

ΠΛΗΡΟΦΟΡΙΚΗ ΓΙΑ ΤΗΝ ΤΕΧΝΗ Η' ΕΞΑΜΗΝΟ

ΑΜΑΛΙΑ ΦΩΚΑ
ΕΠΙΚΟΥΡΗ ΚΑΘΗΓΗΤΡΙΑ

Τμήμα Πλαστικών Τεχνών & Επιστημών της Τέχνης

Περιεχόμενο Μαθήματος

2

- Arduino

Τι είναι το Arduino;

3

- μικρή συσκευή (μικροεπεξεργαστής) που συνδέεται με USB στον υπολογιστή
- μια πλατφόρμα hardware και software
- έχει τη δυνατότητα να ελέγξει συσκευές στον φυσικό κόσμο
- δέχονται ερεθίσματα από το περιβάλλον τους (μέσω των αισθητήρων) και αντιδρούν ανάλογα με το πως έχουν προγραμματιστεί.
- open source (ανοιχτό υλικό και λογισμικό)
- υπάρχει μια ολόκληρη κοινότητα που χρησιμοποιεί το Arduino στις κατασκευές άρα υπάρχει μεγάλος όγκος ελεύθερης πληροφορίας



Φιλοσοφία Arduino

4

- Πρωτότυπα (Prototyping)
 - Επιβεβαίωση ότι κάτι που σχεδιάστηκε δουλεύει πριν κατασκευαστεί
 - Αναζήτηση ελεύθερων πρωτοτύπων και προσαρμογή τους

Φιλοσοφία Arduino

5

- Μαστορέματα (Tinkering)
 - ▣ Επαναχρησιμοποίηση υπάρχουσας τεχνολογίας
 - Φτηνά παιχνίδια ή παλιός εξοπλισμός για τη δημιουργία κάτι νέου

Φιλοσοφία Arduino

6

□ Patching

- Δημιουργία σύνθετων συστημάτων από τη σύνδεση πολλών απλών συσκευών
- Στο λογισμικό αναφέρεται στη χρήση εργαλείων οπτικού προγραμματισμού
 - Max, Pure Data, VVVV
 - κουτιά με διάφορες λειτουργικότητες που συνδέονται μεταξύ τους

Φιλοσοφία Arduino

7

□ Circuit Bending

- Δημιουργικό βραχυκύκλωμα ηλεκτρονικών ηχητικών συσκευών χαμηλής τάσης που λειτουργούν με μπαταρίες για τη δημιουργία νέων μουσικών οργάνων
 - Παιχνίδια, μικρά συνθεσάιζερ, πεντάλ εφέ κιθάρας κ.λπ.

Φιλοσοφία Arduino

8

□ Keyboard Hacking

- Το πληκτρολόγιο είναι ακόμα ο βασικός τρόπος αλληλεπίδρασης με έναν υπολογιστή
- Αντικατάσταση πλήκτρων με συσκευές αισθητήρων
 - αν συνδεθεί ένας αισθητήρας ανίχνευσης κίνησης
 - θα πατιέται ένα πλήκτρο κάθε φορά που θα κάποιος περνάει μπροστά από τον υπολογιστή

Φιλοσοφία Arduino

9

- Παλιές Συσκευές
 - ▣ Παλιοί εκτυπωτές, υπολογιστές, μηχανές γραφείου κ.λπ.
 - ▣ χρήση ανταλλακτικών τους

Φιλοσοφία Arduino

10

- Παιχνίδια
 - Τροποποίηση ηλεκτρονικών παιχνιδιών
 - <http://lowtech.propositions.org.uk/>

Φιλοσοφία Arduino

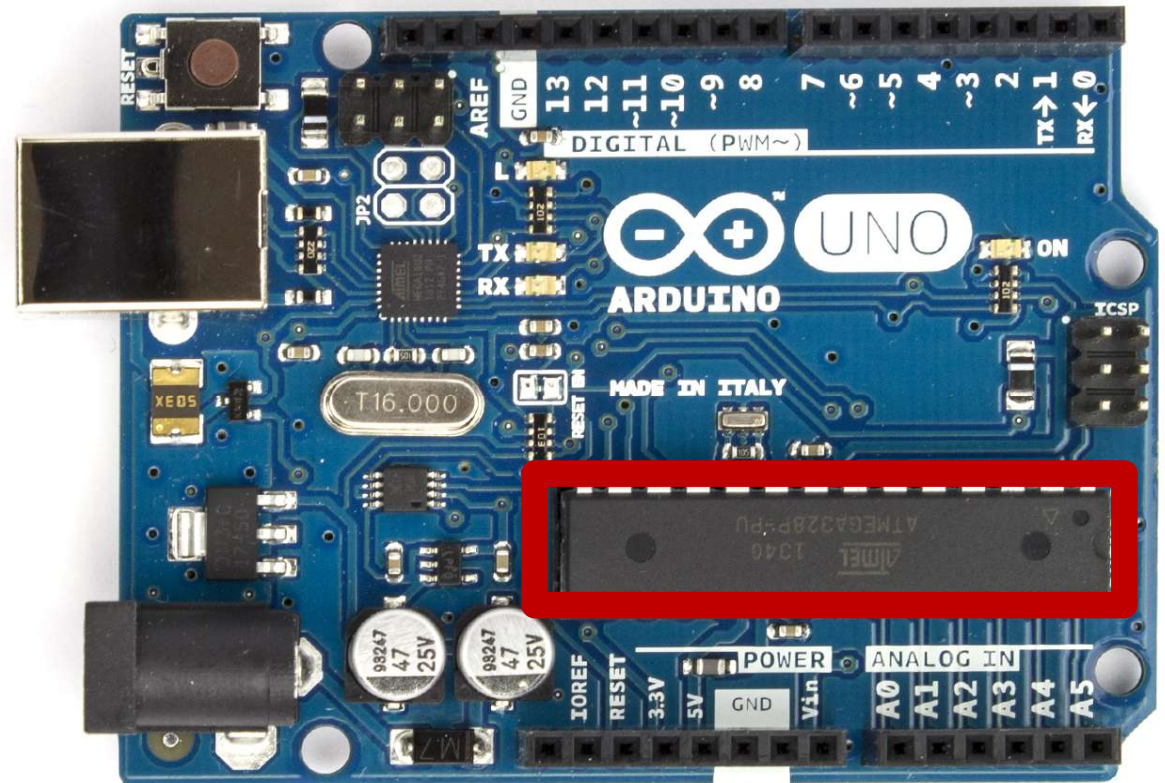
11

- Συνεργασία
 - ▣ Arduino Playground
 - <http://playground.arduino.cc/>

Πλακέτα Arduino

12

- Μικροελεγκτής
 - ▣ ATmega28
 - ▣ 28 pins
 - ▣ η καρδιά της πλακέτας



Πλακέτα Arduino

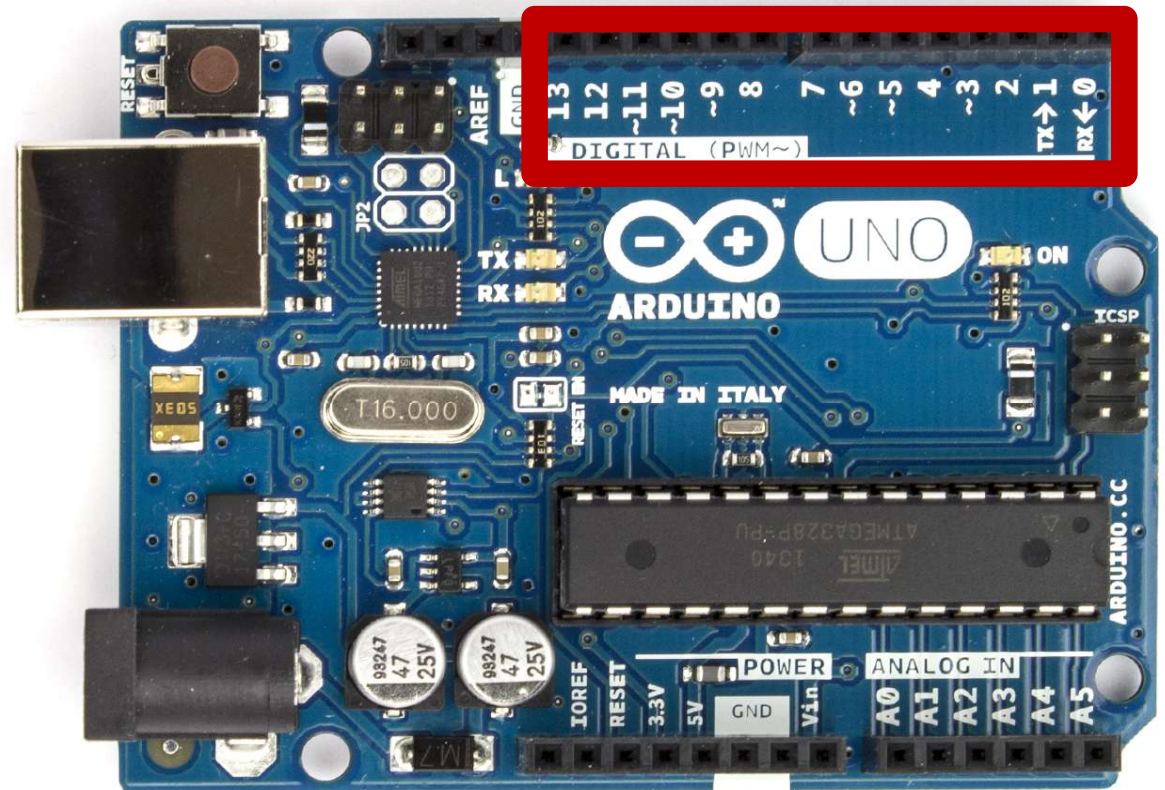
13

- 14 Digital IO pins
 - ▣ pins 0-13
 - ▣ Είσοδοι ή Έξοδοι
 - ▣ Καθορίζεται από τον κώδικα

- Analogue out pins
 - ▣ 3,5,6,9,10,11
 - ▣ Μπορούν να προγραμματιστούν ώστε να δίνουν αναλογική έξοδο

- Pin 0
 - ▣ Serial In (RX)

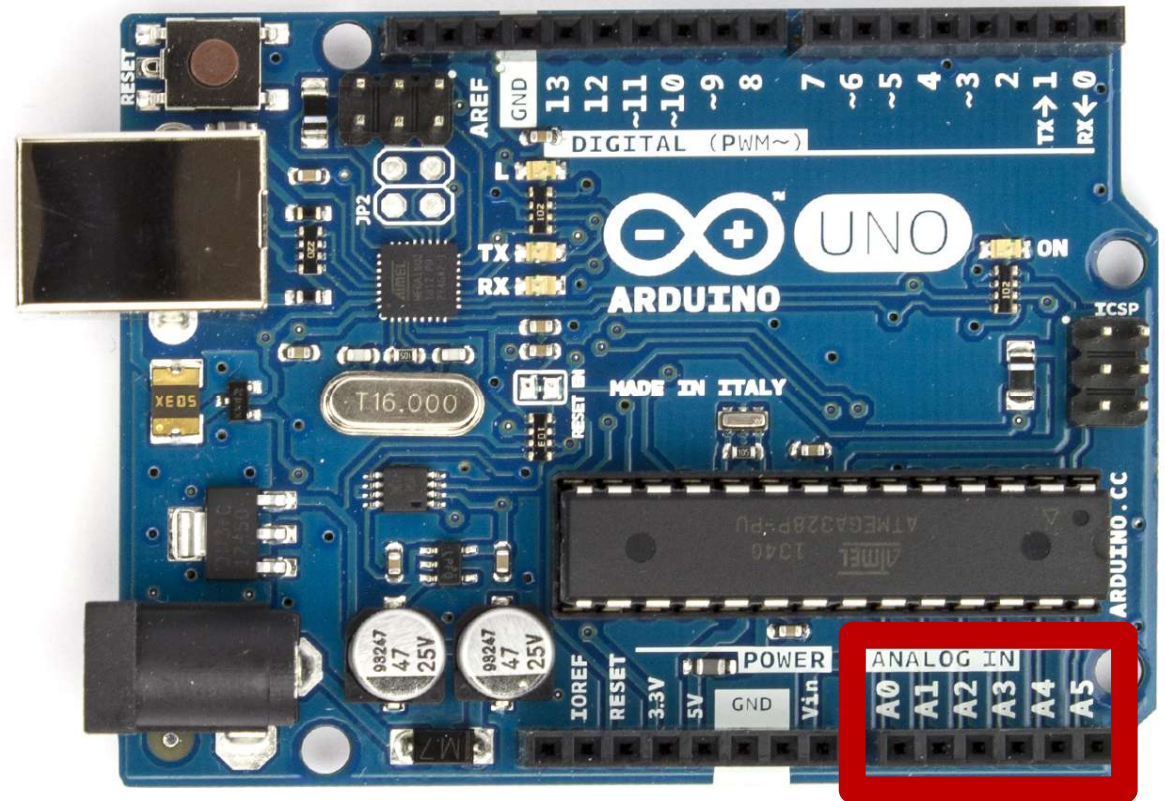
- Pin 1
 - ▣ Serial Out (TX)



Πλακέτα Arduino

14

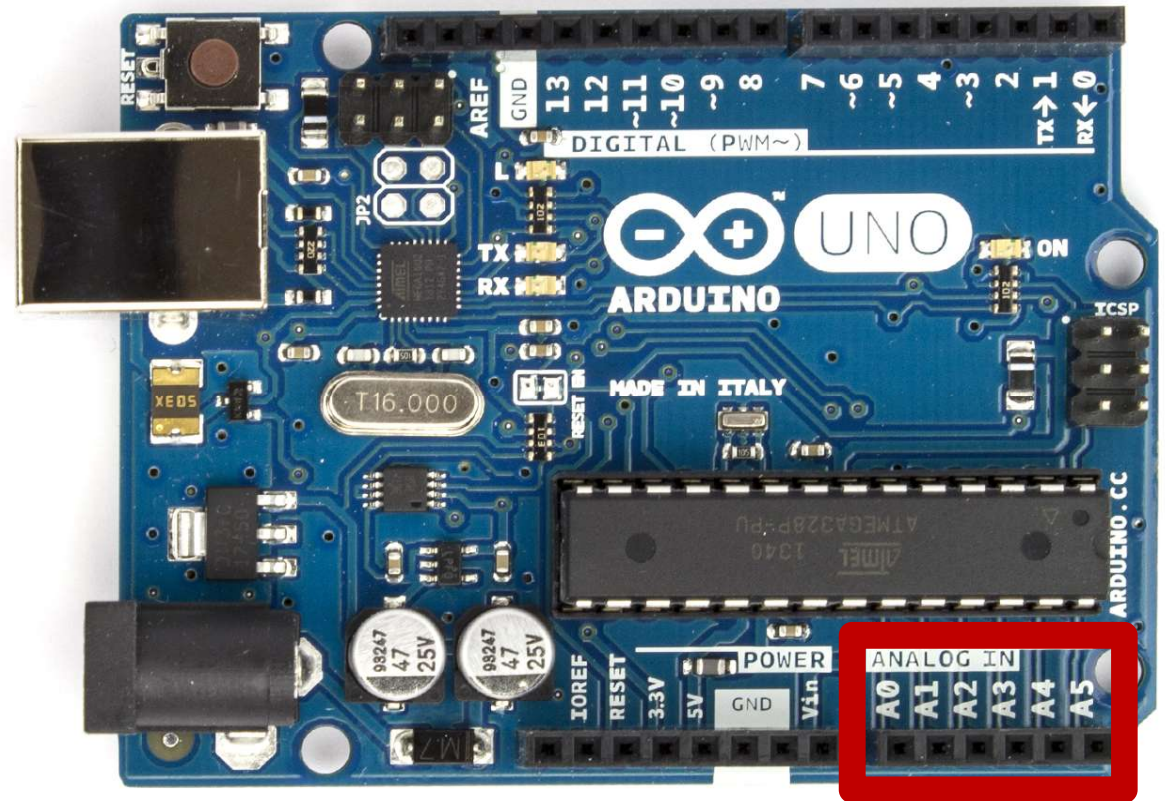
- 6 Analogue In pins
 - ▣ pins 0-5
 - ▣ Δέχονται αναλογικές τιμές
 - Π.χ. τιμές από αισθητήρα
 - ▣ Τις μετατρέπουν σε έναν αριθμό από 0 έως 1023



Πλακέτα Arduino

15

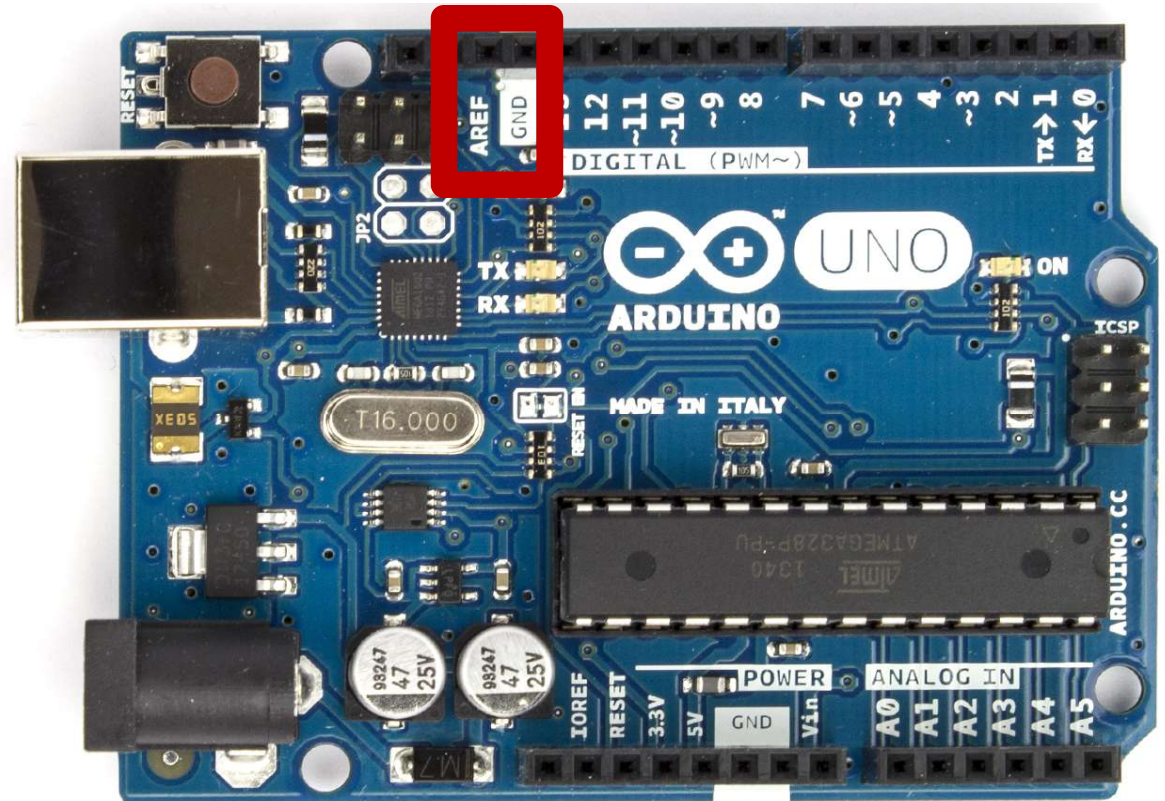
- 6 Analogue In pins
 - ▣ pins 0-5
 - ▣ Δέχονται αναλογικές τιμές
 - Π.χ. τιμές από αισθητήρα
 - ▣ Τις μετατρέπουν σε έναν αριθμό από 0 έως 1023



Πλακέτα Arduino

16

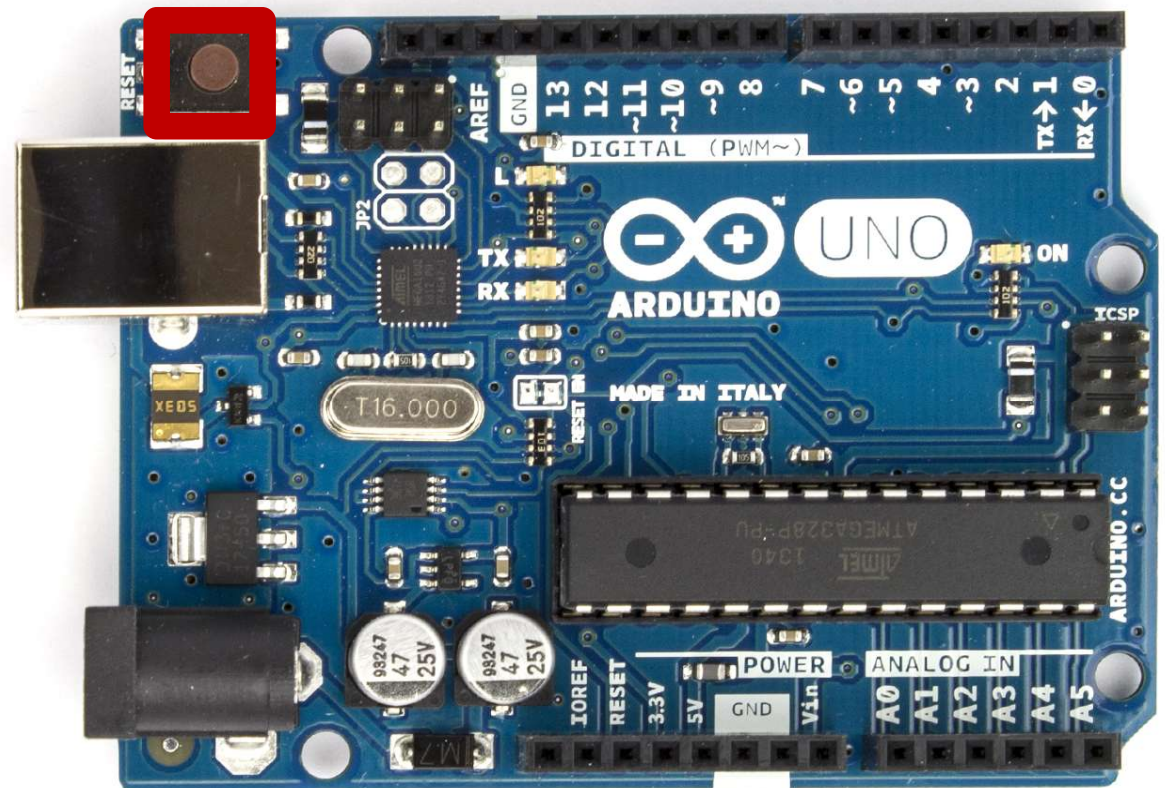
- AREF
 - ▣ Analogue REference
 - ▣ Δυνατότητα μέτρησης τάσης εισόδου με μεγαλύτερη ακρίβεια
 - ▣ Παροχή εισόδου τάσης που καθορίζει την μέγιστη τάση
- GND
 - ▣ GrouND
 - ▣ γείωση



Πλακέτα Arduino

17

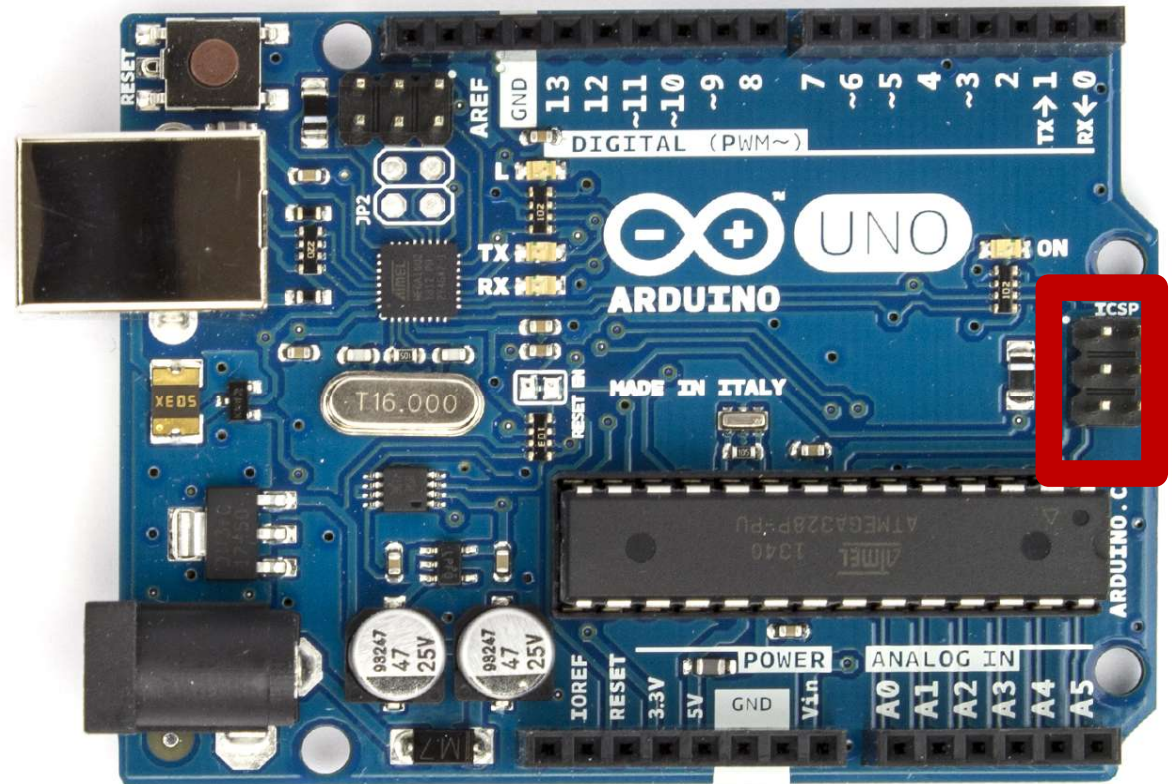
- Reset Button
 - ▣ Κουμπί επανεκκίνησης



Πλακέτα Arduino

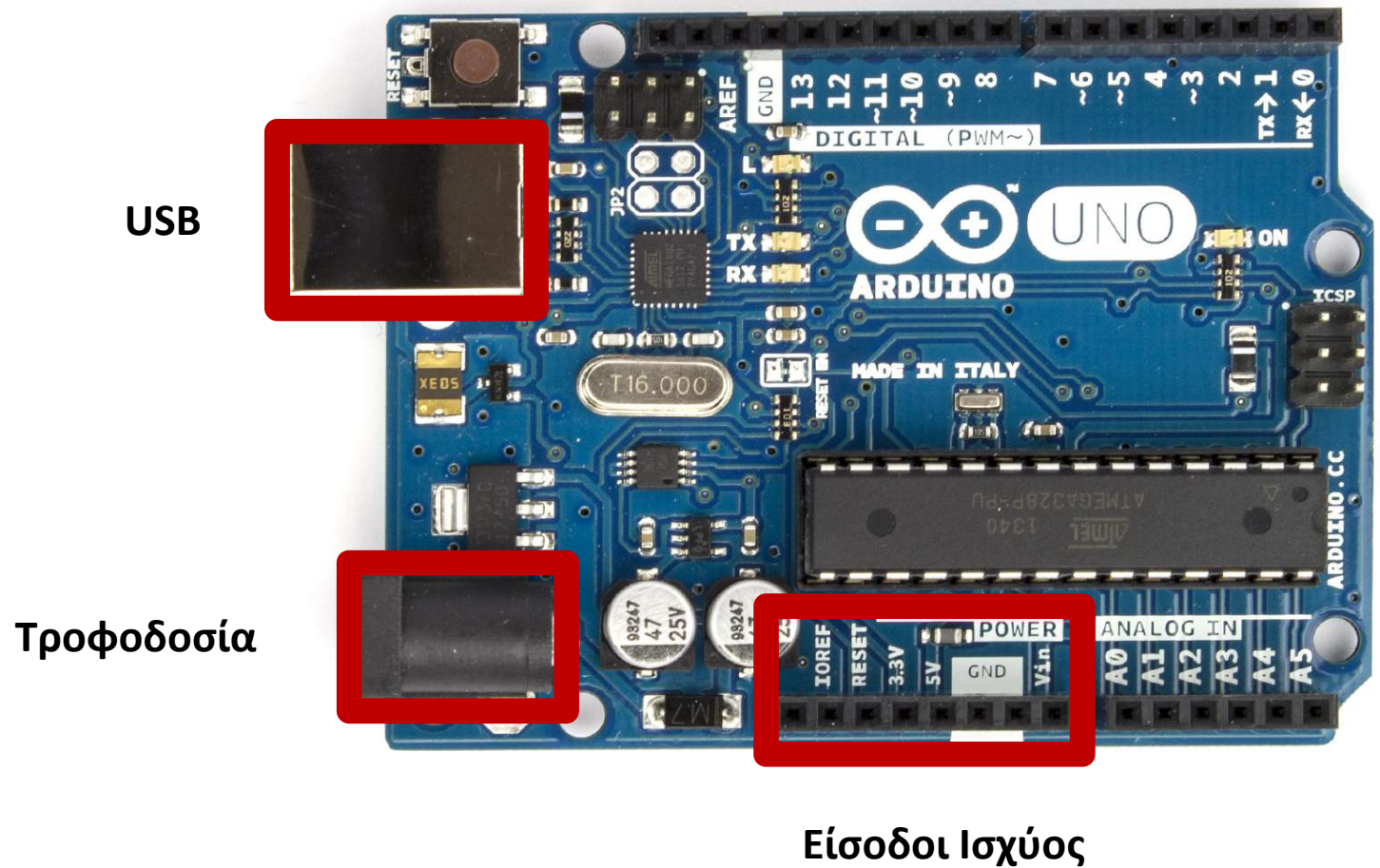
18

- ICSP
 - ▣ In Circuit Serial Programmer
 - ▣ Δυνατότητα προγραμματισμού του μικροελεγκτή απευθείας σε γλώσσα μηχανής



Πλακέτα Arduino

19



Λογισμικό Προγραμματισμού

20

- Σύνδεση πλακέτας με USB σε υπολογιστή
- Συγγραφή κώδικα προγράμματος
- Ανέβασμα προγράμματος στην πλακέτα μέσω USB και επανεκκίνηση της.
- Η πλακέτα θα εκτελεί τον κώδικα που γράψατε

Λογισμικό Προγραμματισμού

21

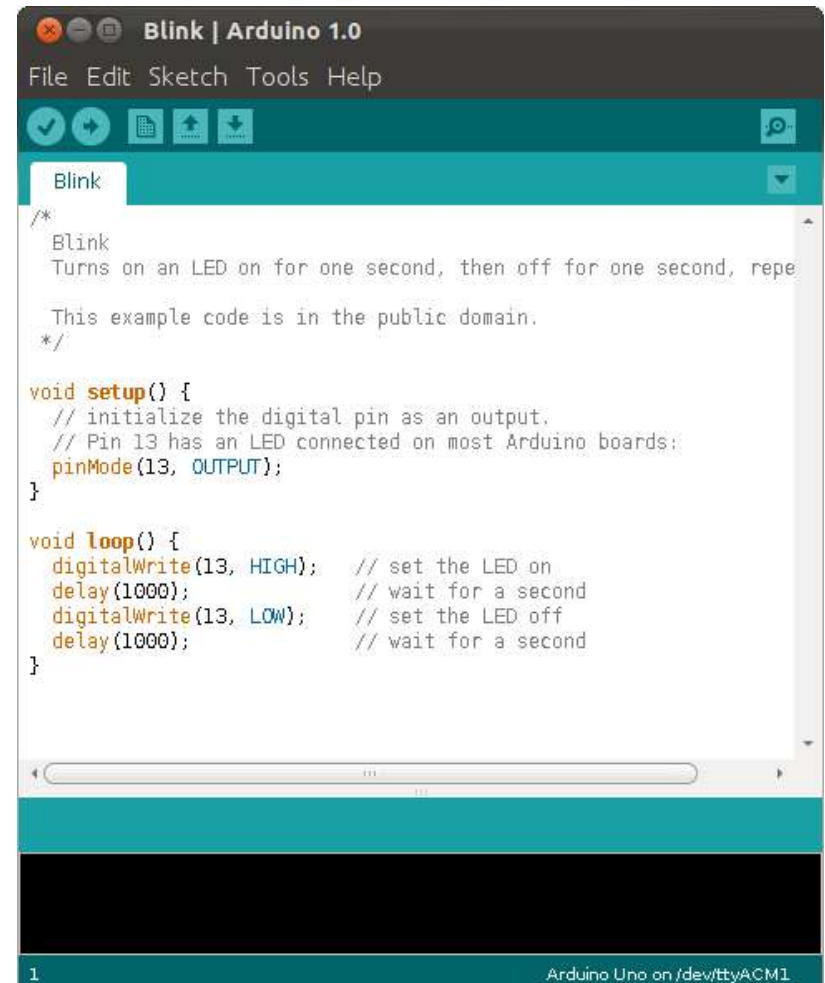
- Σύνδεση πλακέτας με USB σε υπολογιστή
- Συγγραφή κώδικα προγράμματος
- Ανέβασμα προγράμματος στην πλακέτα μέσω USB και επανεκκίνηση της.
- Η πλακέτα θα εκτελεί τον κώδικα που γράψατε

Λογισμικό Προγραμματισμού

22

- Ο κώδικας μπορεί να γραφεί στο περιβάλλον προγραμματισμού (IDE) Arduino

- ▣ <http://www.arduino.cc/en/Main/Software>



The screenshot shows the Arduino IDE interface with the 'Blink' sketch open. The code is as follows:

```
Blink | Arduino 1.0
File Edit Sketch Tools Help

Blink
/*
 * Blink
 * Turns on an LED on for one second, then off for one second, repeatedly.
 *
 * This example code is in the public domain.
 */

void setup() {
  // initialize the digital pin as an output.
  // Pin 13 has an LED connected on most Arduino boards:
  pinMode(13, OUTPUT);
}

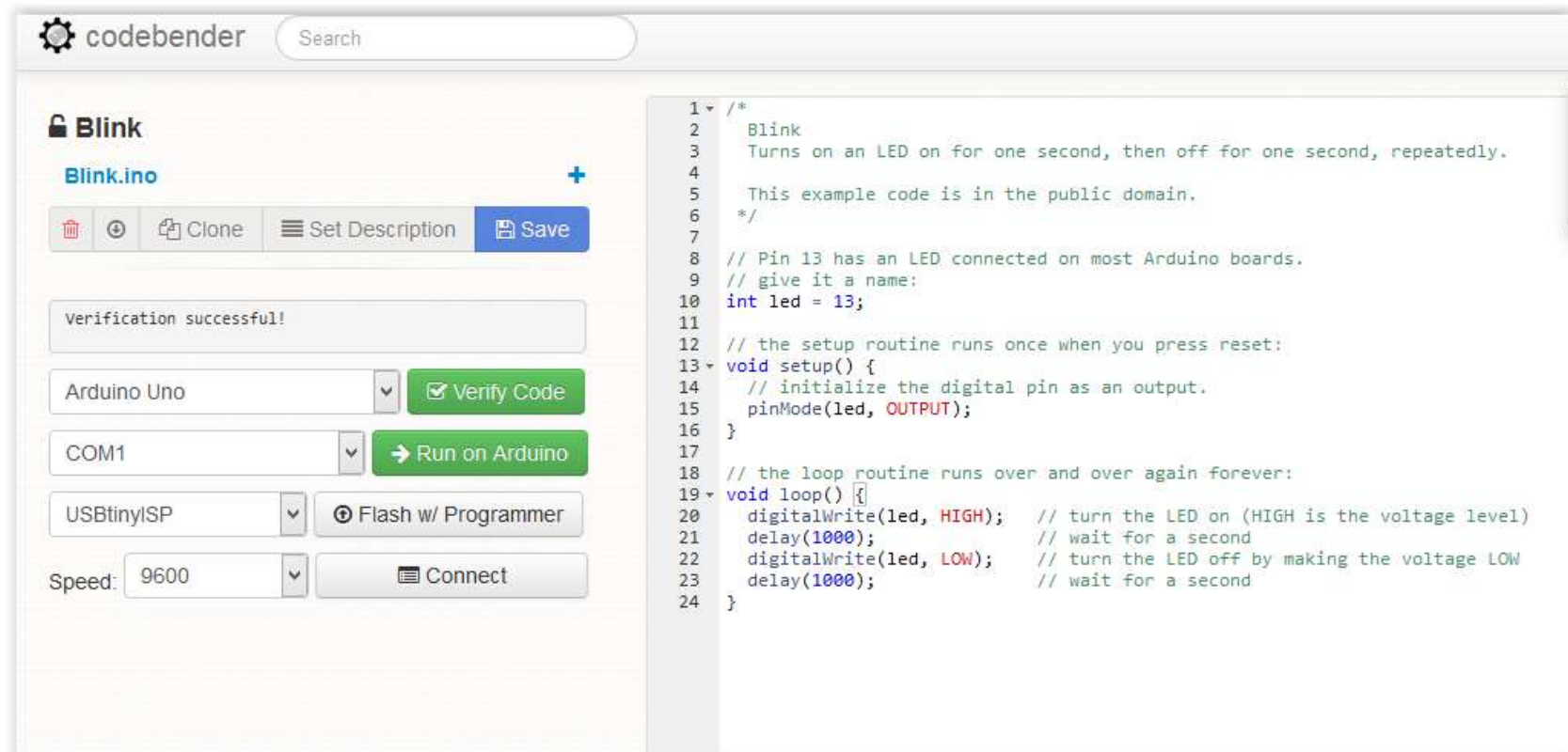
void loop() {
  digitalWrite(13, HIGH); // set the LED on
  delay(1000);             // wait for a second
  digitalWrite(13, LOW);  // set the LED off
  delay(1000);            // wait for a second
}

1 Arduino Uno on /dev/ttyACM1
```

Λογισμικό Προγραμματισμού

23

- Εναλλακτικά μπορεί να χρησιμοποιηθούν άλλα περιβάλλοντα προγραμματισμού
 - <https://codebender.cc/>
 - web-based



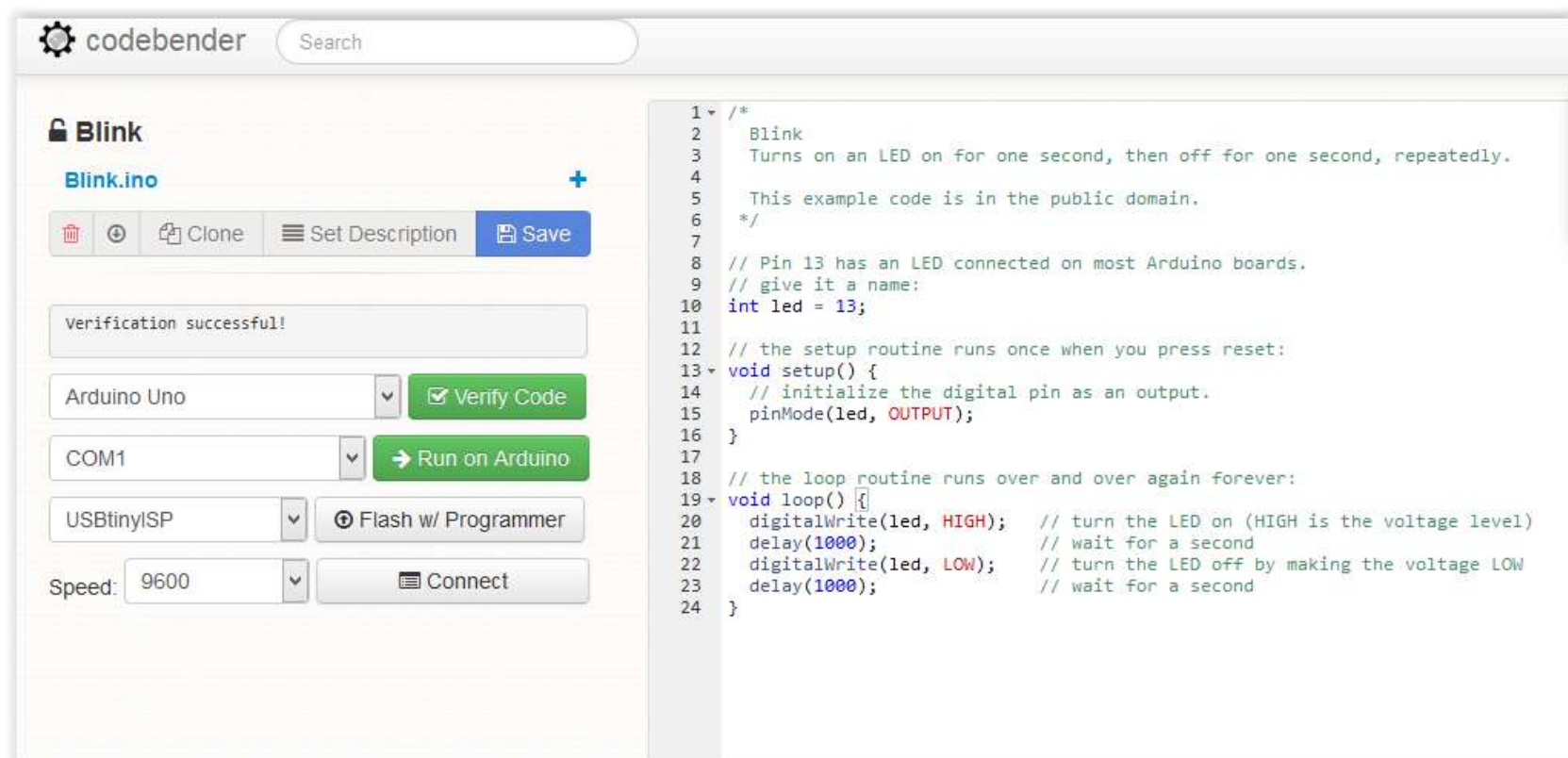
The screenshot displays the CodeBender web-based IDE interface. On the left, the project name is "Blink" with the file "Blink.ino". Below this, there are buttons for "Clone", "Set Description", and "Save". A message box indicates "verification successful!". The hardware configuration is set to "Arduino Uno" with a baud rate of "9600". There are buttons for "Verify Code", "Run on Arduino", "Flash w/ Programmer", and "Connect". On the right, the code for the Blink.ino file is shown, including comments and the setup and loop functions.

```
1 /*
2  Blink
3  Turns on an LED on for one second, then off for one second, repeatedly.
4
5  This example code is in the public domain.
6  */
7
8  // Pin 13 has an LED connected on most Arduino boards.
9  // give it a name:
10 int led = 13;
11
12 // the setup routine runs once when you press reset:
13 void setup() {
14   // initialize the digital pin as an output.
15   pinMode(led, OUTPUT);
16 }
17
18 // the loop routine runs over and over again forever:
19 void loop() {
20   digitalWrite(led, HIGH); // turn the LED on (HIGH is the voltage level)
21   delay(1000); // wait for a second
22   digitalWrite(led, LOW); // turn the LED off by making the voltage LOW
23   delay(1000); // wait for a second
24 }
```

Λογισμικό Προγραμματισμού

24

- Εναλλακτικά μπορεί να χρησιμοποιηθούν άλλα περιβάλλοντα προγραμματισμού
 - <https://codebender.cc/>
 - web-based



The screenshot displays the CodeBender web-based IDE interface. On the left, the file name "Blink.ino" is shown, along with a "verification successful!" message. Below this, there are dropdown menus for selecting the board ("Arduino Uno"), port ("COM1"), and programmer ("USBtinyISP"). A "Speed" dropdown is set to "9600". Action buttons include "Verify Code", "Run on Arduino", "Flash w/ Programmer", and "Connect". On the right, the Arduino C++ code for the Blink sketch is displayed, including comments and the setup and loop functions.

```
1 /*
2  Blink
3  Turns on an LED on for one second, then off for one second, repeatedly.
4
5  This example code is in the public domain.
6  */
7
8  // Pin 13 has an LED connected on most Arduino boards.
9  // give it a name:
10 int led = 13;
11
12 // the setup routine runs once when you press reset:
13 void setup() {
14   // initialize the digital pin as an output.
15   pinMode(led, OUTPUT);
16 }
17
18 // the loop routine runs over and over again forever:
19 void loop() {
20   digitalWrite(led, HIGH); // turn the LED on (HIGH is the voltage level)
21   delay(1000); // wait for a second
22   digitalWrite(led, LOW); // turn the LED off by making the voltage LOW
23   delay(1000); // wait for a second
24 }
```


Αν δεν έχω πλακέτα Arduino;

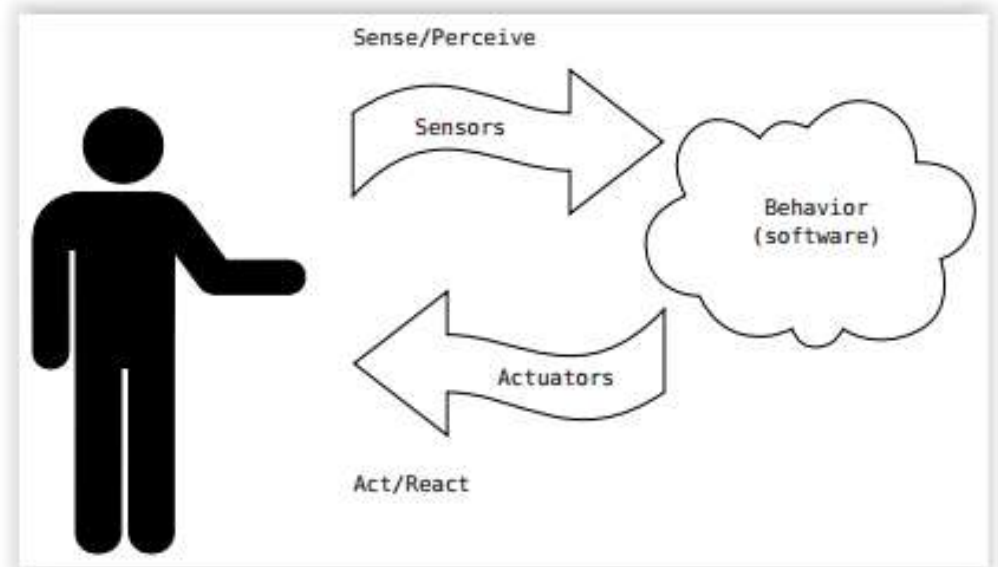
25

- Προσομοιωτής Arduino
- <http://123d.circuits.io/>
- web-based

Ξεκινώντας με το Arduino

26

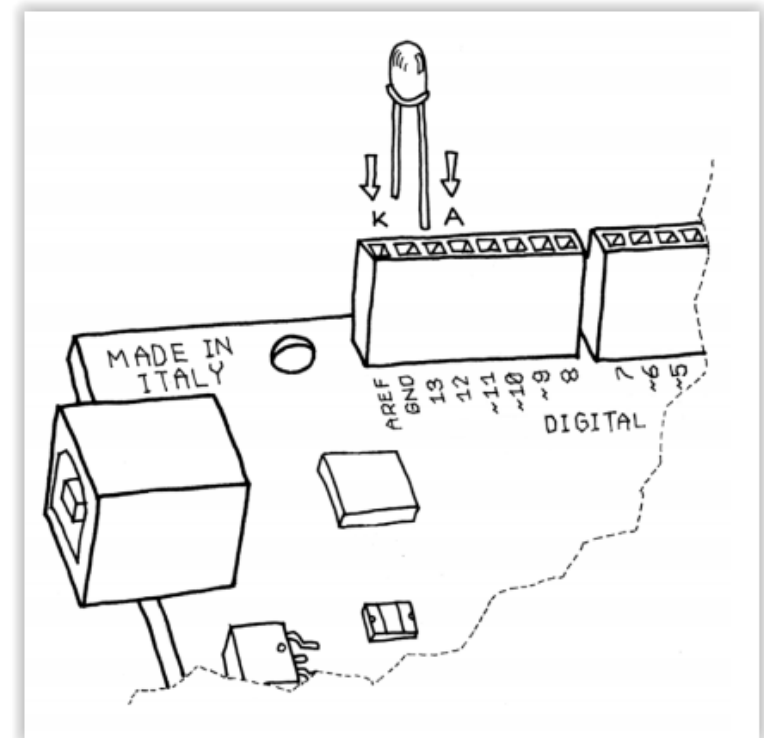
- Κατασκευή και Προγραμματισμός μιας αλληλεπιδραστικής συσκευής
- Αλληλεπιδραστική Συσκευή
 - ▣ Αίσθηση του περιβάλλοντος με αισθητήρες
 - ▣ Επεξεργασία της πληροφορίας από τους αισθητήρες
 - ▣ Προγραμματισμός συμπεριφοράς
 - ▣ Καθορίζει την αλληλεπίδραση της συσκευής



1^ο Παράδειγμα - Blink

27

- Ένα LED που αναβοσβήνει
- Σύνδεση LED στην πλακέτα
 - A – άνοδος (θετικό)
 - Pin 13 – ψηφιακή έξοδος
 - K – κάθοδος (αρνητικό)
 - GND – γείωση



1^ο Παράδειγμα - Blink

28

- Κώδικας που καθορίζει τι πρέπει να κάνει ο μικροελεγκτής με το LED που συνδέσαμε

```
1 // Pin 13 has an LED connected on most Arduino boards.
2 // give it a name:
3 int led = 13;
4
5 // the setup routine runs once when you press reset:
6 void setup() {
7   // initialize the digital pin as an output.
8   pinMode(led, OUTPUT);
9 }
10
11 // the loop routine runs over and over again forever:
12 void loop() {
13   digitalWrite(led, HIGH); // turn the LED on (HIGH is the voltage level)
14   delay(1000); // wait for a second
15   digitalWrite(led, LOW); // turn the LED off by making the voltage LOW
16   delay(1000); // wait for a second
17 }
```

2^ο Παράδειγμα – Χρήση διακόπτη

29

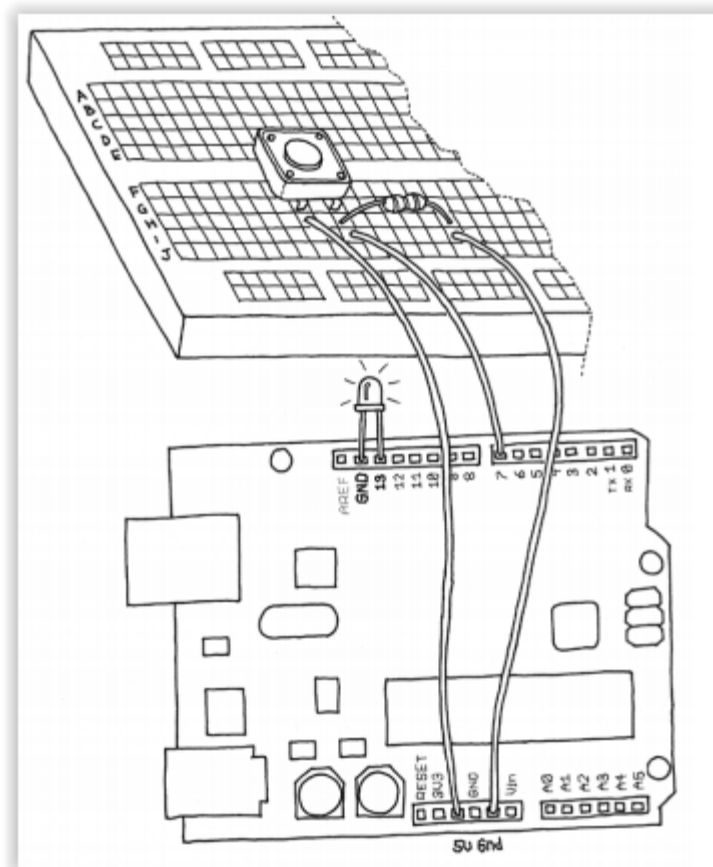
```
// Example 02: Turn on LED while the button is pressed

const int LED = 13; // the pin for the LED
const int BUTTON = 7; // the input pin where the
                    // pushbutton is connected
int val = 0; // val will be used to store the state
            // of the input pin

void setup() {
  pinMode(LED, OUTPUT); // tell Arduino LED is an output
  pinMode(BUTTON, INPUT); // and BUTTON is an input
}

void loop(){
  val = digitalRead(BUTTON); // read input value and store it

  // check whether the input is HIGH (button pressed)
  if (val == HIGH) {
    digitalWrite(LED, HIGH); // turn LED ON
  } else {
    digitalWrite(LED, LOW);
  }
}
```



Παραδείγματα

30

- Εικαστικά έργα με Arduino
 - ▣ <http://www.creativeapplications.net/tag/arduino/>