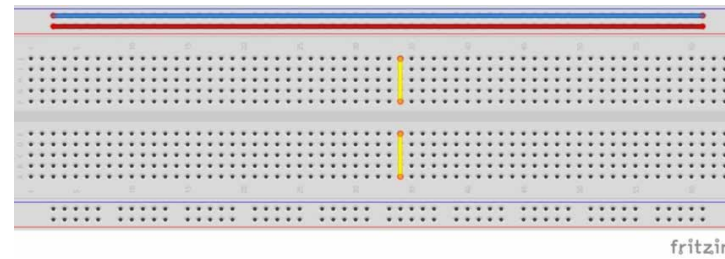
Λαμπάκι LED



Αντίσταση 220 Ω



Πλακέτα Arduino UNO



Breadboard



Καλώδια (x2)
(jumper wires, M-M)



Υπολογιστής

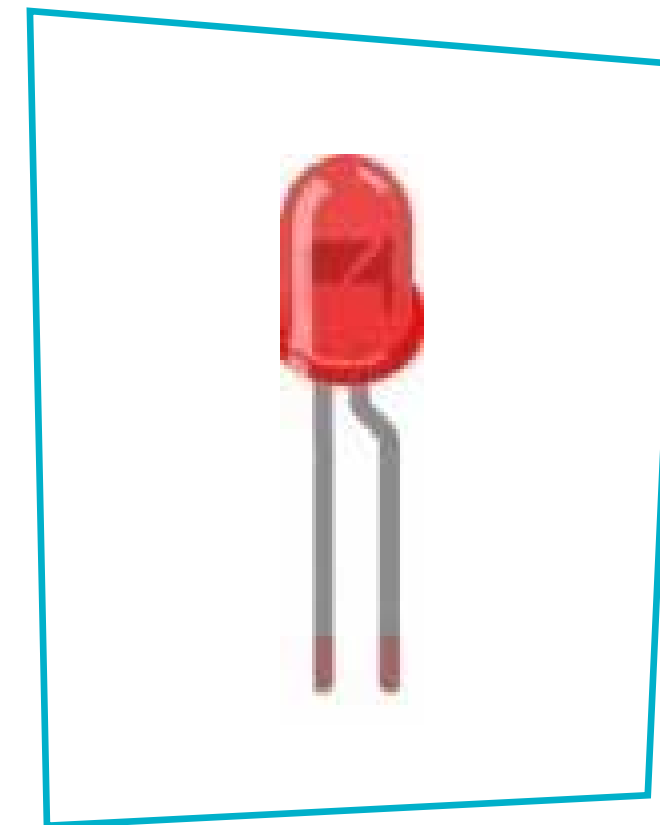


Το LED

Το LED (Light Emitting Diode) είναι ένα στοιχείο, το οποίο όταν διαρρέεται από ρεύμα φωτοβολεί, δηλαδή το γνωστό μας λαμπάκι!

Όσο μεγαλύτερη είναι η ένταση του ρεύματος που διαρρέει το LED, τόσο εντονότερο είναι το φως που παράγεται.

Για να λειτουργήσει το λαμπάκι, οι δύο ακροδέκτες του θα πρέπει να συνδεθούν με το σωστό τρόπο σε μια πηγή ρεύματος (όπως μια μπαταρία).





Η αντίσταση

Γιατί χρειαζόμαστε την αντίσταση;

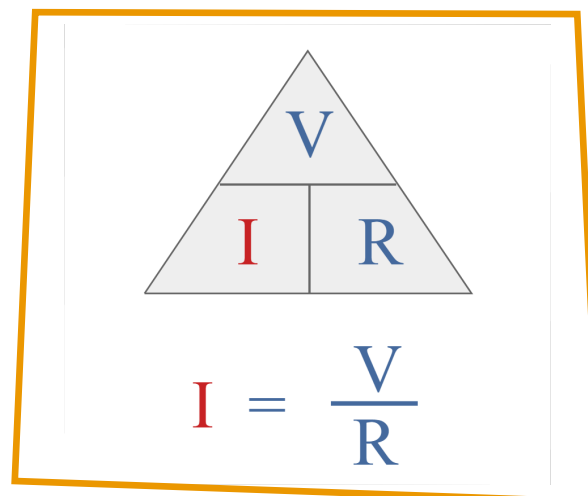
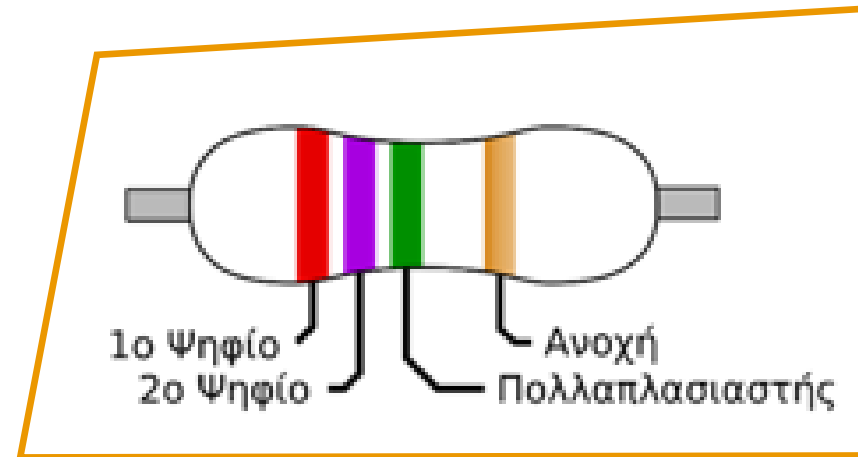
- 1 Προστατεύουμε το λαμπάκι για να μην καεί
- 2 Ρυθμίζουμε την ένταση του φωτός αλλάζοντας την ένταση που διαρρέει το λαμπάκι!



Αντίσταση 220 Ω

Πώς δουλεύουν οι αντιστάτες;

- Προσαρμόζουν το ρεύμα κατάλληλα για τις συσκευές:
 - Διαιρούν την τάση (σύνδεση σε σειρά)
 - Διαιρούν το ρεύμα/ένταση (παράλληλη σύνδεση)
- Δεν έχει σημασία η πολικότητα



Ο νόμος του Ohm

Χρώμα	1 ^η λωρίδα	2 ^η λωρίδα	3 ^η λωρίδα (πολλαπλασιαστής)	4 ^η λωρίδα (ανοχή)
Μαύρο	0	0	$\times 10^0$	$\pm 1\%$ (F)
Καφέ	1	1	$\times 10^1$	$\pm 2\%$ (G)
Κόκκινο	2	2	$\times 10^2$	
Πορτοκαλί	3	3	$\times 10^3$	
Κίτρινο	4	4	$\times 10^4$	
Πράσινο	5	5	$\times 10^5$	$\pm 0.5\%$ (D)
Μπλε	6	6	$\times 10^6$	$\pm 0.25\%$ (C)
Μωβ	7	7	$\times 10^7$	$\pm 0.1\%$ (B)
Γκρι	8	8	$\times 10^8$	$\pm 0.05\%$ (A)
Λευκό	9	9	$\times 10^9$	
Χρυσό			$\times 0.1$	$\pm 5\%$ (J)
Ασημί			$\times 0.01$	$\pm 10\%$ (K)

Χρωματικός κώδικας αντιστάσεων

Π.χ. για τον αντιστάτη του κάτω σχήματος είναι:

1^η ζώνη: Κόκκινο = 2

2^η ζώνη: Κόκκινο = 2

3^η ζώνη: Καφέ = 1

(Σημαίνει: Προσθέτω 1 μηδενικό)

Επομένως σχηματίζεται ο αριθμός: 2 2 0 δηλαδή 220 Ω.

4^η ζώνη: Χρυσό = Ανοχή 5%





Το Arduino UNO ως μπαταρία!

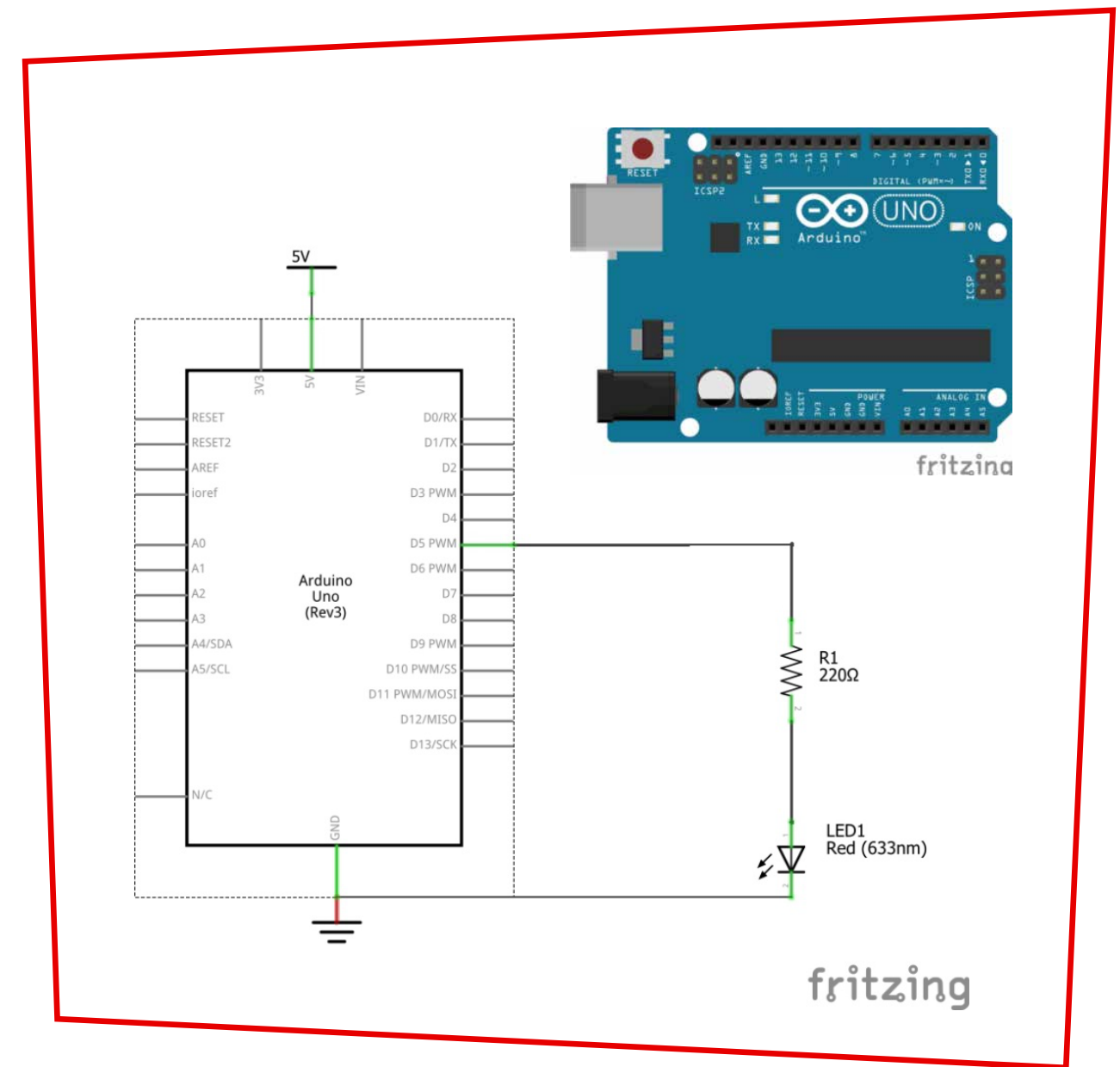
Το Arduino UNO έχει 14 ψηφιακούς ακροδέκτες εισόδου/εξόδου (pin 0-13). Όταν χρησιμοποιούνται ως έξοδοι, μπορούν να τεθούν σε δύο καταστάσεις:

- HIGH (τάση 5V)
- LOW (τάση 0 V)

Το Arduino γίνεται μια προγραμματιζόμενη μπαταρία:

(+) Θετικός πόλος = ψηφιακός ακροδέκτης εξόδου. Όταν βρίσκεται σε κατάσταση HIGH το λαμπάκι θα ανάβει, και όταν βρίσκεται σε κατάσταση LOW το λαμπάκι θα σβήνει

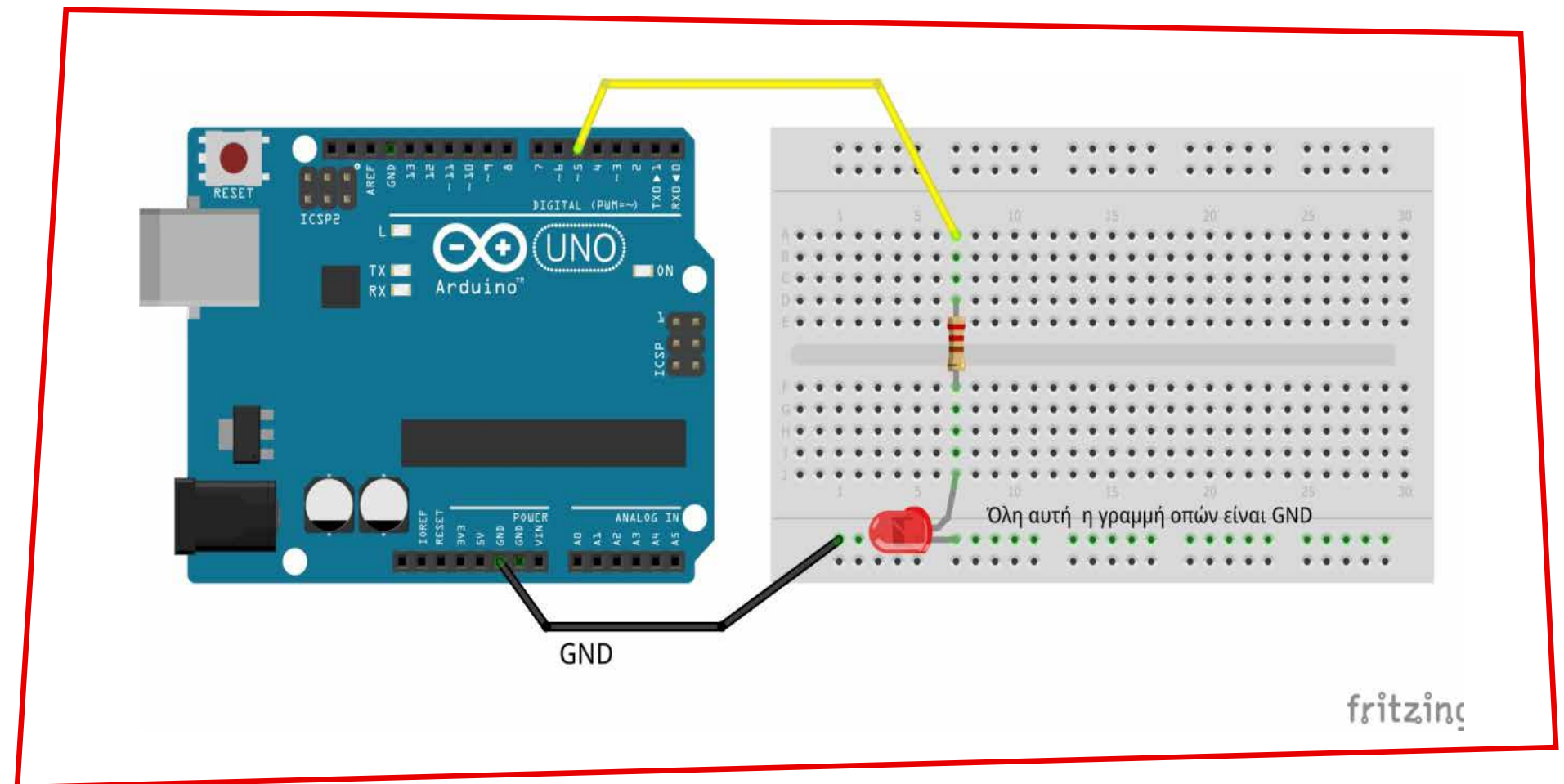
(-) Αρνητικός πόλος = πόλος γείωσης GND





Η κατασκευή του κυκλώματος

- 1 Σύνδεση αντιστάτη με ένα ψηφιακό πιν (0-13)
- 2 Σύνδεση του άλλου άκρου του με το + του LED
- 3 Σύνδεση του - του LED με τη γείωση (GND)



Αυτό πρέπει να είναι το τελικό αποτέλεσμα!



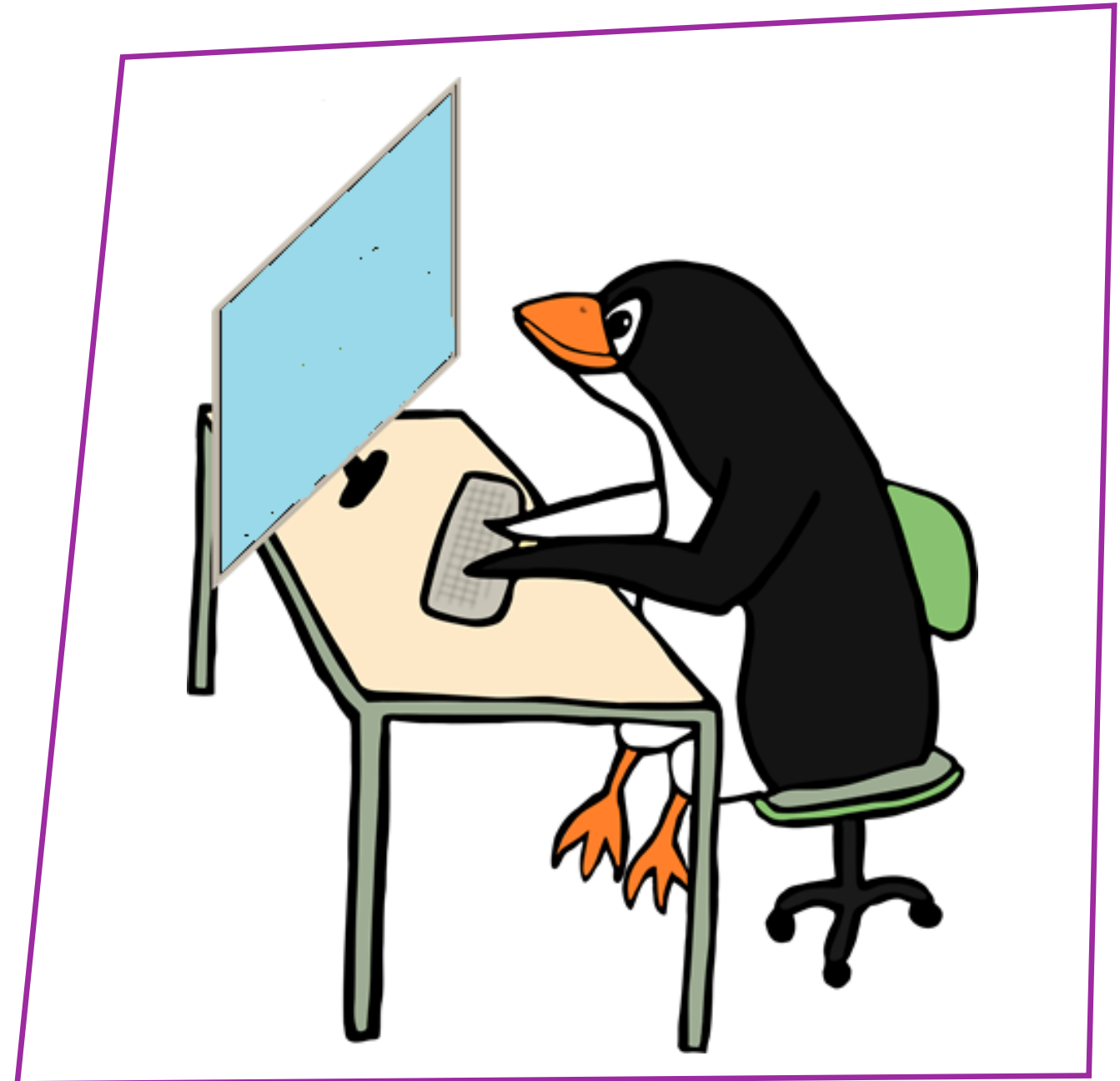
Το πρόγραμμα

Βήμα 1: Γράφουμε το πρόγραμμα

Βήμα 2: Συνδέουμε το Arduino με τον υπολογιστή

Βήμα 3: Φορτώνουμε το πρόγραμμα στο Arduino

Βήμα 4: Πειραματιζόμαστε





Το πρόγραμμα σε Arduino IDE: Οι βασικές εντολές

pinMode: Ορίζει κάποιο pin (0-13) του Arduino ως ΕΙΣΟΔΟ (INPUT) ή ΕΞΟΔΟ (OUTPUT).

digitalWrite: Δίνει στο pin 5 ρεύμα (HIGH) ή όχι (LOW)

delay: Αναμονή για τόσα ms όσα ορίζονται στην παρένθεση

Στο Arduino IDE: Οι εντολές
τελειώνουν με ελληνικό ερωτηματικό

Στον οδηγό της δραστηριότητας θα βρείτε αναλυτικά τον κώδικα.

```
sketch_oct20a | Arduino 1.8.7
Αρχείο Επεξεργασία Σχέδιο Εργαλεία Βοήθεια
Ανέβασμα
sketch_oct20a $
void setup() {
  pinMode(5, OUTPUT); // Ορίσε το pin 5 ως έξοδο (LED)
}
void loop() {
  digitalWrite(5, HIGH); // Αναψε το LED
  delay(1000); // Περίμενε 1 δευτερόλεπτο εδώ
  digitalWrite(5, LOW); // Σβήσε το LED
  delay(1000); // Περίμενε 1 δευτερόλεπτο εδώ
}
8 Arduino/Genuino Uno στο COM4
```



Πειραματιστείτε!

Δοκιμάστε να αλλάξετε πόσο χρόνο το λαμπάκι θα μένει σβηστό και αναμένο.

Μπορούμε να φτιάξουμε έναν κώδικα σημάτων Morse αναβοσβήνοντας το λαμπάκι με καθορισμένο τρόπο;

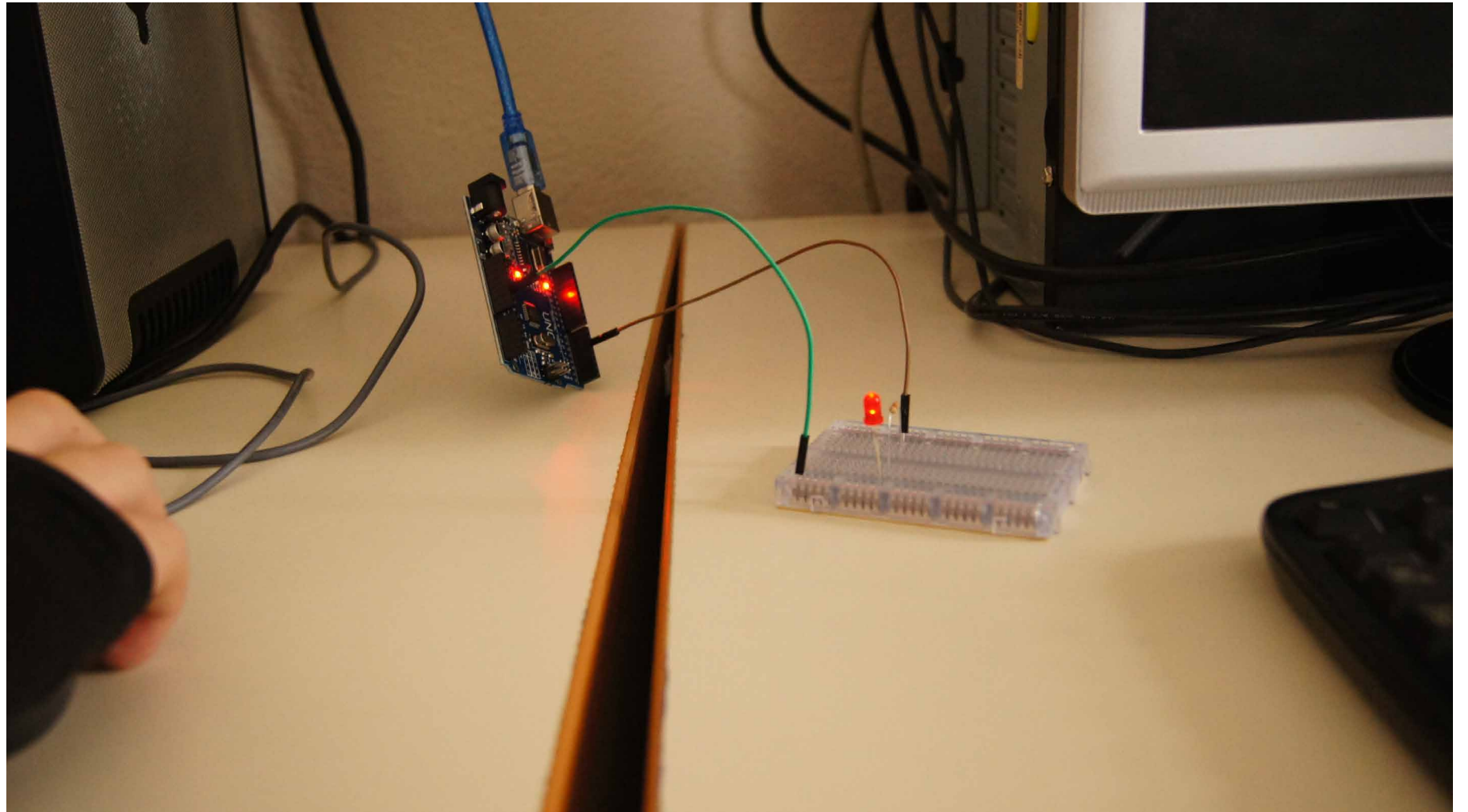


Image credit: Δ. Μαμούρας, STEMpowering Youth