

A close-up photograph of an Arduino PCB, showing various components like a microcontroller, capacitors, and connectors. The image is heavily blurred with a blue bokeh effect, creating a futuristic and technical atmosphere. The text is overlaid on the right side of the image.

Εισαγωγή στο Arduino

single-board μικροελεγκτής
Αριστείδης Γκάτσης - ΕΚΦΕ Καρδίτσας

Τι είναι ο Arduino;

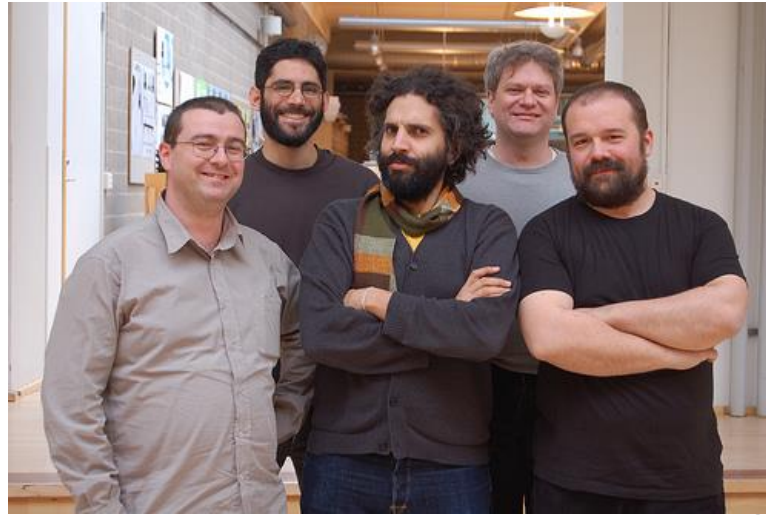
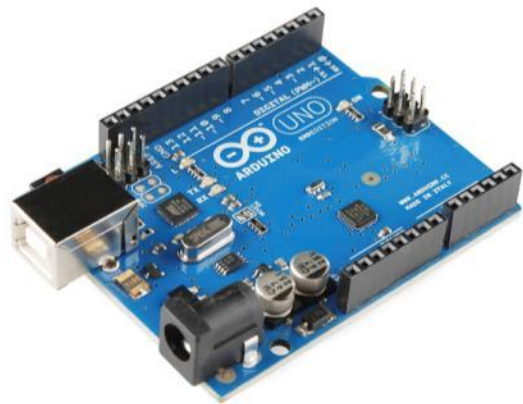
- ▶ Ο Arduino είναι ένας **single-board** μικροελεγκτής, δηλαδή μια απλή μητρική πλακέτα ανοικτού κώδικα με ενσωματωμένο μικροελεγκτή και εισόδους/εξόδους, η οποία μπορεί να προγραμματιστεί με τη γλώσσα **Wiring** (που ουσιαστικά είναι η γλώσσα προγραμματισμού C++ μαζί με ένα σύνολο από βιβλιοθήκες, υλοποιημένες επίσης στην C++).

Τι είναι το Arduino?

- Βασίζεται σε
 - ένα απλό μικροελεγκτή (micro-controller), και
 - ένα περιβάλλον ανάπτυξης (development environment, IDE) για τη συγγραφή λογισμικού
- Μπορεί να χρησιμοποιηθεί για την ανάπτυξη διαδραστικών αντικειμένων
- Είσοδος από ποικιλία διακοπών και sensors,
- Έξοδος σε λαμπτήρες, μοτέρ, κ.α. φυσικές εξόδους.
- Ένα Arduino project μπορεί να είναι stand-alone, ή να επικοινωνεί με λογισμικό που τρέχει σε υπολογιστή.
- Οι πλακέτες μπορούν να αγοραστούν ή να φτιαχτούν
- Το IDE είναι open-source.
- Η γλώσσα προγραμματισμού είναι μια υλοποίηση της Wiring.

Arduino Board

- Επινοήθηκε στην Ivrea, Italy
- Το 2005
- Massimo Banzi & David Cuartielles
- Open Source Hardware
- Processor














Προδιαγραφές Arduino Uno

- ▶ Microcontoller: ATmega 328
- ▶ Operating Voltage 5V
- ▶ Input Voltage (recommended) 7-12V
- ▶ Input Voltage (limits) 6-20V
- ▶ Digital I/O Pins 14 (of which 6 provide PWM output)
- ▶ Analog Input Pins 6
- ▶ Flash Memory 32 KB (of which 0.5 KB used by bootloader)
- ▶ SRAM 2 KB (ATmega328)
- ▶ EEPROM 1 KB (ATmega328)
- ▶ Clock Speed 16 MHz

Είδη Arduino

The screenshot displays the Arduino website's product page. The browser's address bar shows the URL arduino.cc/en/Main/Products?from=Main.Hardware. The navigation menu includes links for Buy, Download, Products, Learning, Reference, Support, and Blog, along with LOG IN and SIGN UP options. The main content is organized into four columns: BOARDS (with a 'Compare Specs' link), SHIELDS, KITS, and ACCESSORIES. Each column features images of specific products with their names listed below them.

BOARDS (Compare Specs)	SHIELDS	KITS	ACCESSORIES
 Arduino Uno	 Arduino Leonardo	 Arduino GSM Shield	 The Arduino Starter Kit
 Arduino Due	 Arduino Yún	 Arduino Ethernet Shield	
 Arduino Mega	 Arduino Pro Mini	 Arduino Ethernet Shield	 Arduino Pro Mini

TEI of Athens We x setDaemon - Avai x What is a daemon x How setDaemon x Μετάφραση Google x ict europe - Avai x Icteupe.gr x Arduino - Product x

arduino.cc/en/Main/Products?from=Main.Hardware

Buy Download Products Learning Reference Support Blog LOG IN SIGN UP


 COMING SOON!			
Arduino Tre	Arduino Micro	Arduino WiFi Shield	TFT LCD screen
			
Arduino Robot	Arduino Esplora	Arduino Wireless SD Shield	USB/Serial Light Adapter
			
Arduino Mega ADK	Arduino Ethernet	Arduino USB Host	Arduino ISP

Start | EN | 3:45 μμ


TEI of Athens We x setDaemon - Avol x What is a daemon x How setDaemon v x Μετάφραση Google x Ict europe - Avol x Icteuropa.gr x Arduino - Product x

arduino.cc/en/Main/Products?from=Main.Hardware


Buy Download Products Learning Reference Support Blog LOG IN SIGN UP




Arduino Mega 2560




Arduino Mini




Arduino Motor Shield




Mini USB/Serial Adapter




LilyPad Arduino USB




LilyPad Arduino Simple




Arduino Wireless Proto Shield



LilyPad Arduino



LilyPad Arduino



Arduino Wireless Proto Shield

Start | EN | 3:45 μμ

Shields









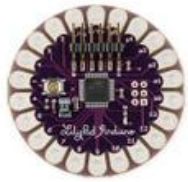

The screenshot shows the Arduino website's product page. The browser's address bar displays 'arduino.cc/en/Main/Products?from=Main.Hardware'. The navigation menu includes 'Buy', 'Download', 'Products', 'Learning', 'Reference', 'Support', 'Blog', 'LOG IN', and 'SIGN UP'. The page is organized into four columns: 'BOARDS (Compare Specs)', 'SHIELDS', 'KITS', and 'ACCESSORIES'. Each column contains images of various Arduino products with their names listed below them.

Category	Product Name
BOARDS (Compare Specs)	Arduino Uno
	Arduino Leonardo
	Arduino Due
SHIELDS	Arduino GSM Shield
	Arduino Ethernet Shield
KITS	The Arduino Starter Kit
ACCESSORIES	(Image of an accessory, possibly a screen or module)

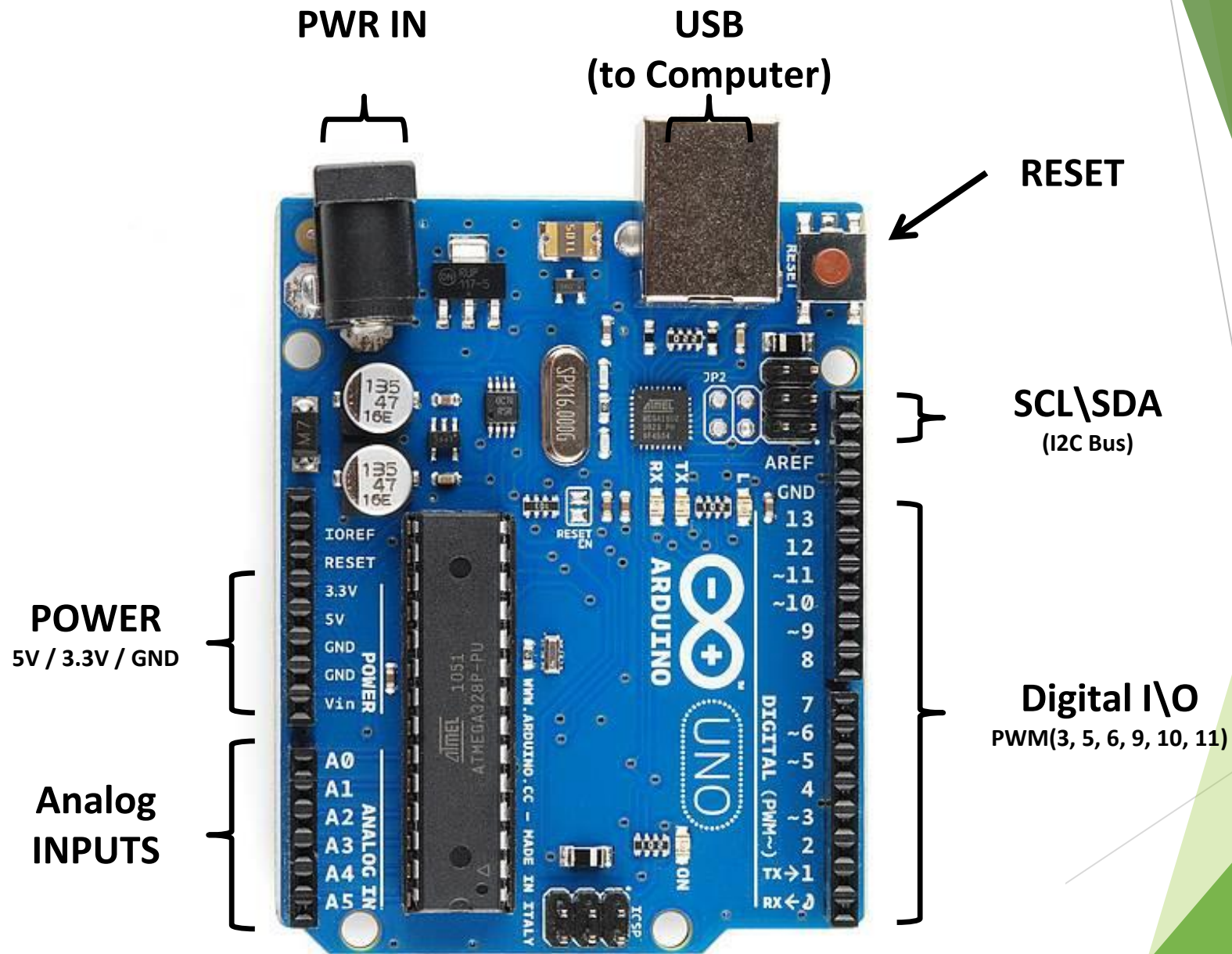
TEI of Athens We x setDaemon - Ava x What is a daemon x How setDaemon v x Μετάφραση Googl x ict europe - Ava x Icteuropa.gr x Arduino - Product x

arduino.cc/en/Main/Products?from=Main..Hardware

Buy Download Products Learning Reference Support Blog LOG IN SIGN UP

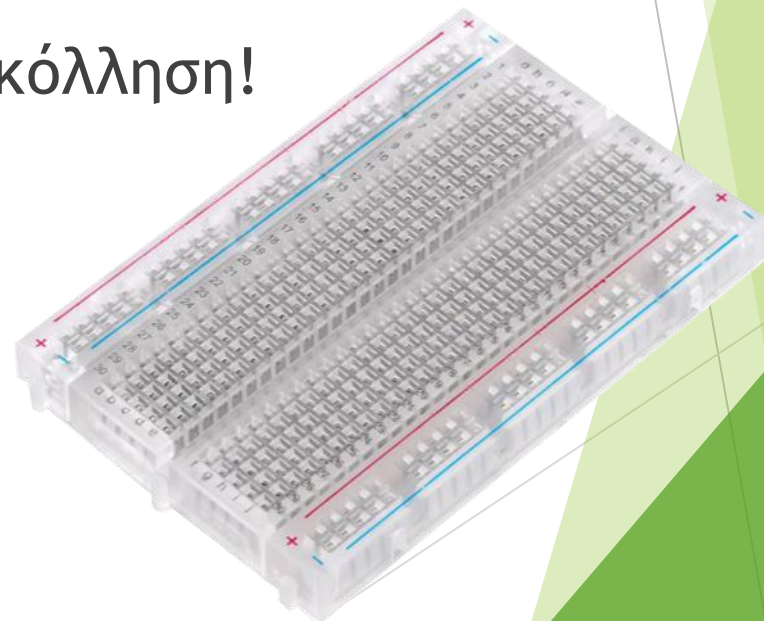
 <p>Arduino Mega 2560</p>	 <p>Arduino Mini</p>	 <p>Arduino Motor Shield</p>	 <p>Mini USB/Serial Adapter</p>
 <p>LilyPad Arduino USB</p>	 <p>LilyPad Arduino Simple</p>	 <p>Arduino Wireless Proto Shield</p>	
 <p>LilyPad Arduino</p>	 <p>LilyPad Arduino</p>		

Start | EN | 3:45 μμ



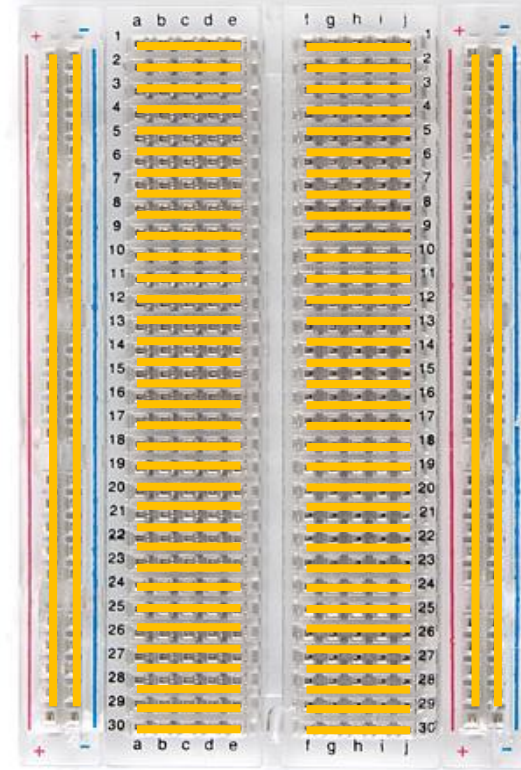
Breadboard

- ❑ Απαραίτητο για την ανάπτυξη κατασκευών.
- ❑ Σημαντικά σημεία:
 - Η χρήση του είναι πιο εύκολη από την κόλληση!
 - Έχει πολλές μικρές τρύπες.
 - Μερικές φορές σπάει...



Breadboard

- ▶ Οι οριζόντιες ομάδες 5 τρυπών είναι συνδεμένες.
- ▶ Οι στήλες της τάσης και της γείωσης συνδέονται κάθετα



Είσοδος και έξοδος

Είσοδος: σήμα που έρχεται στο board.

Buttons

Switches

Light Sensors

Flex Sensors

Humidity Sensors

Temperature Sensors



Έξοδος: Σήμα που φεύγει από το board.

LED

DC motor

servo motor

buzzer

relay

RGB LED



Sensors



Gas Sensor

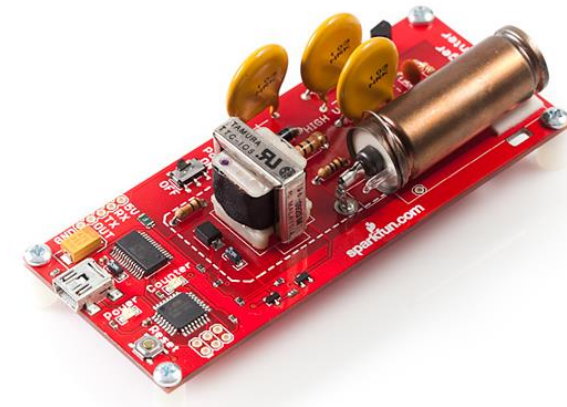
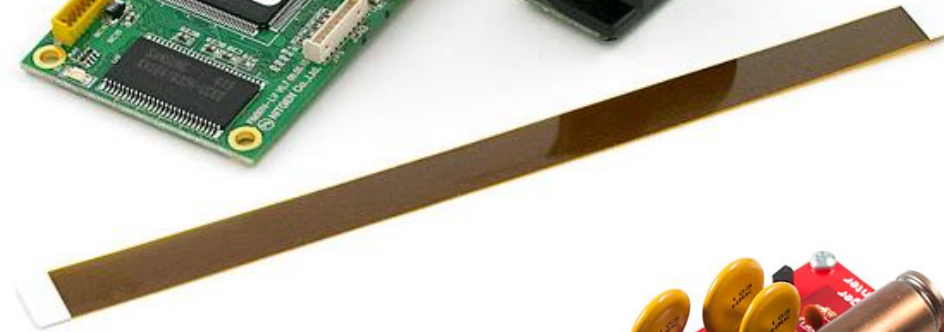
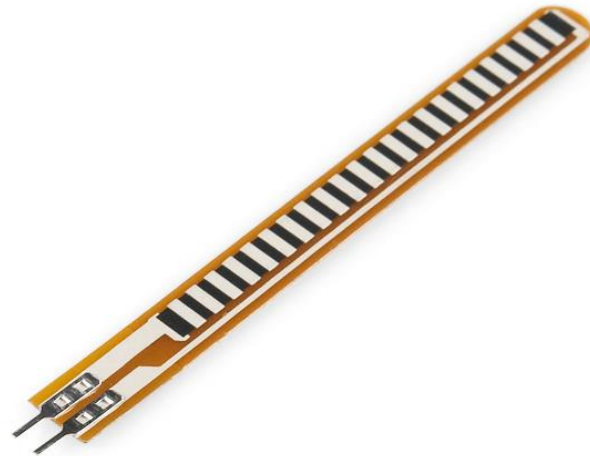


Temp & Humidity

Fingerprint Scanner



Flex Sensor



Geiger Counter

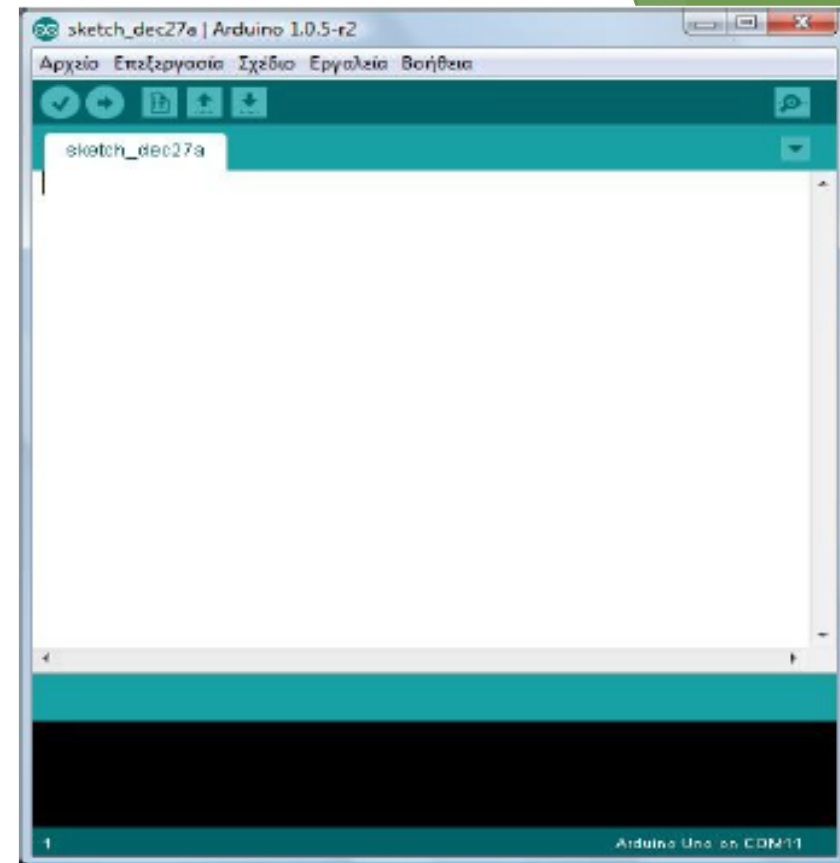
Αισθητήρες

Photo/thermistor, infrared, force sensitive resistor, Hall effect, Piezo, tilt sensor..



Προγραμματισμός Arduino

- ▶ Για να προγραμματίσετε τη μονάδα σας θα χρειαστείτε το περιβάλλον προγραμματισμού Arduino IDE (εικόνα 1). Στο περιβάλλον αυτό γράφετε κώδικα (βασίζεται στη γλώσσα C/C++) τον οποίο μετά μεταγλωττίζετε και μεταφορτώνετε στη μονάδα σας. Το Arduino IDE υπάρχει σε εκδόσεις για Windows, Mac και Linux και μπορείτε να το κατεβάσετε εντελώς δωρεάν από την επίσημη ιστοσελίδα:
- ▶ <http://arduino.cc/en/Main/Software>



Εικόνα 1 – Περιβάλλον προγραμματισμού Arduino IDE

Το περιβάλλον αυτό έχει εξελληνισμένο μενού, καθώς και αρκετά έτοιμα παραδείγματα χρήσης βασικών λειτουργιών (Αρχείο => Παραδείγματα).

Φωτοдиодοι - Leds



- ▶ Οι φωτοдиодοι, ή τα leds όπως έχουν κυριαρχήσει, υπάρχουν σε διάφορα χρώματα και τάσεις λειτουργίας. Ανάλογα με την προβλεπόμενη τάση λειτουργίας του led που έχουμε πρέπει να χρησιμοποιήσουμε μαζί και την κατάλληλη αντίσταση, ώστε να αποφύγουμε κάποια καταστροφή από υπέρταση. Για να βρούμε την αντίσταση που χρειαζόμαστε αρκεί να θυμηθούμε τον τύπο

$$R = (V_{\pi} - V_{\lambda}) / I$$

- ▶ όπου R είναι η αντίσταση που χρειαζόμαστε, V_{π} η παρεχόμενη τάση από την πηγή μας (5V από το Arduino), V_{λ} η τάση λειτουργίας του led και I το ρεύμα λειτουργίας του led. Τυπικά, μια αντίσταση γύρω στα 220 Ω καλύπτει τα περισσότερα led που έχουμε χρησιμοποιήσει.
- ▶ Επίσης, τα leds έχουν πολικότητα, δηλαδή δουλεύουν μόνο αν συνδεθούν στην κατάλληλη φορά ρεύματος. Συνήθως το πόδι που πρέπει να συνδεθεί στη θετική κατεύθυνση (+) είναι πιο μακρύ από το αντίστοιχο για την αρνητική φορά (-). Επίσης, το λαμπάκι από την αρνητική μεριά (-) είναι συνήθως επίπεδο κι όχι στρογγυλό όπως είναι από το άλλο πόδι (+). Μην ανησυχείτε όμως, μια ανάποδη σύνδεση δεν θα το καταστρέψει, απλά δεν θα ανάψει στην ανάποδη φορά (εκτός από την ιδιαίτερη περίπτωση να έχετε δώσει πολύ μεγάλη τάση).

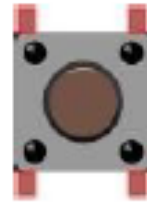
Έγχρωμοι φωτοδίοδοι - RGB Leds

- ▶ Εκτός από τα απλά leds, υπάρχουν και τα RGB leds (εικόνα 9), τα οποία μπορούν να εμφανίσουν οποιοδήποτε χρώμα, βασιζόμενα στο σύστημα RGB. Μπορούμε να τα σκεφτούμε ως τρία led (κόκκινο - Red, πράσινο - Green, μπλε - Blue) σε ένα, με τα οποία δίνοντας χωριστά φωτεινότητα σε κάθε ένα μπορούμε να συνδυάσουμε χρώματα και να παράγουμε οποιοδήποτε χρωματισμό, όπως ακριβώς συμβαίνει στις ψηφιακές οθόνες.



Τα RGB leds έχουν τέσσερα σημεία σύνδεσης - ένα για κάθε χρώμα (τρία σύνολο) κι ένα για την άνοδο ή κάθοδο. Υπάρχουν δύο τύπου, ανόδου (+) και καθόδου (-), ανάλογα με τη σύνδεση που ακολουθούν. Τα καθόδου, που χρησιμοποιούμε στα παραδείγματά μας, συνδέονται με την κάθοδο (-), δηλαδή τη γείωση (GND). Έτσι, συνδέουμε από ένα Pin που θα ελέγχει την τάση που θα δοθεί σε κάθε χρώμα και τη γείωση του Arduino, καταφέροντας να έχουμε απόλυτο έλεγχο σε όλους τους δυνατούς χρωματισμούς.

Κουμπιά - Buttons



- ▶ Στα κυκλώματά μας θα χρειαστεί πολλές φορές να χρησιμοποιήσουμε κουμπιά (εικόνα 10) ώστε να μπορούμε να παρέμβουμε στο κύκλωμα όταν εμείς το θέλουμε, π.χ. για να δώσουμε ρεύμα σε ένα τμήμα του ή να διακόψουμε την τροφοδότηση με ρεύμα ενός άλλου τμήματος.

Ποτενσιόμετρο - Potentiometer

- ▶ Ένα ποτενσιόμετρο μπορεί να περιγραφεί ως μια διάταξη με κύκλωμα μεταβλητής αντίστασης.
- ▶ Γυρίζοντας το χειριστήριο που έχει, αυξάνει ή μειώνει την αντίστασή του, οπότε περνάει λιγότερο ή περισσότερο ρεύμα από αυτό



Ηχείο - Sounder

- ▶ Ένα επίσης απλό εξάρτημα που μπορούμε να συνδέσουμε με το Arduino μας είναι ένα ηχείο.
- ▶ Δουλεύει ακριβώς όπως ένα led, με δύο καλώδια, ένα για την πηγή (5V) κι ένα για τη γείωση (GND), όπως βλέπετε και στην εικόνα 14. Αν του δώσουμε σταθερή τάση (π.χ. 5V) θα μας δώσει σταθερό ήχο, ενώ μεταβλητή τάση (π.χ. αντί στην πηγή (5V), σύνδεσή του με ένα PWM pin στο Arduino και αυξομείωση της τάσης που του παρέχεται) θα μας δώσει μεταβλητό ήχο, δηλαδή ηχητικό εφέ.



Ηχείο (απλό)



Πιεζοηλεκτρικό ηχείο

Φωτοευαίσθητη αντίσταση - Photoresistor (LDR)



- ▶ Υπάρχουν αντιστάσεις για τις οποίες αλλάζουν οι ιδιότητές τους ανάλογα με το πόσο φως πέφτει πάνω τους. Η αντίσταση τους μικραίνει με το φως (όταν βρίσκονται σε περιβάλλον μεγαλύτερης φωτεινότητας) και μεγαλώνει όσο πλησιάζουν σε πιο σκοτεινό περιβάλλον. Έτσι, αν συνδεθούν σε ένα κύκλωμα με σταθερή τάση (π.χ. 5V - GND του Arduino) το ρεύμα που τις διαρρέει θα είναι μεταβαλλόμενο αν είναι μεταβλητή και η φωτεινότητα που πέφτει πάνω τους.

Αισθητήρας υπερήχων - *Ultrasonic sensor*



- ▶ Ο αισθητήρας υπερήχων χρησιμοποιεί δύο διατάξεις, μία για να στείλει ένα υπερηχητικό σήμα κι μια για να το λάβει. Έτσι, από το χρόνο που μεσολαβεί από το να στείλει μέχρι να λάβει το σήμα πίσω μπορούμε να υπολογίσουμε την απόσταση στην οποία μπροστά μας βρίσκεται κάποιο αντικείμενο. Στην εικόνα βλέπουμε έναν κλασσικό τέτοιο αισθητήρα, τον HC-SR04 Ultrasonic Distance Sensor.

Σέρβος - Servos

Ένα ενδιαφέρον εξάρτημα που μπορούμε να διαχειριστούμε μέσω του Arduino είναι το σέρβο, και ειδικότερα θα χρησιμοποιήσουμε το TowerPro SG90. Πρόκειται για μια διάταξη που μπορεί να γυρίζει έναν άξονα από τις 0 μέχρι τις 180 μοίρες. Η διάταξη αυτή έχει εφαρμογή σε κατασκευές που θέλουμε να κινείται κάποιο μέρος ελεγχόμενα. Αν σταθεροποιήσουμε κάπου τη βάση του, μπορούμε να χρησιμοποιήσουμε το σέρβο για να εισάγουμε κίνηση σε εύρος 180 μοιρών στην κατασκευή μας. Το σέρβο συνδέεται με ένα καλώδιο στην πηγή, ένα στη γείωση και το τρίτο σε ένα Pin του Arduino ώστε να μπορούμε να δίνουμε εντολές για το πώς και πόσο θα στρίψει.

Χρειάζεται να εισάγουμε την αντίστοιχη βιβλιοθήκη στο πρόγραμμά μας για να δουλέψει (`#include <Servo.h>`), να ορίσουμε μια μεταβλητή τύπου `servo` (`servo myservo;`) και το pin μέσω του οποίου θα το χειριστούμε.

```
#include <Servo.h>
int servoPin = 13;
Servo myservo;
```

Αντί να ορίσουμε το `servoPin` ως εξόδου στη συνάρτηση `setup`, το συσχετίζουμε με το σέρβο μας στη συνάρτηση `setup()` με την εντολή

```
myservo.attach(servoPin);
```

Το σέρβο έχει τρία καλώδια - ένα (πορτοκαλί) συνδέεται στην πηγή (5V), ένα (καφέ) στη γείωση (GND) κι ένα (κίτρινο) στο pin μέσω του οποίου του στέλνουμε είσοδο (εντολές), χρησιμοποιώντας την εντολή

```
servo.write(num);
```

όπου το `num` είναι ένας αριθμός από 0 έως 179, αντιπροσωπεύοντας τις 180 μοίρες στρέψης του.



Ασπίδες - Shields

- ▶ Το Arduino μπορεί να επεκταθεί με πλακέτες που ονομάζουμε ασπίδες (shields) και οι οποίες έχουν ενσωματωμένα κυκλώματα ώστε να επεκτείνουν με τα υπάρχοντα pins τη λειτουργία του. Για παράδειγμα, υπάρχει WiFi Shield, BlueTooth Shield κτλ, δίνοντας στο Arduino τη λειτουργικότητα που λείπει και το όνομά τους. Τις ασπίδες αυτές μπορούμε να χρησιμοποιήσουμε σε επόμενο στάδιο, έχοντας δηλαδή κατανοήσει τις βασικές λειτουργίες και έχοντας φτάσει σε ένα ικανοποιητικό στάδιο χειρισμού και προγραμματισμού, οπότε δεν θα τις αναλύσουμε εδώ περισσότερο.



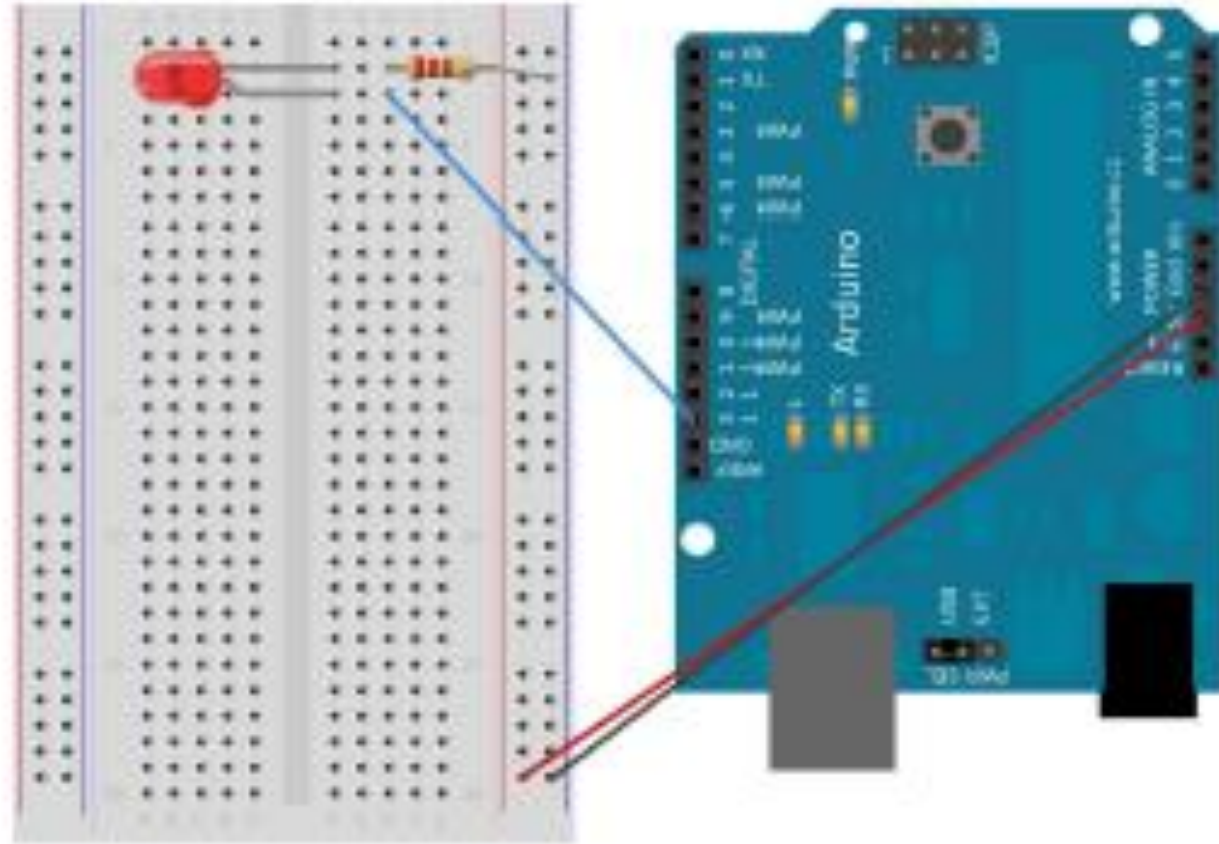
Η μόνη ασπίδα με την οποία θα ασχοληθούμε εδώ, είναι η ArduinoSensor Shield (εικόνα 21) η οποία μας επεκτείνει τα pin που ήδη υπάρχουν σε τριάδες της μορφής (Ground, Voltage, Pin) ώστε να μπορώ για κάθε pin (είτε αναλογικό είτε ψηφιακό) να έχω μαζί και την πηγή (5V) και τη γείωση (GND).

Το πρώτο μας πρόγραμμα

- ▶ Τώρα ας γράψουμε ένα πρόγραμμα για τον έλεγχο της LED. Κάθε πρόγραμμα πρέπει να περιλαμβάνει τουλάχιστον δύο λειτουργίες. Μια συνάρτηση είναι μια σειρά εντολών προγραμματισμού που μπορούν να ονομαστούν.
 - ▶ 1. `setup ()` που καλείται μία φορά κατά την εκκίνηση του προγράμματος.
 - ▶ 2. `loop ()` που καλείται επαναλαμβανόμενα ξανά και ξανά για όσο διάστημα το Arduino έχει ισχύ.
- ▶ Έτσι το μικρότερο έγκυρο πρόγραμμα Arduino (αν και δεν κάνει τίποτα) είναι:
- ▶ `void setup()`
- ▶ `{`
- ▶ `3 }`
- ▶ `4`
- ▶ `5 void loop()`
- ▶ `6 {`
- ▶ `7 }`

- ▶ Στις περισσότερες γλώσσες προγραμματισμού, ξεκινάτε με ένα πρόγραμμα που απλά εκτυπώνει την οθόνη "Hello, World". Το ισοδύναμο στον κόσμο του μικροελεγκτή είναι να ανάβει και να σβήνει το φως. Αυτό είναι το απλούστερο πρόγραμμα που μπορούμε να γράψουμε για να δείξουμε ότι όλα λειτουργούν σωστά.

- ▶ `const int kPinLed = 13;`
- ▶ `void setup()`
- ▶ `{`
- ▶ `pinMode(kPinLed, OUTPUT)`
- ▶ `}`
- ▶ `void loop()`
- ▶ `{`
- ▶ `digitalWrite(kPinLed, HIGH)`
- ▶ `delay(500);`
- ▶ `digitalWrite(kPinLed, LOW)`
- ▶ `delay(500);`
- ▶ `}`



Σας ευχαριστώ

