

## Δημοσίευση Μετρήσεων στο Twitter

Για να μπορέσει να συνδεθεί ο μετεωρολογικός σταθμός στο Twitter θα πρέπει

1. Να προμηθευτούμε ένα Arduino Uno WiFi.
2. Να δημιουργήσουμε έναν λογαριασμό στο Twitter.
3. Να κατεβάσουμε και να εγκαταστήσουμε τις κατάλληλες βιβλιοθήκες από το ο <https://github.com/NeoCat/Arduno-Twitter-library>.
4. Να συνδεθούμε στο ιστοτόπο <http://arduino-tweet.appspot.com> για να λάβουμε κωδικό σύνδεσης της εφαρμογής μας στο Twitter.

### Το Arduino Uno WiFi

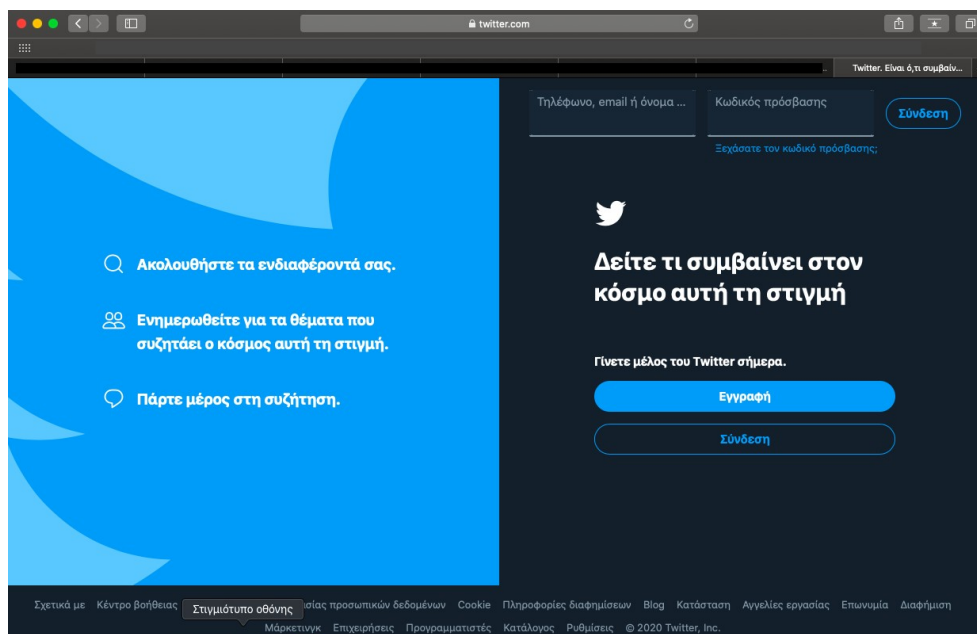
Πρόκειται για μια έκδοση του Arduino με ενσωματωμένο WiFi Module για σύνδεση σε δίκτυο WiFi.



### Δημιουργία Λογαριασμού στο Twitter

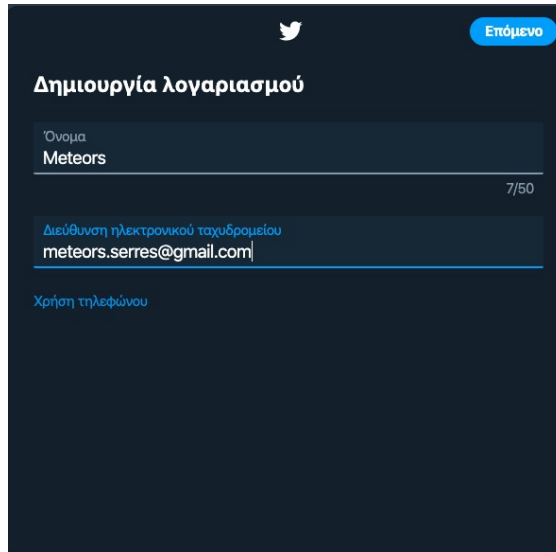
Για να δημιουργήσουμε ένα λογαριασμό στο Twitter ακολουθούμε τα παρακάτω βήματα

**ΒΗΜΑ 1ο.** Συνδεόμαστε στη σελίδα [twitter.com](https://twitter.com)



και επιλέγουμε την Εγγραφή για να δημιουργήσουμε νέο λογαριασμό.

**ΒΗΜΑ 2ο.** Το παράθυρο που εμφανίζεται είναι το ακόλουθο



Στο παράθυρο αυτό δίνουμε

- α.** το όνομα το οποίο θα εμφανίζεται στο λογαριασμό μας (εδώ Meteors)
- β.** μια διεύθυνση ηλεκτρονικού ταχυδρομείου (εδώ `meteors.serres@gmail.com`)
- γ.** Πατάμε: Επόμενο

**ΒΗΜΑ 3ο.** Το επόμενο παράθυρο είναι το ακόλουθο



Το παράθυρο αυτό είναι διαφημιστικό, οπότε μπορούμε να προχωρήσουμε στο επόμενο.

**ΒΗΜΑ 4ο.** Στο επόμενο παράθυρο πατάμε Εγγραφή

← Βήμα 3 από 5

### Δημιουργία λογαριασμού


Meteors

meteors.serres@gmail.com

Με την εγγραφή σας, αποδέχεστε τους Όρους χρήσης, την Πολιτική προστασίας προσωπικών δεδομένων και τη Χρήση των Cookies. Συμφωνείτε επίσης ότι είστε πάνω από 15 ετών.

Εγγραφή

**ΒΗΜΑ 5ο.** Ο λογαριασμός δημιουργήθηκε!

←  Επόμενο

### Σας στείλαμε έναν κωδικό

Εισαγάγετε τον παρακάτω για επαλήθευση meteors.serres@gmail.com.

Κωδικός επαλήθευσης

Δεν λάβατε το email;

Εισάγουμε τον κωδικό επαλήθευσης και ο λογαριασμός είναι έτοιμος.

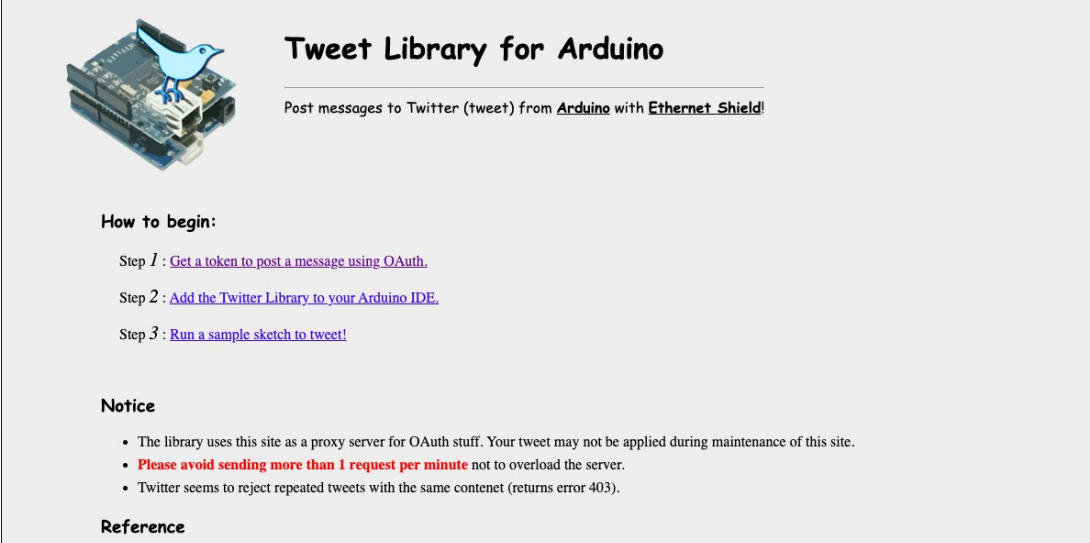
## Σύνδεση του Arduino στο Twitter

Αφού δημιουργήσετε λογαριασμό στο Twitter, το επόμενο βήμα είναι συνδέσετε το Arduino με το λογαριασμό σας.

**ΒΗΜΑ 1ο.** Κατεβάστε τις κατάλληλες βιβλιοθήκες από το <https://github.com/NeoCat/Arduino-Twitter-library>, και εγκαταστήστε τες στο IDE.

**ΒΗΜΑ 2ο.** Μεταβείτε στον ιστοτόπο <http://arduino-tweet.appspot.com>.

Θα εμφανιστεί η επόμενη Σελίδα.



**Tweet Library for Arduino**

Post messages to Twitter (tweet) from **Arduino** with **Ethernet Shield!**

**How to begin:**

Step 1 : [Get a token to post a message using OAuth.](#)

Step 2 : [Add the Twitter Library to your Arduino IDE.](#)

Step 3 : [Run a sample sketch to tweet!](#)

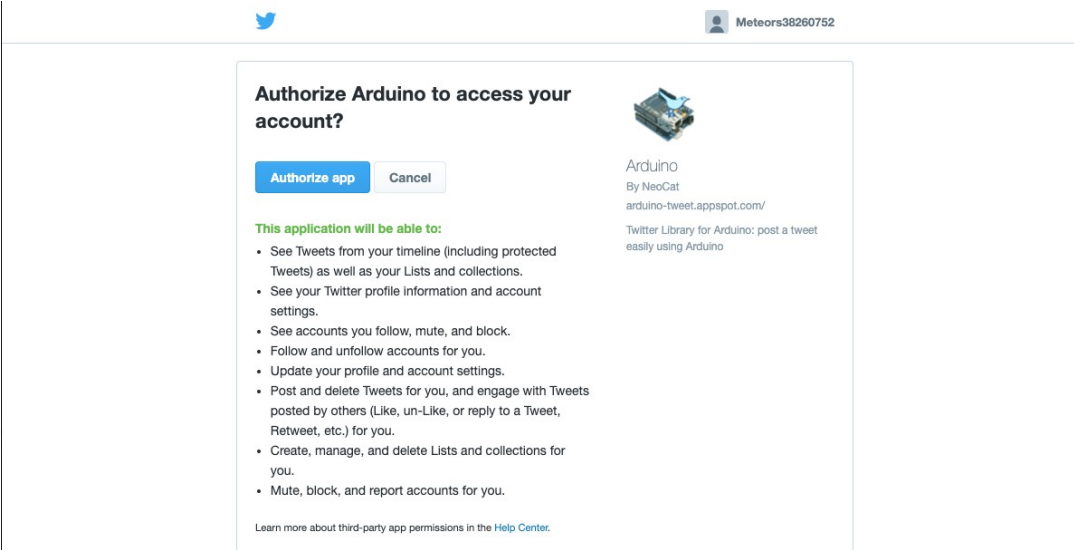
**Notice**

- The library uses this site as a proxy server for OAuth stuff. Your tweet may not be applied during maintenance of this site.
- **Please avoid sending more than 1 request per minute** not to overload the server.
- Twitter seems to reject repeated tweets with the same content (returns error 403).

**Reference**

Επιλέξτε το σύνδεσμο στο Step 1: Get a token to post a message using OAuth (κατάλληλο πρωτόκολλο για την ασφαλή επικοινωνία του Arduino με το Twitter).

**ΒΗΜΑ 3ο.** Θα εμφανιστεί το ακόλουθο παράθυρο



Authorize Arduino to access your account?

[Authorize app](#) [Cancel](#)

**This application will be able to:**

- See Tweets from your timeline (including protected Tweets) as well as your Lists and collections.
- See your Twitter profile information and account settings.
- See accounts you follow, mute, and block.
- Follow and unfollow accounts for you.
- Update your profile and account settings.
- Post and delete Tweets for you, and engage with Tweets posted by others (Like, un-Like, or reply to a Tweet, Retweet, etc.) for you.
- Create, manage, and delete Lists and collections for you.
- Mute, block, and report accounts for you.

Learn more about third-party app permissions in the [Help Center](#).

Arduino  
By NeoCat  
[arduino-tweet.appspot.com/](http://arduino-tweet.appspot.com/)  
Twitter Library for Arduino: post a tweet easily using Arduino

Επιλέξτε Authorize app και θα δημιουργηθεί ένας κατάλληλος κωδικός για ασφαλή επικοινωνία με την εφαρμογή. (Θα αναλύσουμε παρακάτω πως θα χρησιμοποιηθεί ο κωδικός αυτός).

## Κώδικας Ελέγχου Σύνδεσης στο Twitter

Ο έλεγχος της σωστής λειτουργίας της σύνδεσης στο Twitter με τον ακόλουθο κώδικα.

```
#include <WiFiNINA.h> // Βιβλιοθήκη WiFi
#include "Twitter.h" // Βιβλιοθήκη επικοινωνίας με το Twitter

char ssid[] = "ΟΝΟΜΑ_ΤΟΠΙΚΟΥ_ΔΙΚΤΥΟΥ";
char pass[] = "ΣΥΝΟΗΜΑΤΙΚΟ_ΤΟΠΙΚΟΥ_ΔΙΚΤΥΟΥ";
int keyIndex = 0; // Μεταβλητή απαραίτητη για WEP κρυπτογράφηση δικτύου
int status = WL_IDLE_STATUS; // Κατάσταση δικτύου
Twitter twitter("ΜΥΣΤΙΚΟ_ΚΛΕΙΔΙ_ΑΠΟ: arduino-tweet.appspot.com");

char msg[] = "Metors tweeting!"; // Μήνυμα που θα μεταδοθεί

void setup(){
  delay(1000); // Αναμονή για 1 sec
  Serial.begin(9600); // Εκκίνηση οθόνης υπολογιστή
  // Έλεγχος λειτουργίας του WiFi του Arduino
  if (WiFi.status() == WL_NO_MODULE) { // Αν δεν βρεθεί WiFi Module
    Serial.println("Communication with WiFi module failed!"); // Εμφάνισε μήνυμα αποτυχίας
    while (true); // Μην προχωράς στην εκτέλεση του προγράμματος
  }

  String fv = WiFi.firmwareVersion(); // Αποθήκευσε την έκδοση του firmware της διάταξης WiFi
  if (fv < WIFI_FIRMWARE_LATEST_VERSION) { // Αν δεν είναι η πιο πρόσφατη
    Serial.println("Please upgrade the firmware"); // Εμφάνισε μήνυμα ενημέρωσης/αναβάθμισης
  }

  // Προσπάθεια σύνδεσης στο τοπικό δίκτυο
  while (status != WL_CONNECTED) { // Όσο δεν είναι συνδεδεμένο
    Serial.print("Attempting to connect to SSID: "); // Εμφάνισε μήνυμα
    Serial.println(ssid); // προσπάθειας σύνδεσης
    status = WiFi.begin(ssid, pass); // Σύνδεση στο Τοπικό δίκτυο ssid, με κωδικό pass
    delay(1000); // Αναμονή για 1sec
  }
```

```
printWifiStatus(); // Εμφάνισε την κατάσταση σύνδεσης

Serial.println("connecting to Twitter..."); // Μήνυμα προσπάθειας σύνδεσης στο Twitter
if (twitter.post(msg)) { // Αν το μήνυμα (msg) μεταδοθεί
  int status = twitter.wait(&Serial); // Αναμονή απάντηση από το Twitter
  if (status == 200) { // Αν η απάντηση είναι ο αριθμός 200
    Serial.println("OK."); // τότε έγινε επιτυχής αποστολή του μηνύματος
  } else { //αλλιώς
    Serial.print("failed : code "); // αποτυχία:
    Serial.println(status); // κωδικός σφάλματος
  }
} else { // αλλιώς αν το μήνυμα δεν μεταδοθεί
  Serial.println("connection failed."); // εμφάνισε το μήνυμα ότι η σύνδεση με το Twitter απέτυχε
}
delay(10000); // Αναμονή για 10 sec
}

void loop(){ // Δεν χρειάζεται να επαναληφθεί κάτι}

void printWifiStatus() { // Συνάρτηση Ελέγχου Σύνδεσης στο Τοπικό Δίκτυο
  Serial.print("SSID: ");      // Εμφάνισε
  Serial.println(WiFi.SSID()); // το όνομα του τοπικού δικτύου

  IPAddress ip = WiFi.localIP(); // Καταχώρησε τη διεύθυνση IP που απέκτησες από το Τοπικό Δίκτυο
  Serial.print("IP Address: "); // Εμφάνισε
  Serial.println(ip);           // τη διεύθυνση IP

  long rssi = WiFi.RSSI(); // Καταχώρησε την ισχύ του σήματος
  Serial.print("signal strength (RSSI):"); // Εμφάνισε
  Serial.print(rssi);          // την ισχύ του σήματος
  Serial.println(" dBm");      // σε dBm
}
```

## Κύκλωμα Ελέγχου Λειτουργίας

Χρησιμοποιούμε τα ακόλουθα υλικά

- Arduino Uno WiFi
- Αισθητήρας BMP280
- Αισθητήρας DHT11
- Καλώδια σύνδεσης

Στη διπλανή εικόνα βλέπετε το κύκλωμα.

Η συνδεσμολογία έχει ως εξής

BMP280

$V_{IN} \rightarrow 5V$

$3V_o \rightarrow$

$GND \rightarrow GND$

$SCK \rightarrow D5$

$SDO \rightarrow D4$

$SDI \rightarrow D3$

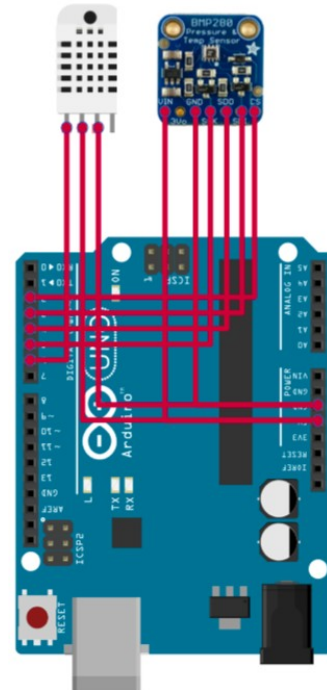
$CS \rightarrow D2$

DHT11

$VCC \rightarrow 5V$

$GND \rightarrow GND$

$DATA \rightarrow D6$



## Κώδικας Αποστολής Μετρήσεων

Ο κώδικας Arduino για την εμφάνιση των μετρήσεων είναι ο ακόλουθος.

Είναι σημειωμένα μόνο τα επιλέον στοιχεία, τα οποία και σχολιάζουμε.

```
#include <SPI.h>
#include <WiFiNINA.h> // Βιβλιοθήκη WiFi
#include "Twitter.h" // Βιβλιοθήκη επικοινωνίας με το Twitter
#include "DHT.h" // Βιβλιοθήκες χειρισμού του αισθητήρα DHT11
#include <Adafruit_Sensor.h> // Βιβλιοθήκες χειρισμού αισθητήρων της εταιρίας Adafruit
#include <Adafruit_BMP280.h> // Βιβλιοθήκες χειρισμού του αισθητήρα BMP280 της εταιρίας Adafruit

#define DHTPIN 6 // Ο DHT11 είναι συνδεδεμένος στο Pin 6 του Arduino
#define DHTTYPE DHT11 // Ο Τύπος DHT είναι ο DHT11
```

```
#define BMP_SCK 5      // Έξοδος SCK του αισθητήρα στο Pin 5 του Arduino
#define BMP_MISO 4     // Έξοδος SDO του αισθητήρα στο Pin 4 του Arduino
#define BMP_MOSI 3     // Έξοδος SDI του αισθητήρα στο Pin 3 του Arduino
#define BMP_CS 2       // Έξοδος CS του αισθητήρα στο Pin 2 του Arduino

char ssid[] = "ΟΝΟΜΑ_ΤΟΠΙΚΟΥ_ΔΙΚΤΥΟΥ";
char pass[] = "ΣΥΝΘΗΜΑΤΙΚΟ_ΤΟΠΙΚΟΥ_ΔΙΚΤΥΟΥ";
int keyIndex = 0;      // Μεταβλητή απαραίτητη για WEP κρυπτογράφηση δικτύου
int status = WL_IDLE_STATUS; // Κατάσταση δικτύου
Twitter twitter("ΜΥΣΤΙΚΟ_ΚΛΕΙΔΙ_ΑΠΟ: arduino-tweet.appspot.com");

DHT dht(DHTPIN, DHTTYPE); // Αντικείμενο Μετρήσεων του DHT11
Adafruit_BMP280 bmp(BMP_CS, BMP_MOSI, BMP_MISO, BMP_SCK); // Αντικείμενο Μετρήσεων του BMP280

char msg[140] = ""; // Πίνακας χαρακτήρων που θα φιλοξενήσει στις μετρήσεις

void setup(){
  Serial.begin(9600);
  dht.begin();      // Εκκίνηση λειτουργίας αισθητήρα DHT11
  bmp.begin();      // Εκκίνηση λειτουργίας αισθητήρα BMP280
  // Προσπάθεια σύνδεσης στο τοπικό δίκτυο
  while (status != WL_CONNECTED) { // Όσο δεν είναι συνδεδεμένο
    Serial.print("Attempting to connect to SSID: "); // Εμφάνισε μήνυμα
    Serial.println(ssid);                          // προσπάθειας σύνδεσης
    status = WiFi.begin(ssid, pass); // Σύνδεση στο Τοπικό δίκτυο ssid, με κωδικό pass
    delay(1000); // Αναμονή για 1sec
  }
  printWifiStatus(); // Εμφάνισε την κατάσταση σύνδεσης
}

void loop(){
  String myData = ""; // Καθαρισμός συμβολοσειράς που περιέχει τα δεδομένα
  myData += "Θερμοκρασία: " + String(bmp.readTemperature()) + " C\n"; // Μέτρηση Θερμοκρασίας
  myData += "Υγρασία: " + String(dht.readHumidity()) + " % \n";      // Μέτρηση Υγρασίας
```



```
myData += "Ατμοσφαιρική Πίεση: " + String(bmp.readPressure()/100) + " mbar"; // Μέτρηση Πίεσης
myData.toCharArray(msg,140); // Τελικό μήνυμα που περιέχει όλες τις μετρήσεις
if (twitter.post(msg)) {
    int status = twitter.wait();
    if (status == 200) {
        Serial.println("OK.");
    } else {
        Serial.print("failed : code ");
        Serial.println(status);
    }
} else {
    Serial.println("connection failed.");
}
delay(60000); // Αναμονή για 60sec = 1min
}
```

