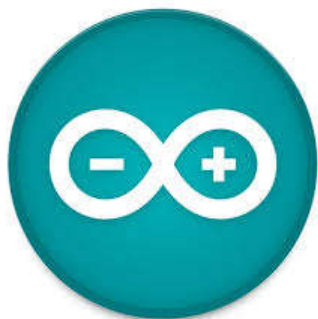


Ε.Κ ΡΕΘΥΜΝΟΥ  
ΜΑΙΟΣ 2014

# ΣΕΜΙΝΑΡΙΟ ARDUINO

Τσαλμπούρης Γεώργιος ΠΕ17.04  
gtsalmpouris@gmail.com  
tsal81@yahoo.com



# ΕΙΣΑΓΩΓΗ

## Τι είναι ο μικροελεγκτής;

- Είναι ένα ολοκληρωμένο κύκλωμα (IC) το οποίο προγραμματίζεται για να εκτελεί μία συγκεκριμένη εργασία.
- Μπορούμε να πούμε με απλά λόγια ότι είναι ένας <<μικρός υπολογιστής>>.

# ΕΙΣΑΓΩΓΗ



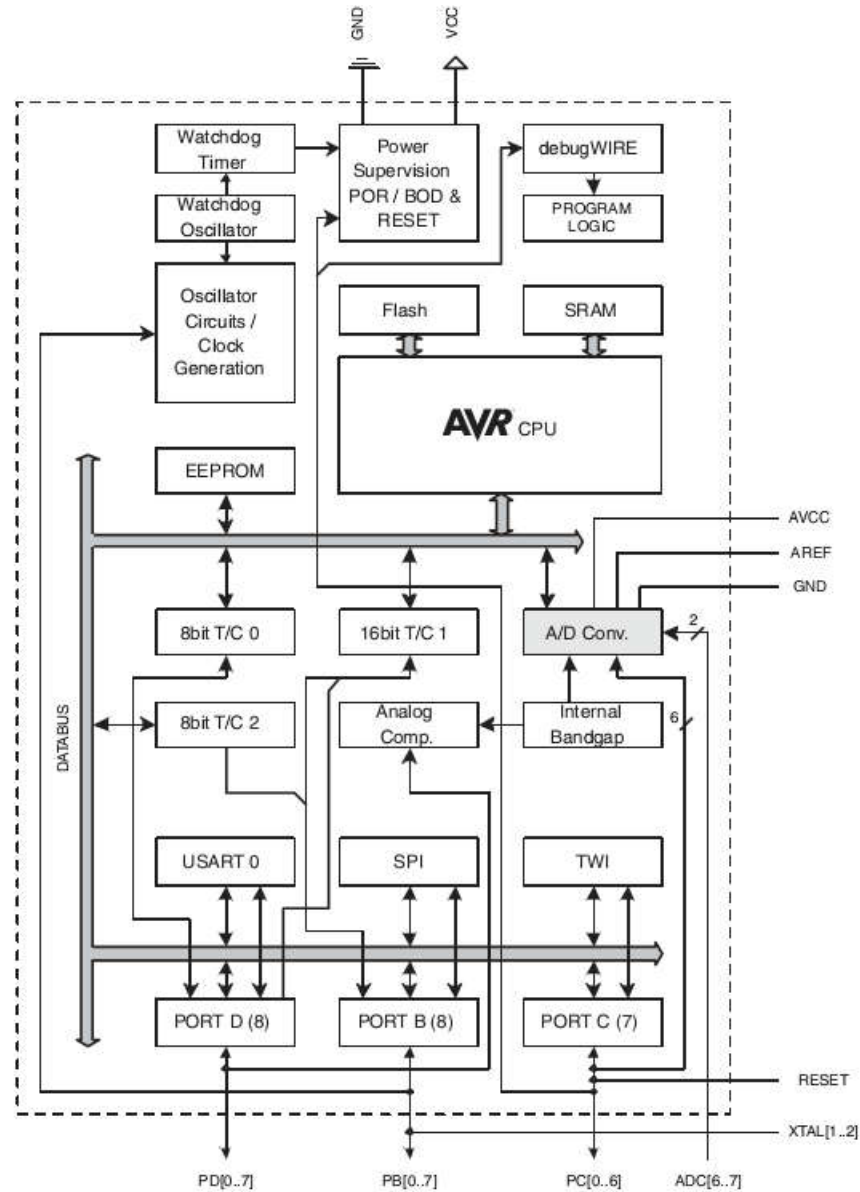
Μικροελεγκτές της εταιρείας Atmel

# ΕΙΣΑΓΩΓΗ

Γνωστές εταιρείες στον χώρο των μΕ:

- Atmel (με τους AVR)
- Microchip (με τους PIC)
- Motorola κτλ

# ΕΙΣΑΓΩΓΗ (Block diagram ATmega328P)



# Μικροελεγκτής vs Μικροεπεξεργαστή

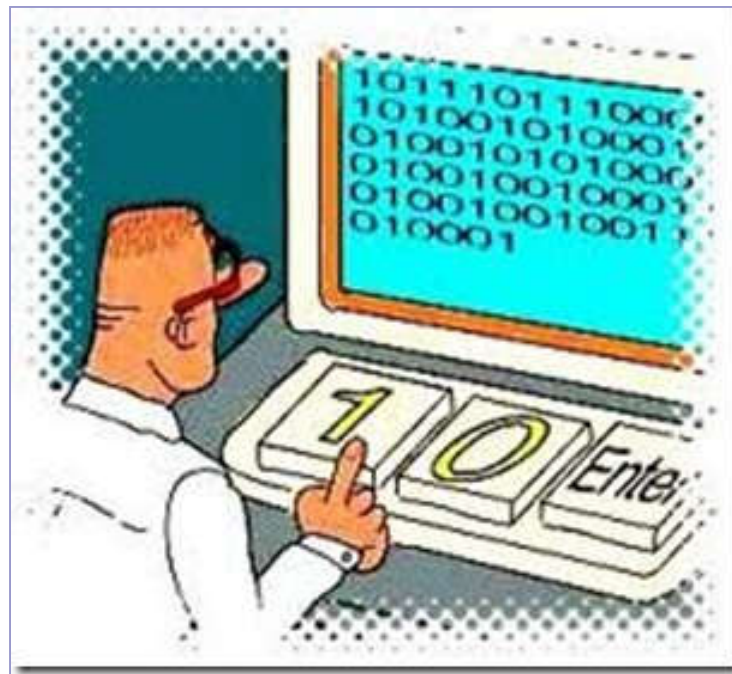
	Μικροελεγκτής	Μικροεπεξεργαστής
Ταχύτητα	Μικρή	Μεγάλη
Εξωτερικά Εξαρτήματα	Λίγα	Πολλά
Κόστος	Μικρό	Μεγάλο
Κατανάλωση ενέργειας	Μικρή	Μεγάλη

# ΤΡΟΠΟΙ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ

- 1) Γλώσσα Μηχανής (Machine Language)
- 2) Συμβολική Γλώσσα (Assembly Language)
- 3) Γλώσσα Ανώτερου Επιπέδου (High Level Language)

# ΤΡΟΠΟΙ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ

## 1) Γλώσσα Μηχανής





# High Level vs Assembly

```
main:
TRISA = 0x00      ' Configure pins as outputs

While TRUE

PORTA = 0x00      ' Turn PORTA LEDs OFF
Delay_ms(1000)    ' 1 second delay
PORTA = 0xFF      ' Turn PORTA LEDs ON
Delay_ms(1000)    ' 1 second delay

wend              ' Endless loop
end.
```

Program written in Basic

```
BSF     STATUS, 5
BCF     STATUS, 6
CLRF    TRISA
```

```
L_main2:
BCF     STATUS, 5
CLRF    PORTA
```

```
MOVLW   11
MOVWF   R11
MOVLW   38
MOVWF   R12
MOVLW   93
MOVWF   R13
```

```
L_main6:
DECFSZ  R13, 1
GOTO    L_main6
DECFSZ  R12, 1
GOTO    L_main6
DECFSZ  R11, 1
GOTO    L_main6
NOP
NOP
```

```
MOVLW   255
MOVWF   PORTA
```

```
MOVLW   11
MOVWF   R11
MOVLW   38
MOVWF   R12
MOVLW   93
MOVWF   R13
```

```
L_main7:
DECFSZ  R13, 1
GOTO    L_main7
DECFSZ  R12, 1
GOTO    L_main7
DECFSZ  R11, 1
GOTO    L_main7
NOP
NOP
GOTO    L_main2
```

```
wend    ' Endless loop
GOTO    $+0
```

The same program compiled into assembly code. As can be seen, each Basic command is broken into several assembly instructions during the process of compiling.

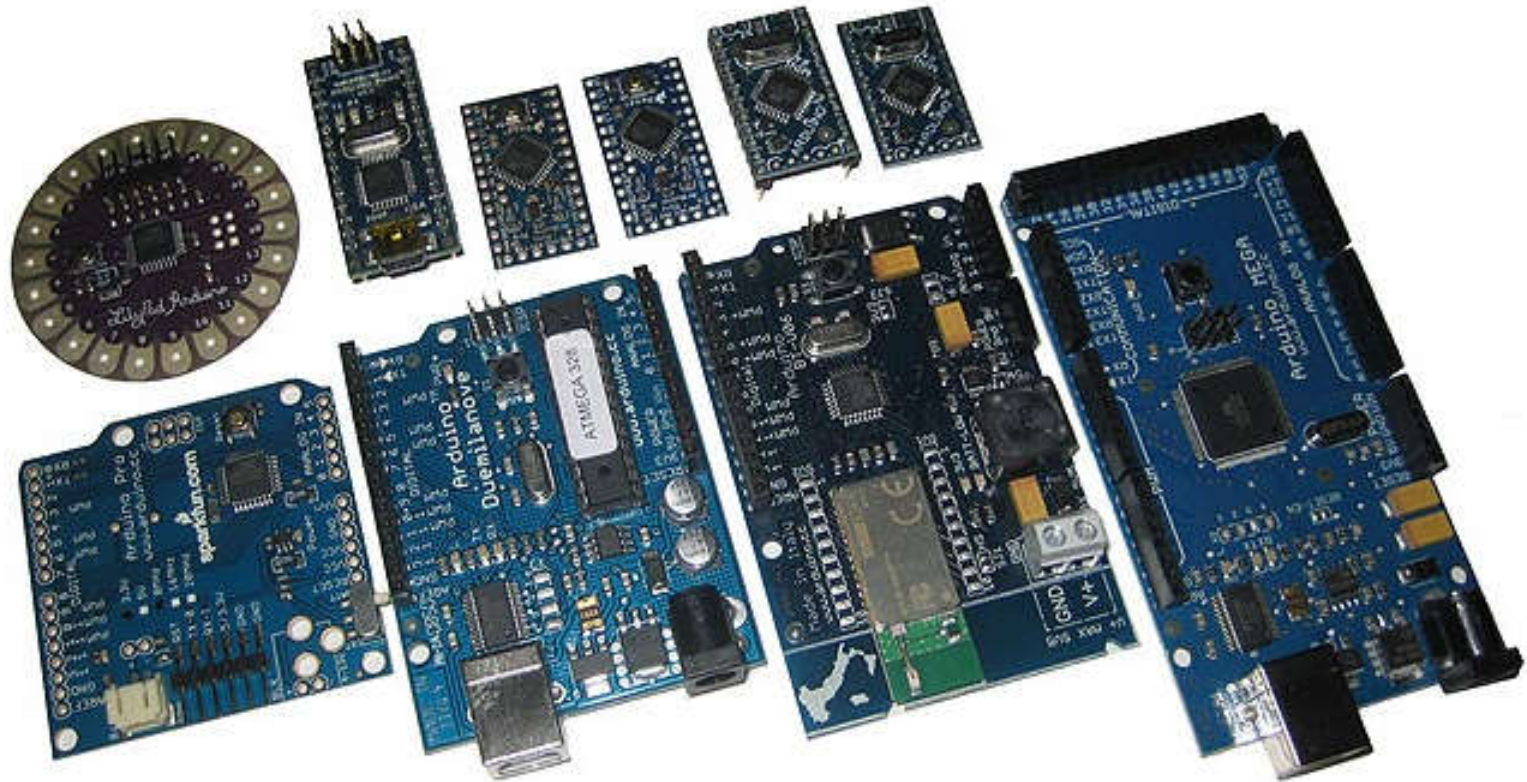
# Εισαγωγή στον Arduino

## Τι είναι ο Arduino;

Ο Arduino είναι μία *αναπτυξιακή πλατφόρμα* ελεύθερου κώδικα που διαθέτει μικροελεγκτή της εταιρείας Atmel, η οποία προγραμματίζεται στη γλώσσα *Wiring* (βασισμένη στη γλώσσα C).



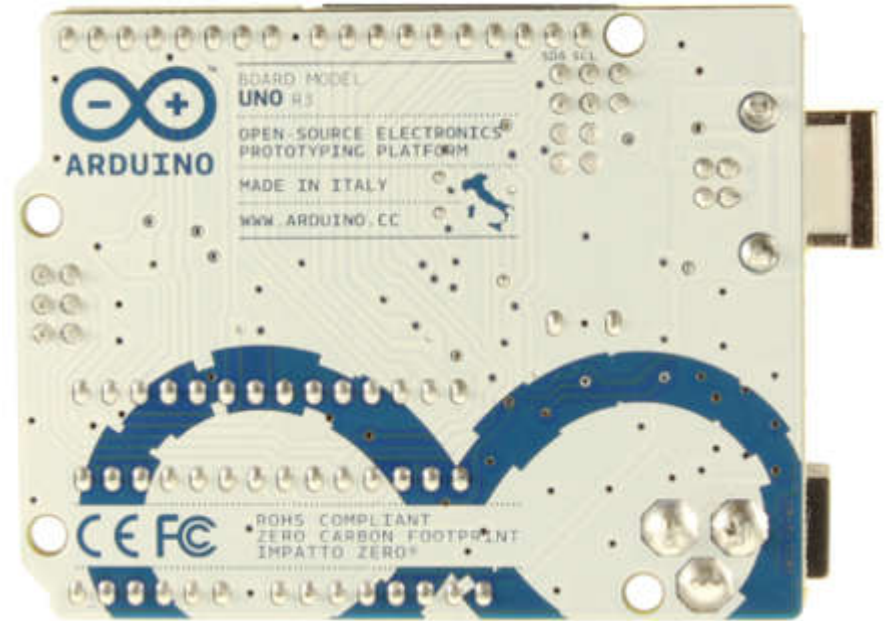
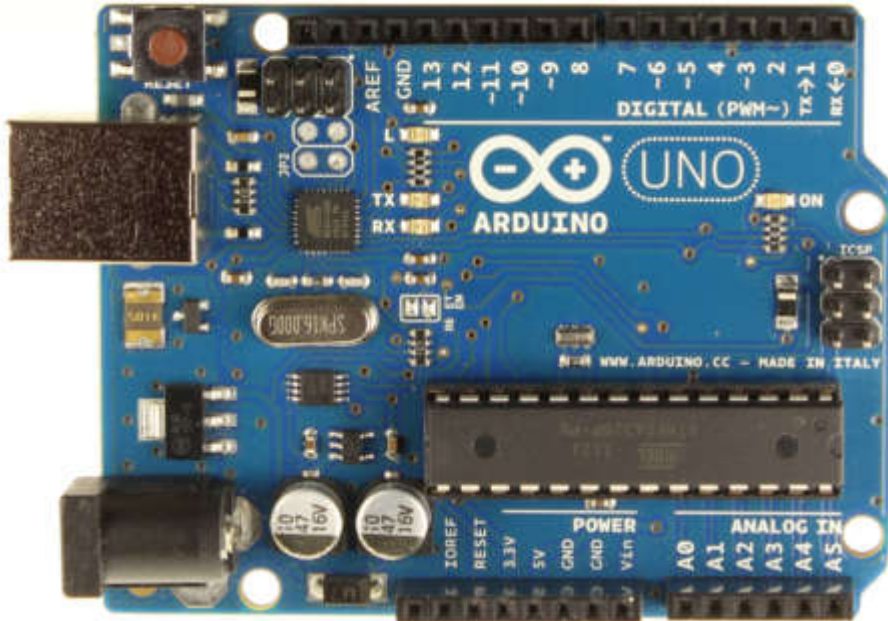
# Arduino Family



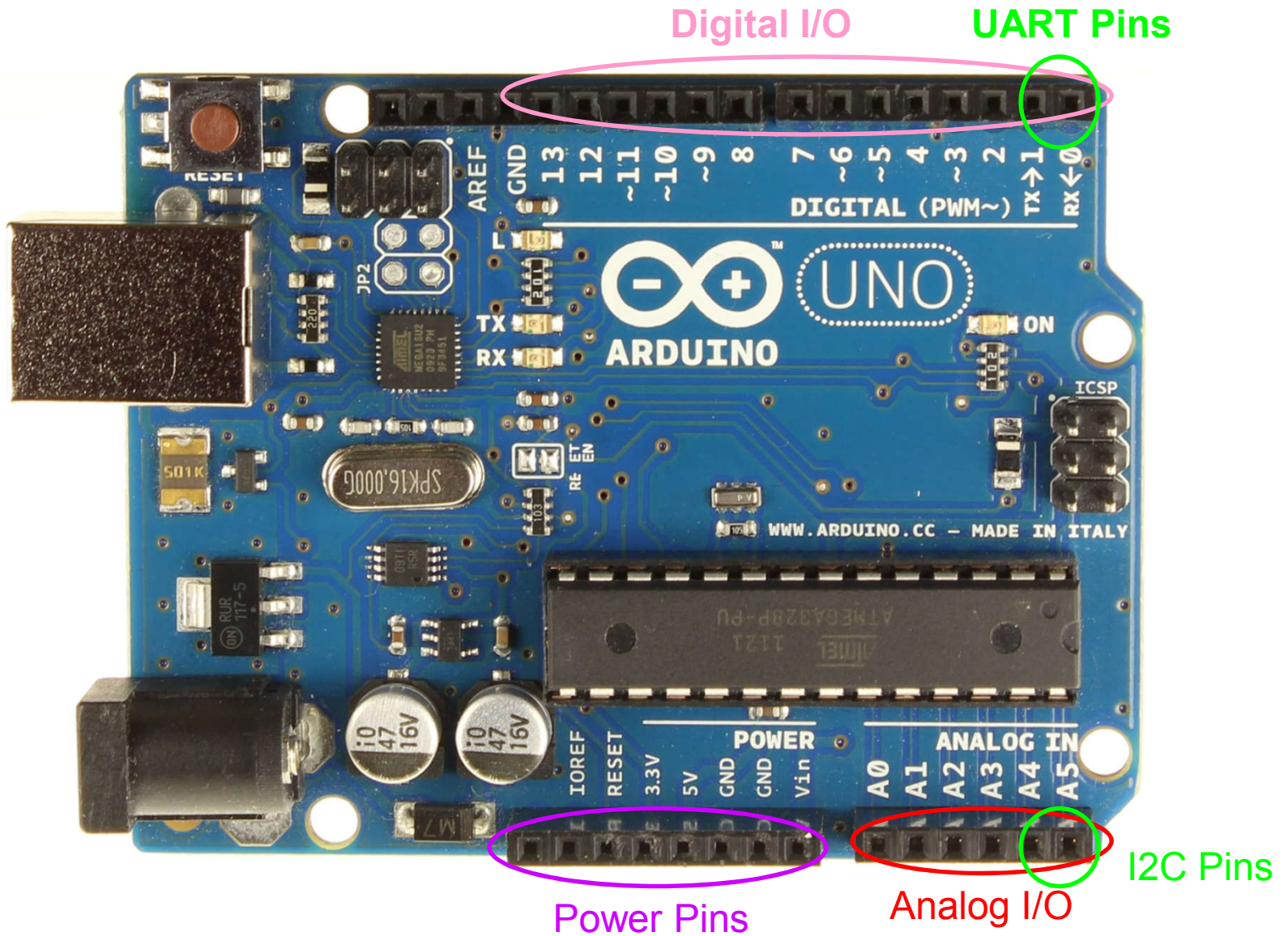
# Γιατί Arduino;

- Πολύ φιλικό λογισμικό προγραμματισμού (κατάλληλο για εκπαιδευτικούς σκοπούς, αρχάριους στον προγραμματισμό)
- Χαμηλό κόστος απόκτησης πλακέτας εφαρμογών
- Δεν απαιτείται εξωτερικός προγραμματιστής
- Πολλές διαθέσιμες πληροφορίες
- Πολλές διαθέσιμες βιβλιοθήκες και έτοιμος κώδικας

# Arduino Uno



# Arduino Uno

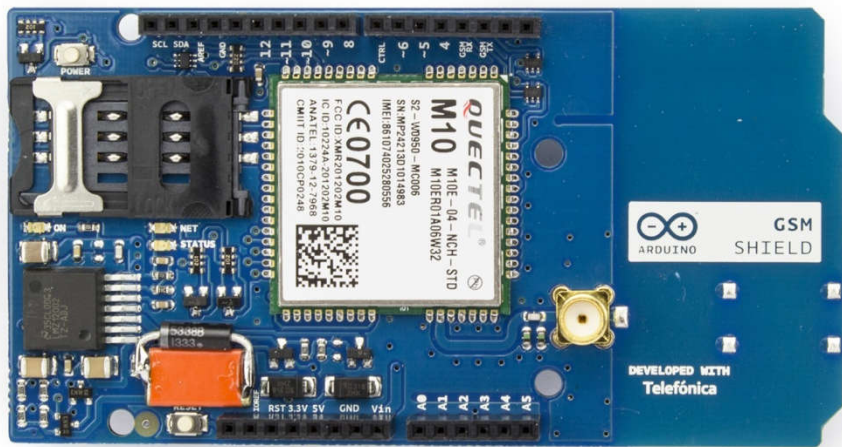


# Τεχνικά Χαρακτηριστικά

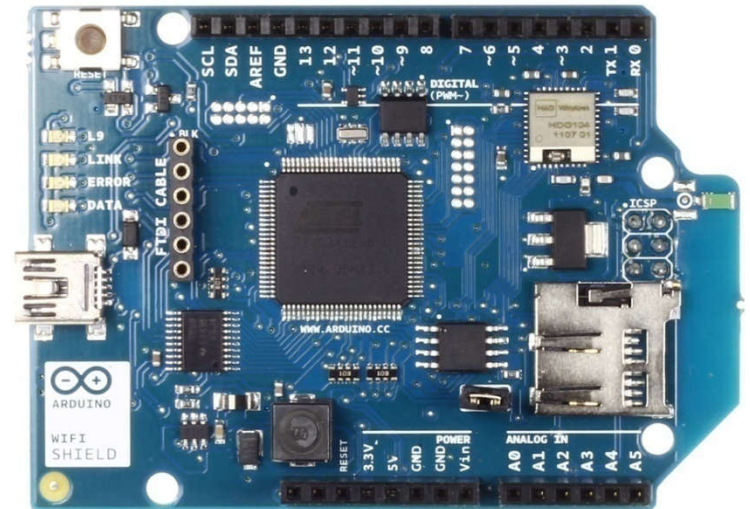
■ Microcontroller	ATmega328
■ Operating Voltage	5V
■ Input Voltage (recommended)	7-12V
■ Input Voltage (limits)	6-20V
■ Digital I/O Pins	14 (of which 6 provide PWM output)
■ Analog Input Pins	6
■ DC Current per I/O Pin	40 mA
■ DC Current total for I/O	200 mA
■ Flash Memory	32 KB (ATmega328)
■ EEPROM	1 KB (ATmega328)
■ SRAM	2 KB (ATmega328)
■ Clock Speed	16 MHz
■ ADC	10bit

# Arduino Shields

- Τα Shields είναι ολοκληρωμένες πλακέτες που κουμπώνουν στον Arduino δίνοντας του επιπλέον δυνατότητες.



Gsm Shield



Wifi Shield



# Πηγές Πληροφορίας

- [www.arduino.cc](http://www.arduino.cc)
- <http://learn.adafruit.com/>
- <http://learning.grobotronics.com/>
- <https://tronixstuff.com/tutorials/>

# Λίγα Λόγια για τη Γλώσσα C

- Στη γλώσσα C δουλεύουμε με **συναρτήσεις**.

**Συνάρτηση** είναι ένα αυτοδύναμο κομμάτι κώδικα σε ένα πρόγραμμα, το οποίο έχει γραφτεί για να εκτελεί μία συγκεκριμένη εργασία και το οποίο μπορεί να χρησιμοποιηθεί ξανά στο ίδιο πρόγραμμα.

# Λίγα Λόγια για τη Γλώσσα C

- Η γλώσσα C είναι μία λιτή γλώσσα με λίγες δεσμευμένες λέξεις.
- Αυτό που την κάνει ισχυρή είναι οι συναρτήσεις βιβλιοθήκης που την συνοδεύουν.

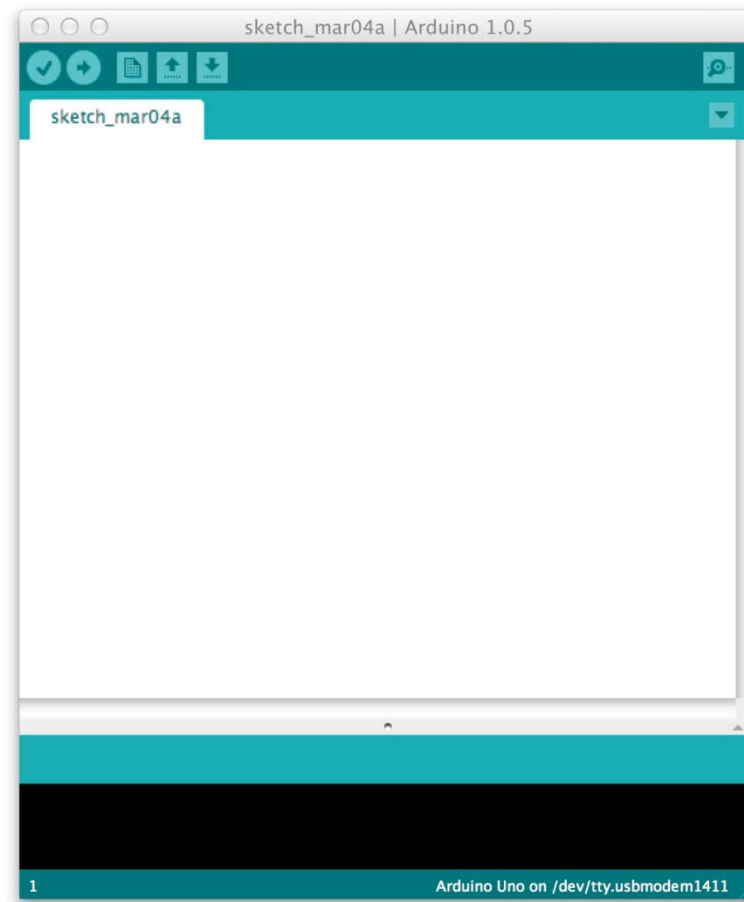
# Λίγα Λόγια για τη Γλώσσα C

```
int mo()  
{  
    int a=3;  
    int b=2;  
    int c=(a+b)/2;  
}
```

# Βασικοί Τύποι Δεδομένων στη C

Τύπος Δεδομένων	Τιμές	Δέσμευση Μνήμης
Boolean	True or False	1 byte
Byte	0 έως 255	1 byte
Integer	-32768 έως 32767	2 byte
Float	-3.4028235E38 έως 3.4028235E38	4 bytes

# Ξεκίνημα-Εγκατάσταση Λογισμικού



# Δομή Κώδικα

1) Σχόλια και Δήλωση Μεταβλητών (Προαιρετικά)

2) `void setup ()`

{

κώδικας παραμετροποίησης

}

3) `void loop ()`

{

κώδικας που εκτελείται επαναλαμβανόμενα

}

# Δομή Κώδικα

```
/*  
 Turns on an LED on for one second, then off for one second, repeatedly  
*/
```

```
// Pin 13 has an LED connected on most Arduino boards.
```

```
int led = 13;
```

```
void setup() {  
  pinMode(led, OUTPUT);  
}
```

```
void loop() {  
  digitalWrite(led, HIGH);  
  delay(1000);  
  digitalWrite(led, LOW);  
  delay(1000);  
}
```

Σχόλια

Δήλωση Μεταβλητών

Συνάρτηση Setup

Συνάρτηση Loop



# Βασικές Εντολές

- `pinMode ()`

Παραμετροποιεί ένα `pin` ώστε να λειτουργεί ως είσοδος ή έξοδος

## Σύνταξη

`pinMode` (pin number, `INPUT` ή `OUTPUT`)

ή

`pinMode` (όνομα μεταβλητής, `INPUT` ή `OUTPUT`)

Πχ `pinMode` (13, `OUTPUT`) ;

ή

`pinMode` (sensor1, `INPUT` )

# Βασικές Εντολές

## ■ digitalWrite ()

Θέτει σε λογικό **HIGH** ή **LOW** ένα digital pin

### Σύνταξη

**digitalWrite** (pin number, **HIGH** ή **LOW**)

ή

**digitalWrite** (όνομα μεταβλητής, **HIGH** ή **LOW**)

Πχ **digitalWrite** (led, **HIGH**) ;

# Βασικές Εντολές

## ■ digitalRead ()

Διαβάζει την λογική τιμή που έχει ένα καθορισμένο digital pin

### Σύνταξη

`digitalRead`(pin number)

ή

`digitalRead`(όνομα μεταβλητής)

Πχ `digitalRead` (13) ;

# Βασικές Εντολές

## ■ delay ()

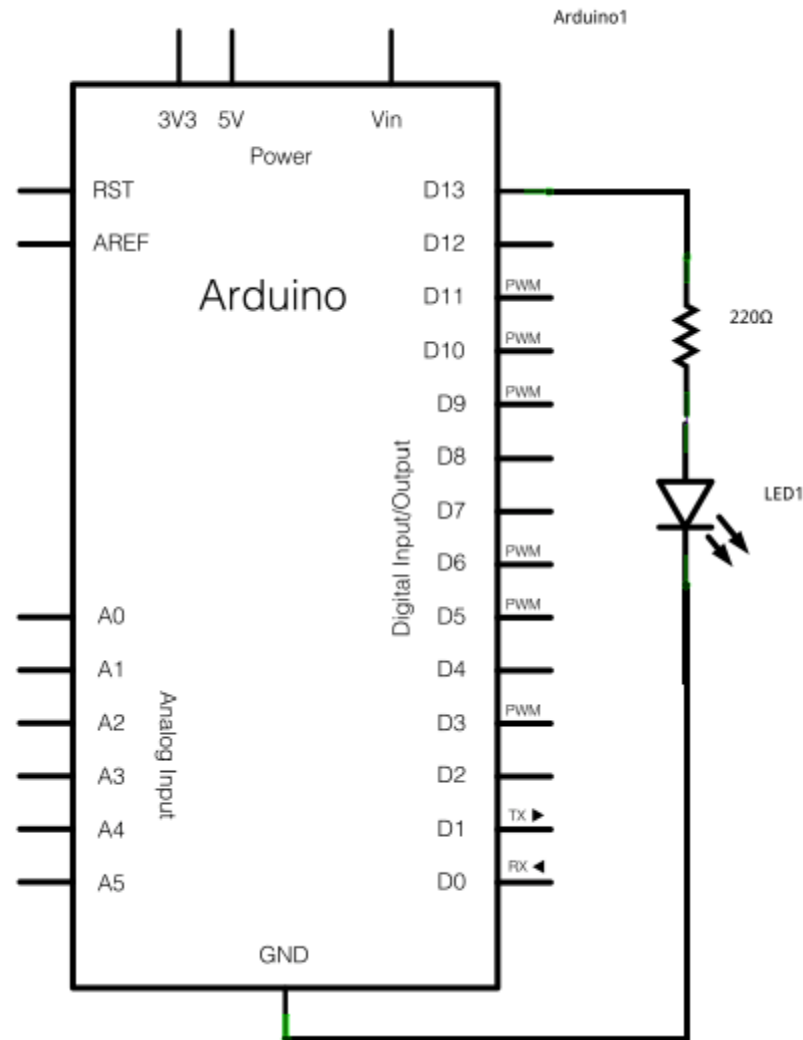
Παγώνει την εκτέλεση του προγράμματος για συγκεκριμένο χρόνο που ορίζεται σε msec.

Σύνταξη

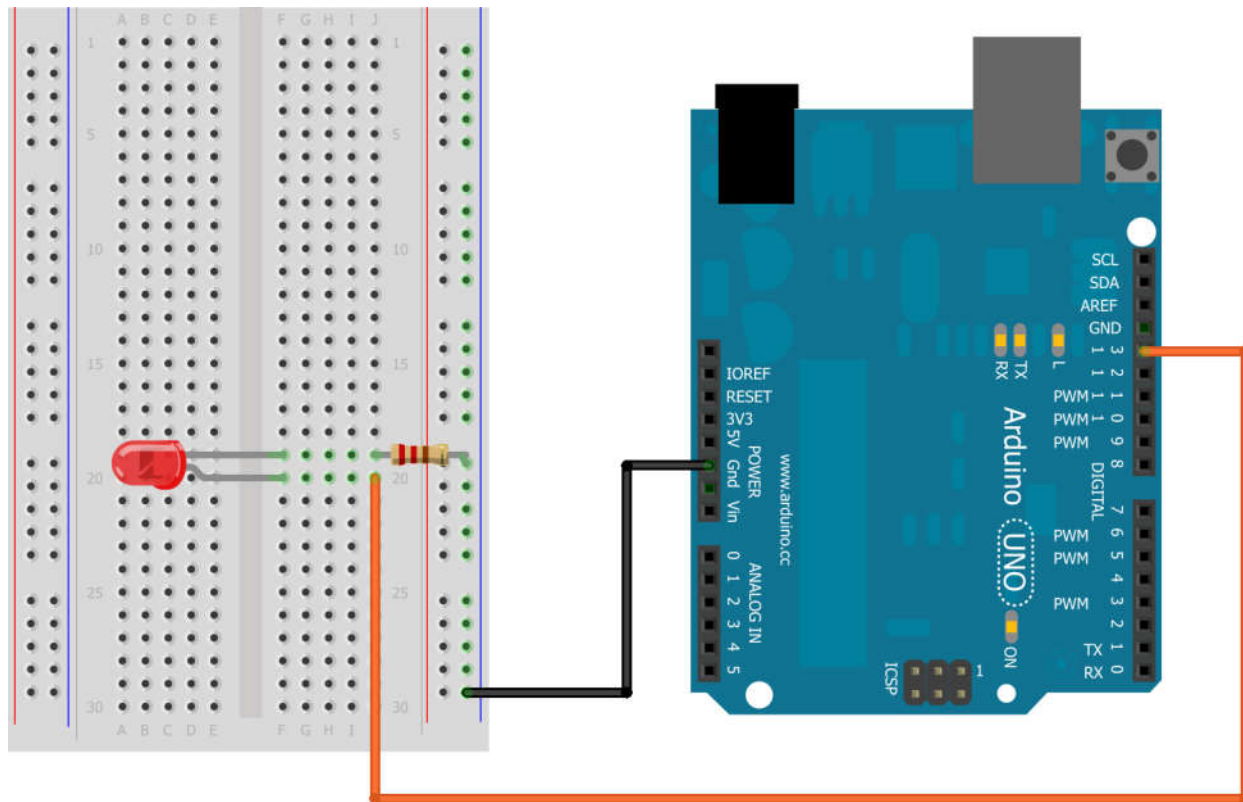
`delay`(αριθμός σε msec)

Πχ `delay` (500) ;

# Εφαρμογή 1 (Blink Led)



# Εφαρμογή 1 (Blink Led)



## ☹ Blink | Arduino 1.0.5

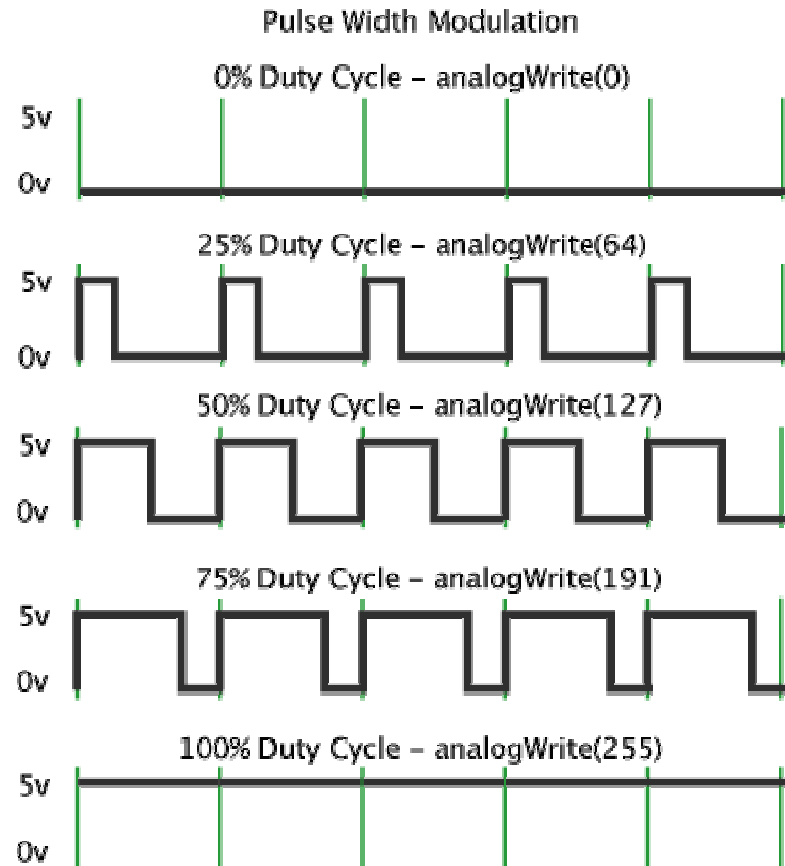
Αρχείο Επεξεργασία Σχέδιο Εργαλεία Βοήθεια



Blink §

```
/*  
  Blink  
  Turns on an LED on for one second, then off for one second, repeatedly.  
  */  
  
// Pin 13 has an LED connected on most Arduino boards.  
// give it a name:  
int led = 13;  
  
// the setup routine runs once when you press reset:  
void setup() {  
  // initialize the digital pin as an output.  
  pinMode(led, OUTPUT);  
}  
  
// the loop routine runs over and over again forever:  
void loop() {  
  digitalWrite(led, HIGH); // turn the LED on (HIGH is the voltage level)  
  delay(1000);             // wait for a second  
  digitalWrite(led, LOW);  // turn the LED off by making the voltage LOW  
  delay(1000);             // wait for a second  
}
```

# Διαμόρφωση PWM (Pulse Width Modulation)





# Εφαρμογή 2 (Dimming Effect 1 LED)

```
Ef_2_dimming_effect | Arduino 1.0.5
Αρχείο Επεξεργασία Σχέδιο Εργαλεία Βοήθεια

Ef_2_dimming_effect

//Dimming Effect me 1 led me xrisi PWM

void setup()
{
  pinMode(3,OUTPUT);
}

void loop()
{
  analogWrite(3,0);
  delay(500);
  analogWrite(3,64);
  delay(500);
  analogWrite(3,127);
  delay(500);
  analogWrite(3,191);
  delay(500);
  analogWrite(3,255);
  delay(500);
}
```

# Εφαρμογή 2 (Dimming Effect 1 LED)

```
pwm_led_2 | Arduino 1.0.5
Αρχείο Επεξεργασία Σχέδιο Εργαλεία Βοήθεια

pwm_led_2
//Dimming Effect me 1 led me xrisi PWM (me xrisi entolis for)

void setup()
{
  pinMode(3,OUTPUT);
}

void loop()
{
  for (int counter=0; counter<256; counter+=25)
  {
    analogWrite(3,counter);
    delay(500);
  }
}
```

# Εφαρμογή 2 (Dimming Effect 1 LED)

```
pwm_led_3 | Arduino 1.0.5
Αρχείο Επεξεργασία Σχέδιο Εργαλεία Βοήθεια

pwm_led_3

//Dimming Effect με 1 led με xrisi PWM (me xrisi entolis for)

void setup()
{
  pinMode(3,OUTPUT);
  Serial.begin(9600);
}

void loop()
{
  for (int counter=0; counter<256; counter+=25)
  {
    analogWrite(3,counter);
    Serial.println(counter);
    delay(500);
  }
}
```

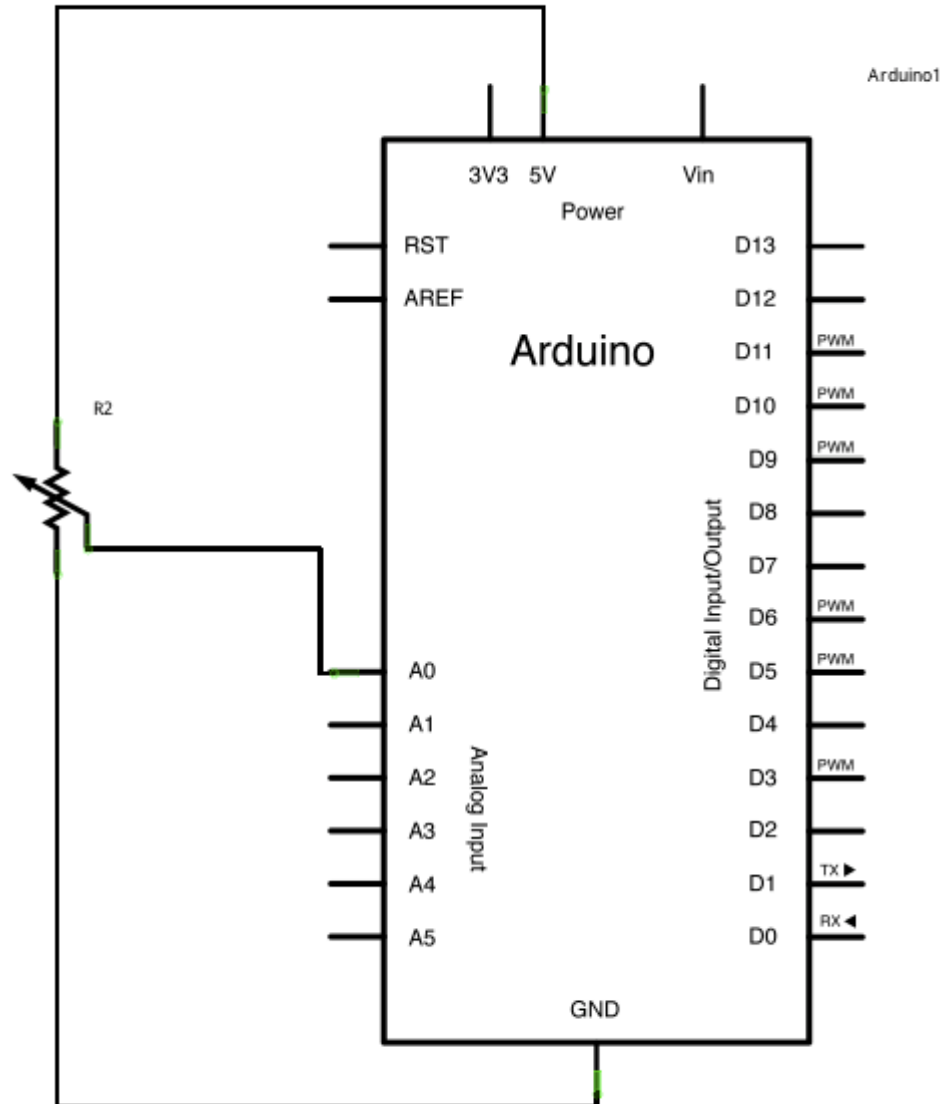
COM5

Αποστολή

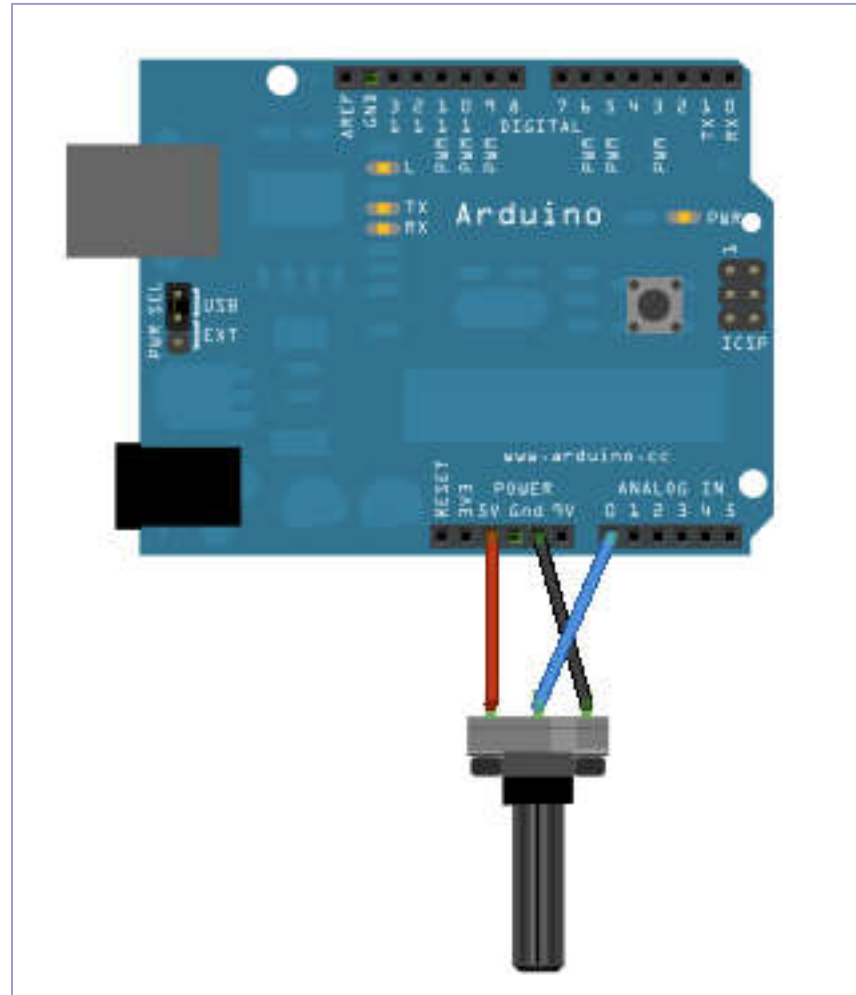
0  
25  
50  
75  
100  
125  
150  
175  
200  
225  
250  
0

Αυτόματο scroll Χωρίς τέλος γραμμής 9600 baud


# Εφαρμογή 3 (ADC 10 bit 0 έως 5V)



# Εφαρμογή 3 (ADC 10 bit 0 έως 5V)



# Εφαρμογή 3 (ADC 10 bit 0 έως 5V)



The image shows a screenshot of the Arduino IDE interface. The title bar reads "adc\_serial | Arduino 1.0.5". Below the title bar is a menu bar with options: "Αρχείο", "Επεξεργασία", "Σχέδιο", "Εργαλεία", and "Βοήθεια". Underneath the menu bar is a toolbar with icons for a checkmark, a right arrow, a document, an upload arrow, and a download arrow. The main workspace shows the filename "adc\_serial" and the following code:

```
//Analog Read kai apeikonisi sti seiriaki othoni
//0 dromeas sindeetai sto A0 tou Arduino

void setup()
{
  Serial.begin(9600);
}

void loop()
{
  int potvalue=analogRead(A0);
  Serial.println(potvalue);
  delay(500);
}
```

# Εφαρμογή 4 (Βολτόμετρο 0-5V)



```
Arduino IDE | Arduino 1.0.5
Αρχείο Επεξεργασία Σχέδιο Εργαλεία Βοήθεια

voltmeter

// Voltmetro 0-5V

void setup()
{
  Serial.begin(9600);
}

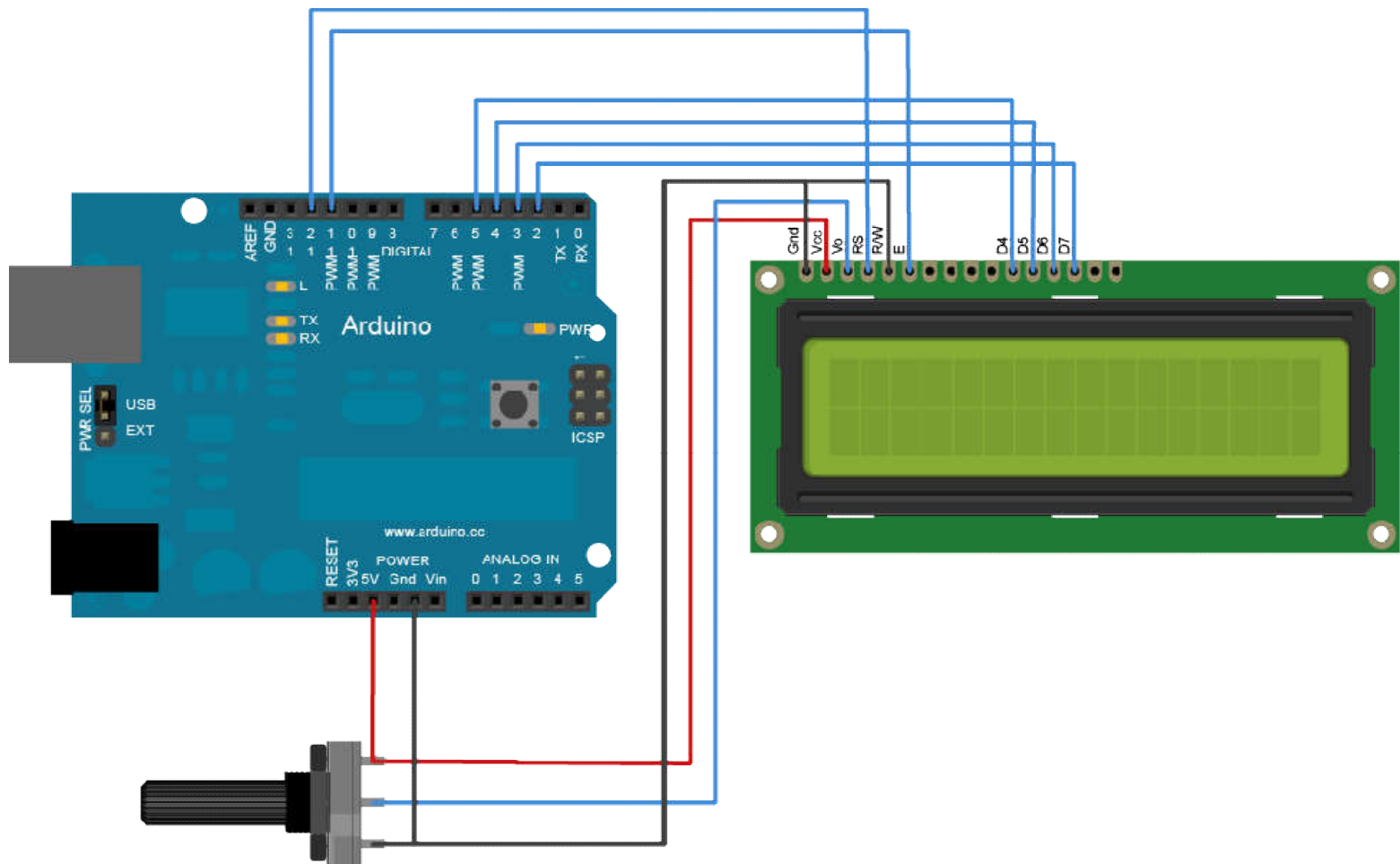
void loop()
{
  int sensorvalue=analogRead(A0);
  float Voltage=sensorvalue*(5.0/1023.0);
  Serial.print("Voltage=");
  Serial.print(Voltage);
  Serial.print("V");
  Serial.println("");
  delay(1000);
}
```

# Εφαρμογή 5 (Μήνυμα σε οθόνη LCD)

PIN NO	PIN Name	Function
1	VSS	GND
2	VDD	Power Supply
3	VEE	Contrast Adjustment Pin for Lcd to which variable resistor is connected
4	RS	Register Select pin if '0' act as Command register and '1' acts data register
5	R/W	Read/Write select '0' write and '1' means read
6	E	Enable Pin Which awake lcd
7	DB0	Dataline0
8	DB1	Dataline1
9	DB2	Dataline2
10	DB3	Dataline3
11	DB4	Dataline4
12	DB5	Dataline5
13	DB6	Dataline6
14	DB7	Dataline7
15	LED+	Power Supply to LCD's light source(LED)
16	LED-	GND to led







# Εφαρμογή 5 (Μήνυμα σε οθόνη LCD)

```
lcd_1 | Arduino 1.0.5
Αρχείο Επεξεργασία Σχέδιο Εργαλεία Βοήθεια

lcd_1
/*
 * LCD RS pin to digital pin 12
 * LCD Enable pin to digital pin 11
 * LCD D4 pin to digital pin 5
 * LCD D5 pin to digital pin 4
 * LCD D6 pin to digital pin 3
 * LCD D7 pin to digital pin 2
 * LCD R/W pin to ground
 * 10K resistor:
 * ends to +5V and ground
 * wiper to LCD V0 pin (pin 3)
 */

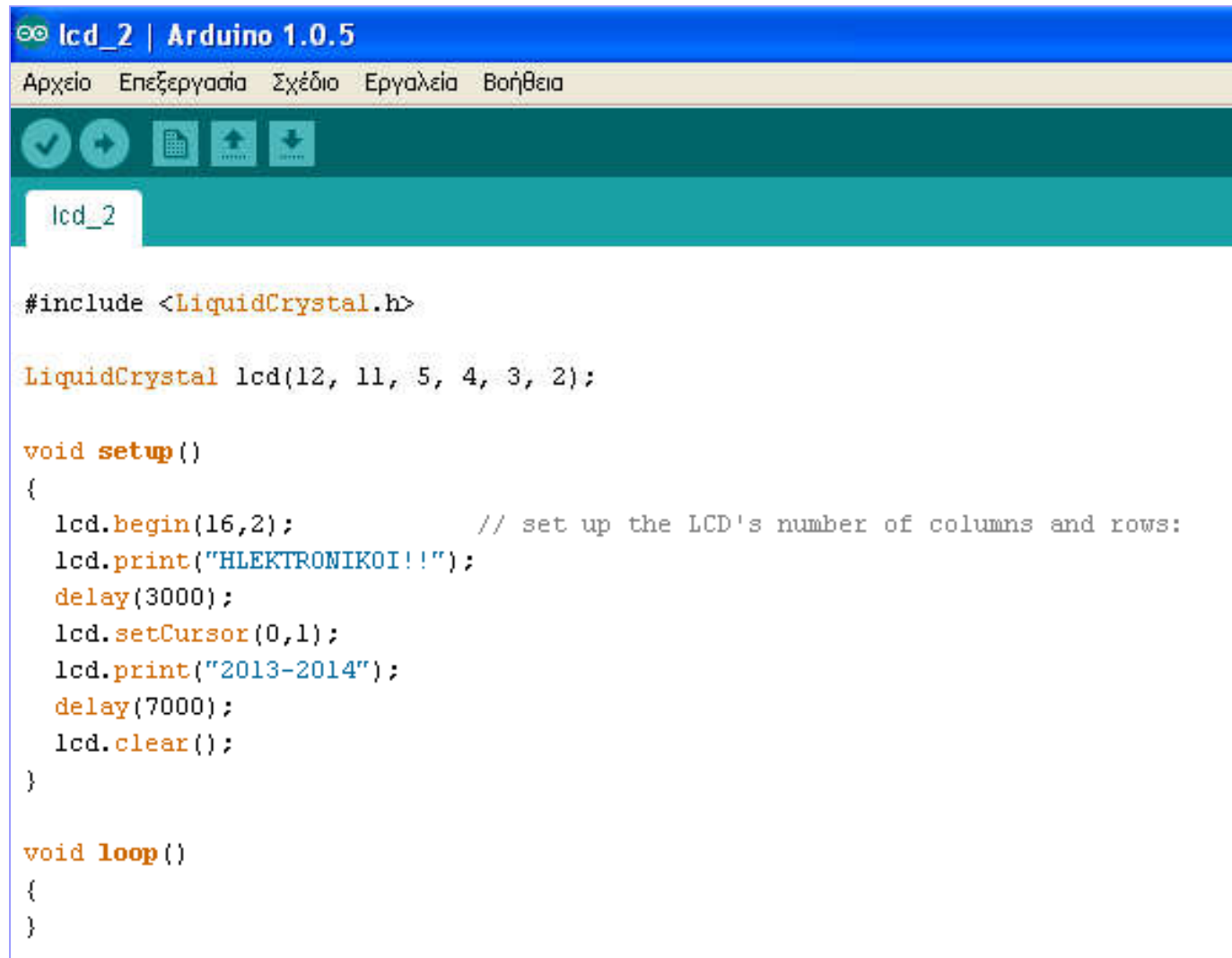
#include <LiquidCrystal.h>

LiquidCrystal lcd(12, 11, 5, 4, 3, 2);

void setup()
{
  lcd.begin(16,2);          // set up the LCD's number of columns and rows:
  lcd.print("HLEKTRONIKOI!!");
}

void loop()
{
}
```

# Εφαρμογή 5 (Μήνυμα σε οθόνη LCD)



The image shows a screenshot of the Arduino IDE interface. The title bar reads "lcd\_2 | Arduino 1.0.5". The menu bar includes "Αρχείο", "Επεξεργασία", "Σχέδιο", "Εργαλεία", and "Βοήθεια". Below the menu bar is a toolbar with icons for a checkmark, a plus sign, a document, an upload arrow, and a download arrow. The main editor area shows the following code:

```
lcd_2

#include <LiquidCrystal.h>

LiquidCrystal lcd(12, 11, 5, 4, 3, 2);

void setup()
{
  lcd.begin(16,2);           // set up the LCD's number of columns and rows:
  lcd.print("HLEKTRONIKOI!!");
  delay(3000);
  lcd.setCursor(0,1);
  lcd.print("2013-2014");
  delay(7000);
  lcd.clear();
}

void loop()
{
}
```

# Εφ.6 (Κυλιόμενο μήνυμα σε οθόνη LCD)

```
lcd_3 | Arduino 1.0.5
Αρχείο Επεξεργασία Σχέδιο Εργαλεία Βοήθεια

lcd_3
// το πρόγραμμα μετακινεί δεξιά-αριστερά ένα μήνυμα

#include <LiquidCrystal.h>

LiquidCrystal lcd(12, 11, 5, 4, 3, 2);

void setup()
{
  lcd.begin(16,2);
  lcd.print("HLEKTRONIKOI!!");
}

void loop()
{
  for (int counter=0; counter<16; counter++)
  {
    lcd.scrollDisplayRight();
    delay(400);
  }

  for (int counter=0; counter<16; counter++)
  {
    lcd.scrollDisplayLeft();
    delay(400);
  }
  delay(500);
}
```

# Εφαρμογή 7 (Θερμόμετρο με το TMP36)

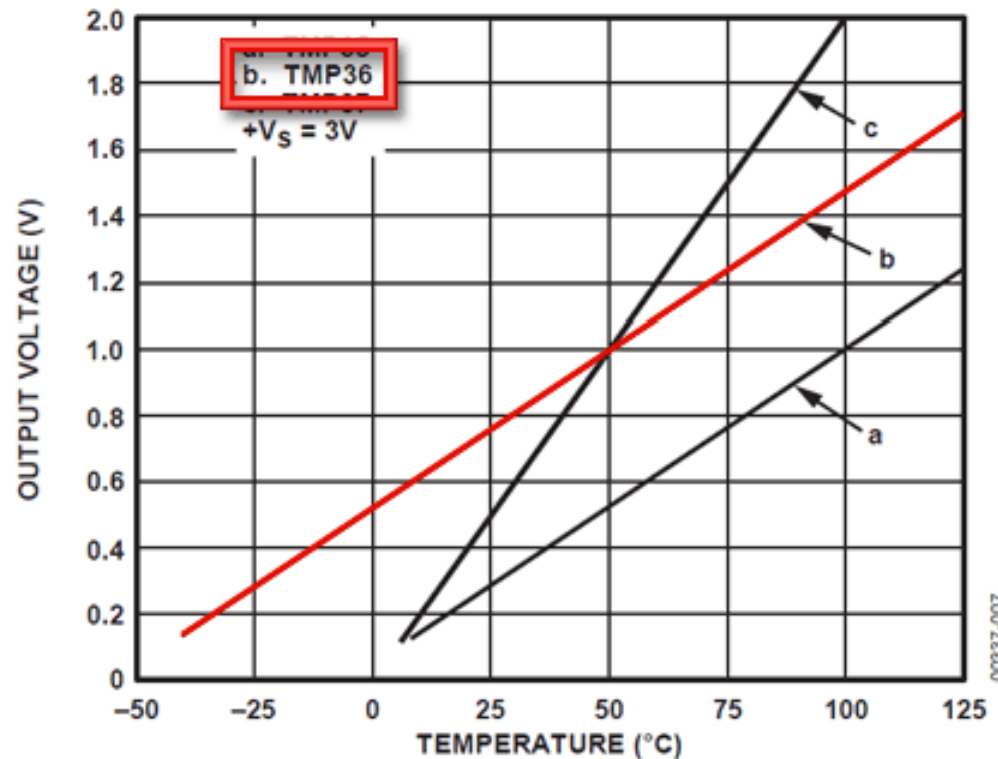
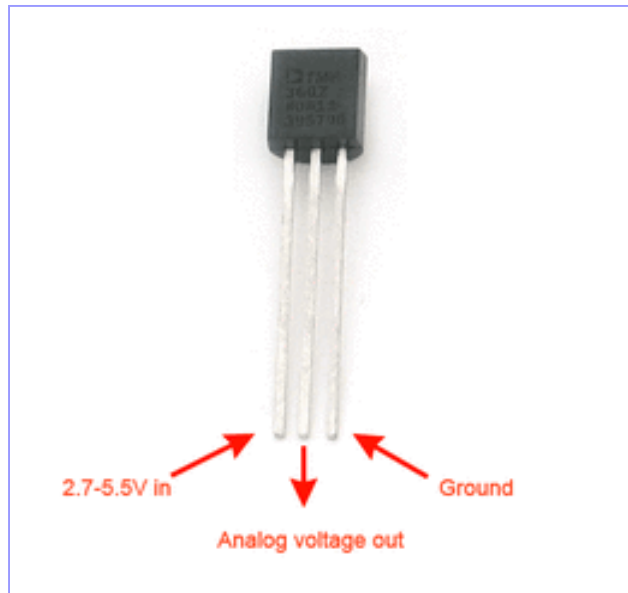
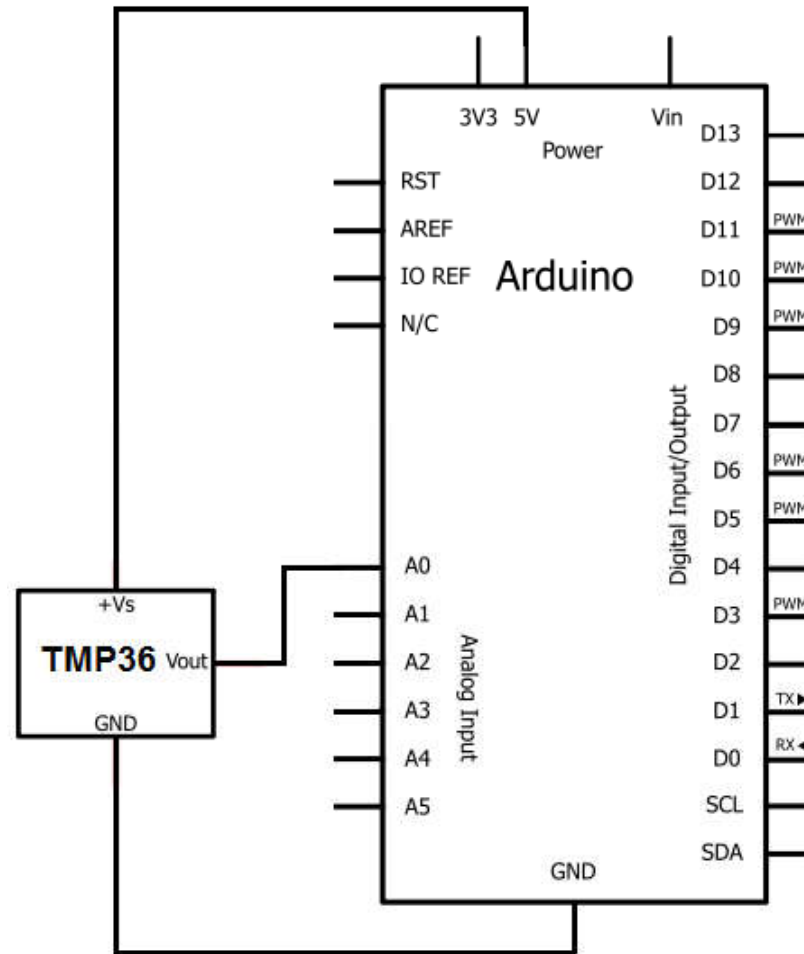
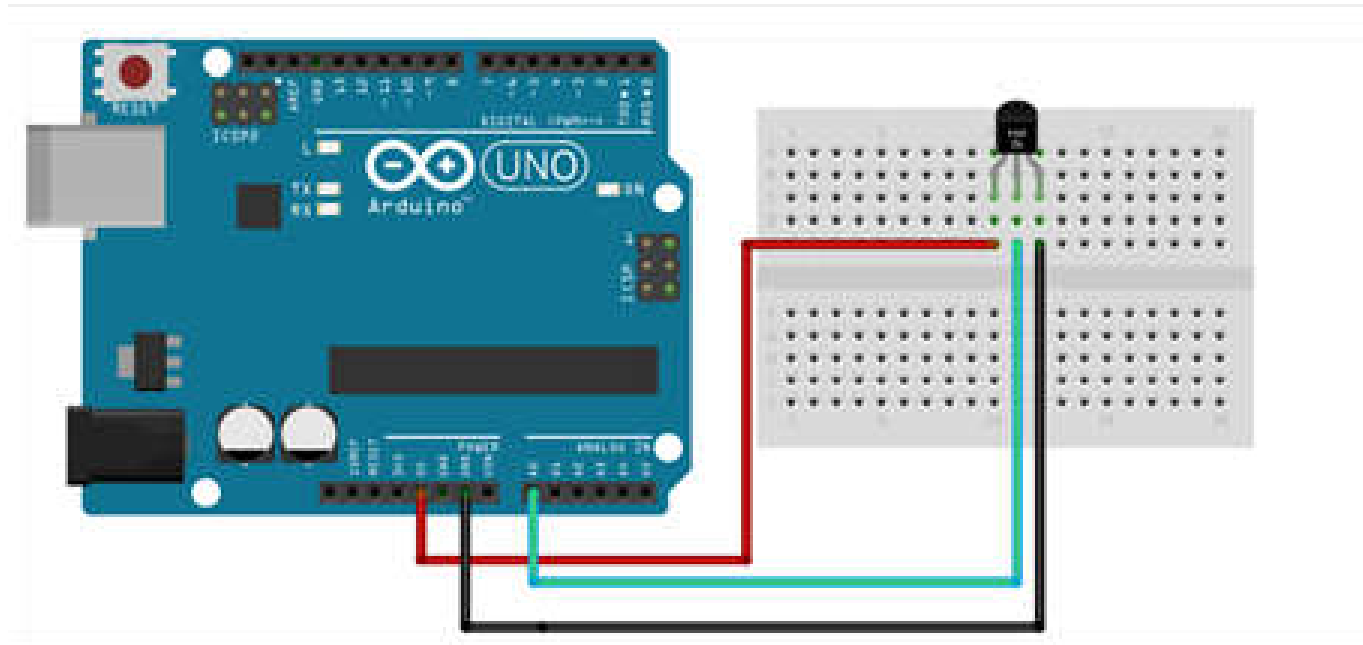


Figure 6. Output Voltage vs. Temperature

# Εφαρμογή 7 (Θερμόμετρο με το TMP36)



# Εφαρμογή 7 (Θερμόμετρο με το TMP36)



# Εφαρμογή 7 (Θερμόμετρο με το TMP36)

```
#include <LiquidCrystal.h>
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);

void setup() {
  lcd.begin(16,2);
  lcd.print("***Thermometer***");
  delay(1500);
  lcd.setCursor(3,1);
  lcd.print("with TMP36");
  delay(3000);
  lcd.clear();
}

void loop() {
  lcd.setCursor(0,0);
  int sensorvalue=analogRead(A0);
  float voltage=sensorvalue*(5000.0/1023.0);
  float tempC=(voltage-500.0)/10.0;
  lcd.print("Voltage=");
  lcd.print(voltage);
  lcd.print("mV");
  lcd.setCursor(0,1);
  lcd.print("Temp=");
  lcd.print(tempC);
  lcd.print("C");
  delay(3000);
}
```