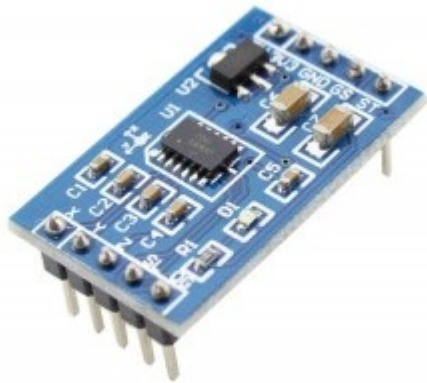


# Επιταχυνσιόμετρο-MMA7361

## Εισαγωγή

Ο αισθητήρας απαιτεί ένα πολύ μικρό ποσό ενέργειας και έχει μια είσοδο g-select που λειτουργεί το επιταχυνσιόμετρο μεταξύ  $\pm 1.5g$  and  $\pm 6g$  ( $g=9.81 \text{ m/s}^2$ ). Άλλα χαρακτηριστικά περιλαμβάνουν την κατάσταση sleep, signal conditioning, μονοπολικό φίλτρο χαμηλής συχνότητας( low pass filter), self test και 0g-detect που ανιχνεύει γραμμική ελεύθερη πτώση. Η ρύθμιση για 0-g και την ευαισθησία είναι εργοστασιακές και δεν χρειάζονται εξωτερικές συσκευές

**Model:SPS07361S**



## Λεπτομέρειες

- § Δύο επιλέξιμες περιοχές μέτρησης ( $\pm 1.5g$ ,  $\pm 6g$ ).
- § Χαμηλή κατανάλωση ρεύματος: 400  $\mu A$  Sleep mode: 3  $\mu A$ .
- § Υψηλή ευαισθησία (800 mV/g at 1.5g).
- § Επιλέξιμη ευαισθησία ( $\pm 1.5g$ ,  $\pm 6g$ ).
- § Ταχεία επαναφορά χρόνου (0.5 ms enable response time).
- § Διαστάσεις: 28 \* 17mm.

## Ορισμός Ακίδων

Pin	Pad Name	Type	Περιγραφή
<b>1</b>	5V	P	5V power
<b>2</b>	3V3	P	3V3 power
<b>3</b>	GND	GND	GND
<b>4</b>	g-select	I	Input pin to initiate Self Test
<b>5</b>	Selftest	I	Self-Test
<b>6</b>	X	A	XOUT
<b>7</b>	Y	A	YOUT
<b>8</b>	Z	A	ZOUT
<b>9</b>	Sleep	I	Sleep mode, Low active
<b>10</b>	0g-detect	O	Linear Freefall digital logic output signal

A: Analog Output (Αναλογική έξοδος)

I: Digital Input (Ψηφιακή είσοδος)

O: Digital Output (Ψηφιακή έξοδος)

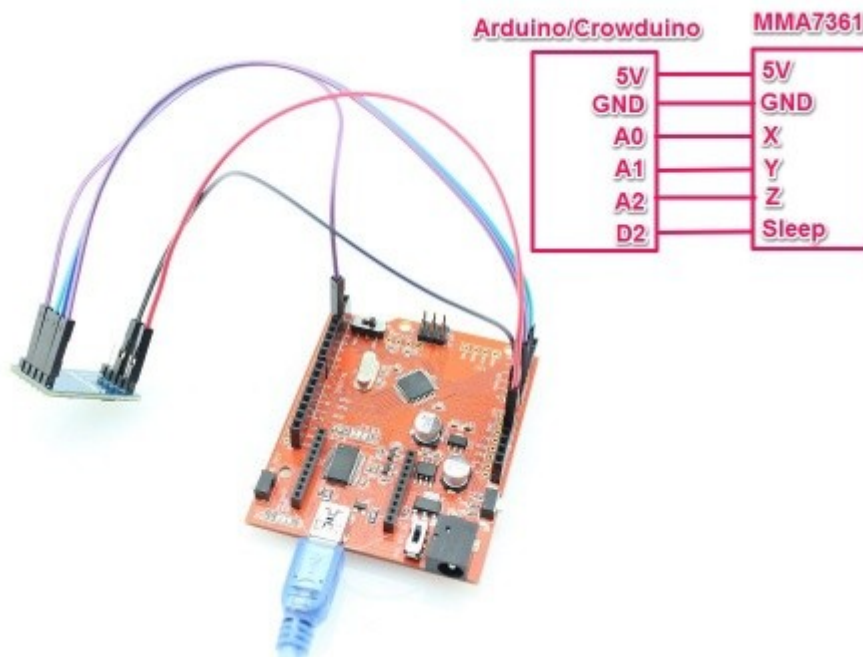
## Χρήση

Η χρήση αυτού του εξαρτήματος είναι σχετικά εύκολη. Εδώ παρουσιάζουμε δύο. Μια απλή μέθοδο και μια περιεκτική - αναλυτική.

### Χρήση με τις προεπιλεγμένες ρυθμίσεις

Συνδέστε τις X/Y/Z ακίδες (pins) του MMA7361 στις A0/A1/A2 του Arduino. και τροφοδοτήστε το με 5V . Συνδέστε το "SL" pin με το λογικό HIGH για να το ενεργοποιήσετε. you can get the test result with analogRead in Arduino IDE as below:

## 1.Σύνδεση υλικού



## 2.Αντιγράψτε το παρακάτω πρόγραμμα στο Arduino IDE και "ανεβάτε" το στο Arduino

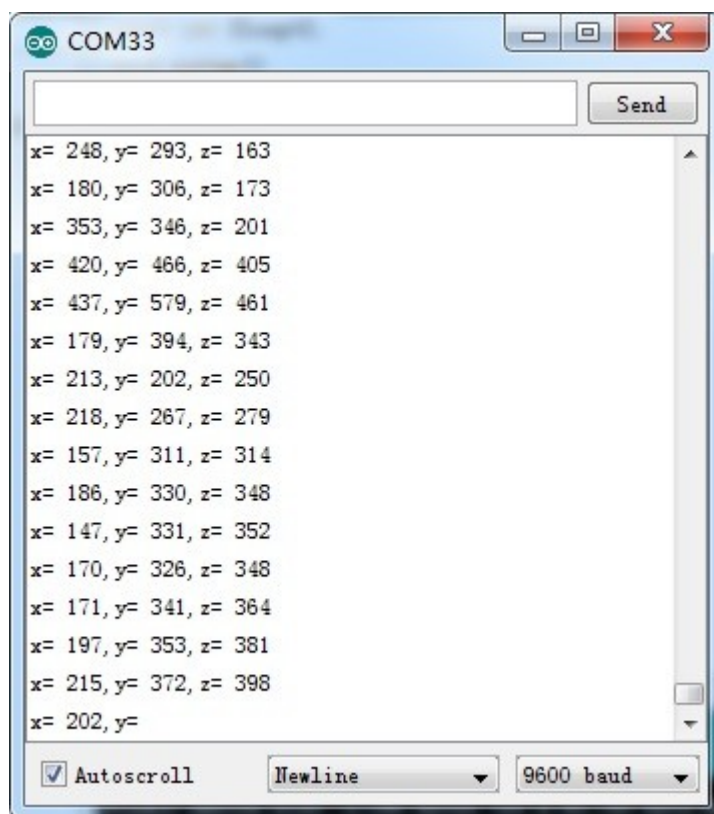
```
// # Description:
// # read the data from the accelerometer in default setting

// # Connection:
// #      x  -> Analog pin 0
// #      y  -> Analog pin 1
// #      z  -> Analog pin 2
// #

const int Sleep=2;
void setup()
{
  Serial.begin(9600); // 9600 bps
  pinMode(Sleep, OUTPUT);
  digitalWrite(Sleep, HIGH);
}
void loop()
{
  int x,y,z;
  x=analogRead(0);
  y=analogRead(1);
  z=analogRead(2);
  Serial.print("x= ");
  Serial.print(x ,DEC);
```

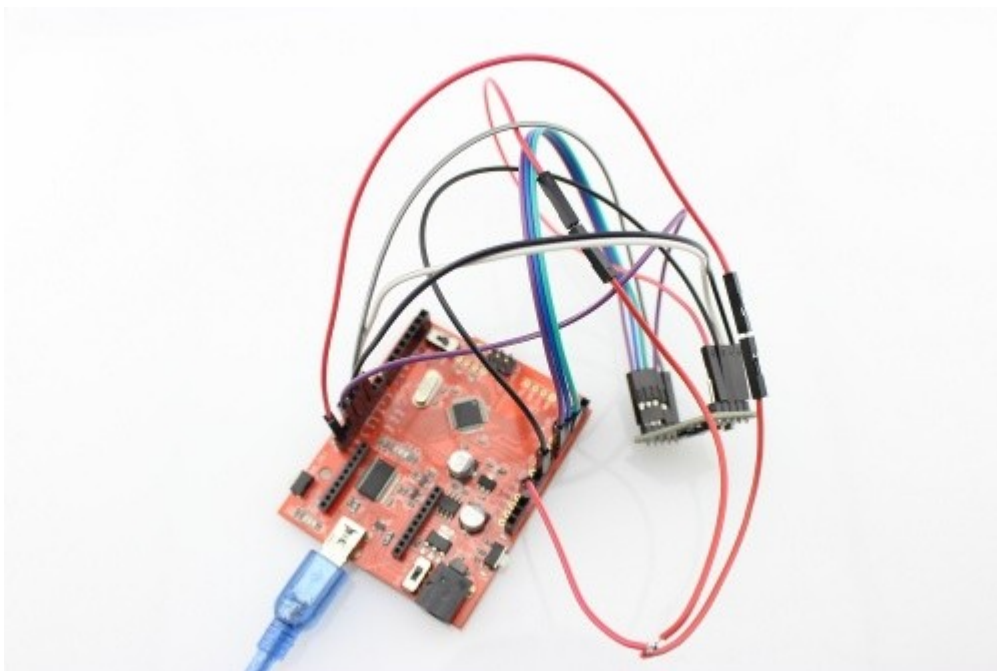
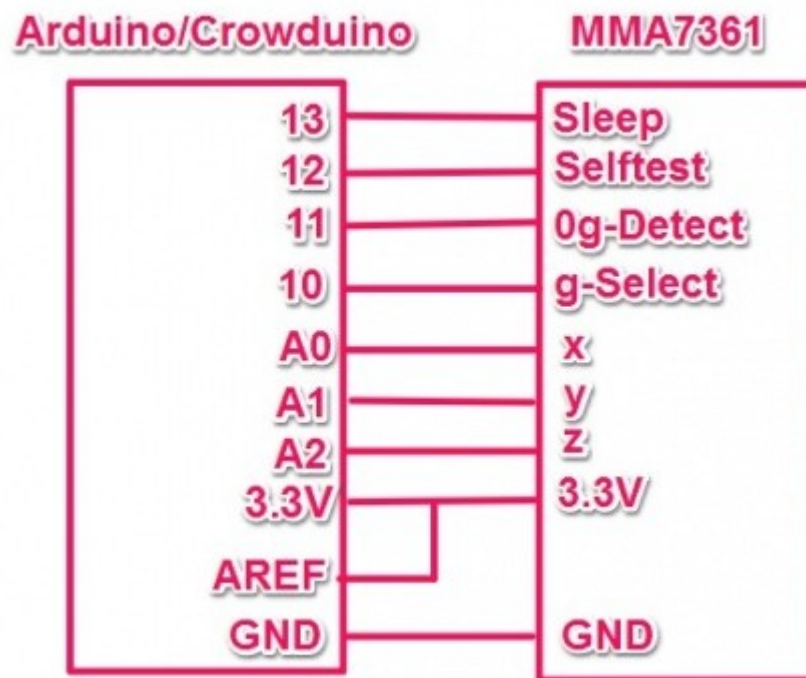
```
Serial.print(',');  
Serial.print("y= ");  
Serial.print(y ,DEC);  
Serial.print(',');  
Serial.print("z= ");  
Serial.println(z ,DEC);  
delay(100);  
}
```

3.Ανοίξτε τη σειριακή οθόνη (Serial monitor) , και θέστε το ρυθμό μετάδοσης (baudrate) σε 9600, για να δείτε κάτι όπως το παρακάτω.



## Χρήση με περιεκτικό τρόπο

### 1. Συνδέσεις υλικού



### 2. Μπορείτε να χρησιμοποιήσετε και την βιβλιοθήκη [MMA7371 Library](#) για το εξάρτημα

αυτό, αφού πρώτα συνδέσετε το εξάρτημα στο Arduino σας όπως παρακάτω :

3.Εγκαταστήστε τη βιβλιοθήκη και "ανεβάστε" το πρόγραμμα στο Arduino.

```
#include <AcceleroMMA7361.h>

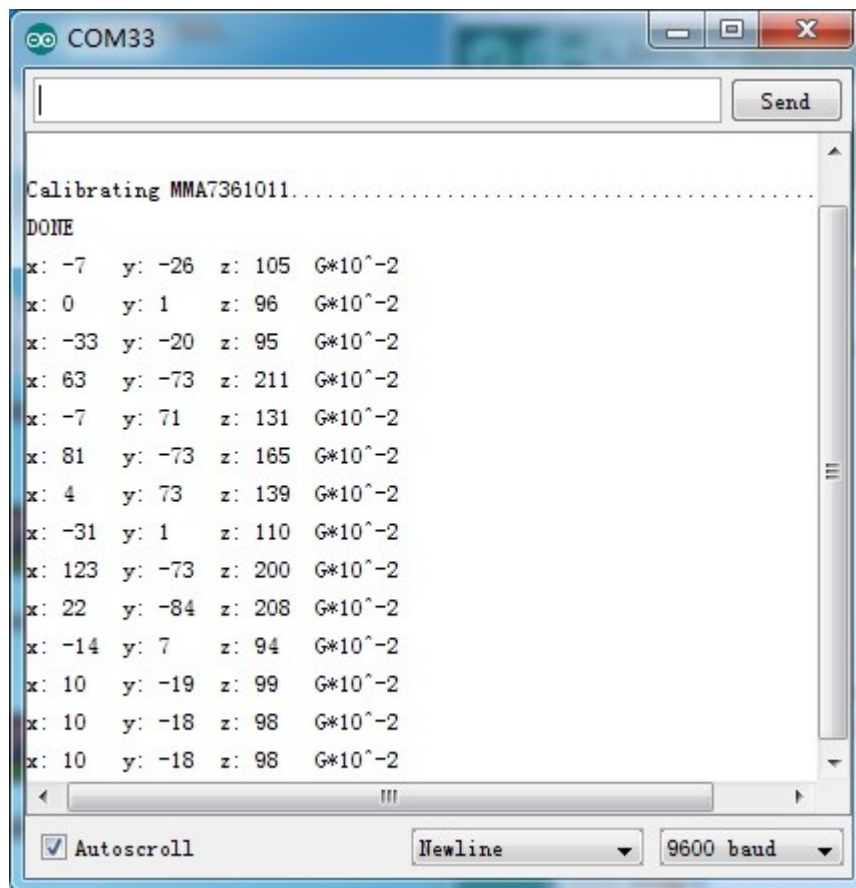
AcceleroMMA7361 accelero;
int x;
int y;
int z;

void setup()
{
  Serial.begin(9600);
  accelero.begin(13, 12, 11, 10, A0, A1, A2);
  accelero.setAREfVoltage(3.3); //sets the AREF voltage to
3.3V
  accelero.setSensitivity(LOW); //sets the sensitivity to +/-
6G
  accelero.calibrate();
}

void loop()
{
  x = accelero.getXAccel();
  y = accelero.getYAccel();
  z = accelero.getZAccel();
  Serial.print("\nx: ");
  Serial.print(x);
  Serial.print(" \ty: ");
  Serial.print(y);
  Serial.print(" \tz: ");
  Serial.print(z);
  Serial.print("\tG*10^-2");
  delay(500); //make it readable
}
```

4.Ανοίξτε τη σειριακή οθόνη, αφού βαθμονομήσετε το MMA 7361, και θα πάρετε στην έξοδο την επιτάχυνση βαρύτητας στους άξονες x, y και z. Η εικόνα που ακολουθεί δείχνει τα δεδομένα εξόδου όταν το επιταχυνσιόμετρο είναι σε επίπεδη θέση. Η επιτάχυνση στον άξονα Z είναι περίπου 100(1 G). Προσοχή κρατήστε το εξάρτημα σε επίπεδη θέση όταν

βαθμονομείται.



## Πηγές

§ MMA7361 Module DataSheet