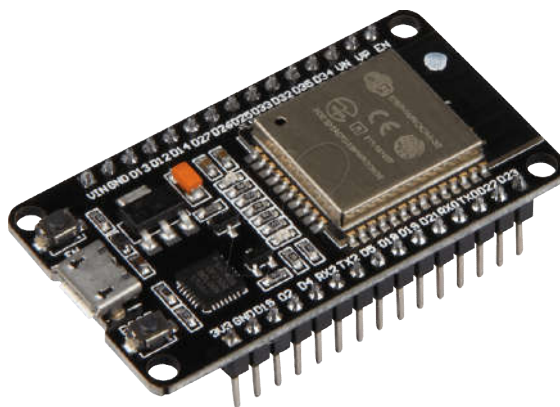




ESP32 – Τεκμηρίωση και αρχικοποίηση

Περιγραφή

Το ESP32 είναι ένας μικροελεγκτής που σχεδιάστηκε Espressif Systems ως διάδοχος του ESP8266. Είναι κατάλληλο για IoT εφαρμογές καθώς χαρακτηρίζεται από χαμηλό κόστος και χαμηλή κατανάλωση ενέργειας, ενώ υποστηρίζει ασύρματη επικοινωνία με πρωτόκολλα Wifi και Bluetooth.



Στο έργο IoT-Κήπος , χρησιμοποιήθηκαν δύο ESP32 development boards με τους παρακάτω ρόλους:

1. Σύνδεση των αισθητήρων BME280 (θερμοκρασία, υγρασία και ατμοσφαιρική πίεση), υγρασίας εδάφους και στάθμης νερού.
2. Σύνδεση των ενεργοποιητών για το φωτισμό (ταινίες led) και την ρύθμιση της θερμοκρασίας (ανεμιστήρες).

Η τεκμηρίωση και ο κώδικας για κάθε ένα από τα development boards διατίθενται στην αντίστοιχη ενότητα στο αποθετήριο του έργου «IoT-Κήπος στο ΕΚ Ευόσμου» στο github.

Τεχνικά χαρακτηριστικά

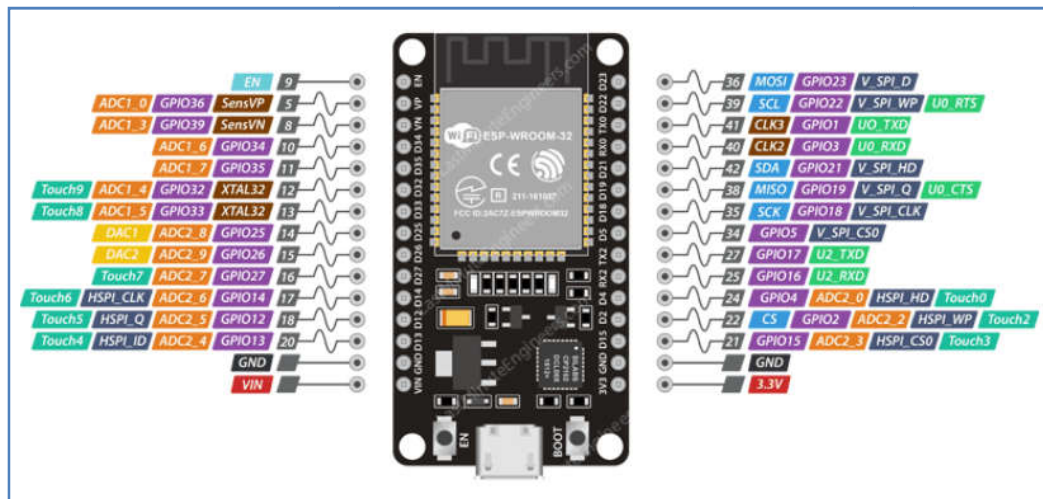
Στην κατασκευή χρησιμοποιήθηκαν development boards με την έκδοση ESP32-WROOM-32, με τα παρακάτω τεχνικά χαρακτηριστικά:

- Επεξεργαστής Dual Core 32-bit, 240MHz
- 520KB SRAM
- 4MB Flash memory



- 802.11/b/g/n Wifi
- Bluetooth 4.2/BLE

Το pinout του development board που χρησιμοποιήθηκε στην κατασκευή φαίνεται στο παρακάτω σχήμα (πηγή: <https://lastminuteengineers.com/esp32-arduino-ide-tutorial/>):



Προγραμματισμός και αρχικοποίηση

Ο προγραμματισμός του ESP32 έγινε στη γλώσσα προγραμματισμού MicroPython. Μετά την εγκατάσταση του MicroPython firmware (όπως περιγράφεται παρακάτω), το ESP32 σε κάθε εκκίνησή του εκτελεί διαδοχικά τα προγράμματα boot.py και main.py. Για τη μεταφόρτωση του κώδικα και την διαδικασία εκσφαλμάτωσής του, το ESP32 development board συνδέθηκε σε Raspberry pi.

Παρακάτω περιγράφονται αναλυτικά τα βήματα για την εγκατάσταση του MicroPython firmware και τη μεταφόρτωση κώδικα σε αυτό.

1. Εγκαταστήστε το esptool και το rshell στο Raspberry pi: Ανοίξτε ένα παράθυρο τερματικού και πληκτρολογήστε:

```
sudo pip3 install esptool
```

```
sudo pip3 install rshell
```

2. Συνδέστε το ESP32 development board στο Raspberry pi μέσω της USB θύρας (απαιτείται καλώδιο USB-to-microUSB)
3. Μπορείτε να δείτε τον αριθμό της συσκευής στην οποία έχει συνδεθεί το ESP32 πληκτρολογώντας:

```
dmesg | grep ttyUSB
```

Παρακάτω θα θεωρήσουμε ότι το EPS32 έχει συνδεθεί ως `tttyUSB0`



4. Εγκαταστήστε το MicroPython firmware σύμφωνα με τις οδηγίες στη σελίδα <https://micropython.org/download/esp32/>
5. Κατεβάστε τα προγράμματα από το αποθετήριο github του έργου IoT Κήπος σε φάκελο στο Raspberry Pi. Μπορείτε να τροποποιήσετε τον κώδικα με οποιονδήποτε κειμενογράφο.
6. Για την πρόσβαση στο σύστημα αρχείων του ESP32, τη μεταφόρτωση και την εκτέλεση του κώδικα MicroPython, πληκτρολογήστε τις παρακάτω εντολές σε ένα παράθυρο τερματικού στο Raspberry pi:

```
rshell --buffer-size=30 -p /dev/ttyUSB0
```

```
cp <filename>.py /pyboard // αντικαταστήστε το <filename> με  
το όνομα του αρχείου python που θέλετε να μεταφορτώσετε στο ESP32
```

```
repl
```

```
import <filename> // αντικαταστήστε το filename με το όνομα του  
αρχείου python που θέλετε να εκτελεστεί στο ESP32
```

Σημείωση: Εναλλακτικά η μεταφόρτωση και εκτέλεση κώδικα MicroPython στο ESP32 μπορεί να γίνει με uPyCraft IDE ή μέσω web interface με το WebRepl.

Απαραίτητες βιβλιοθήκες micropython

Το ESP32 διαθέτει δυνατότητα σύνδεσης σε δίκτυο wifi. Μέσω αυτής της σύνδεσης αποκτά πρόσβαση στον Node-Red server και τον Mosquitto MQTT broker ώστε να μπορεί να στέλνει (publish) και να λαμβάνει(subscribe) δεδομένα από άλλες συνδεδεμένες συσκευές.

- Για τη σύνδεση στο wifi δίκτυο χρησιμοποιήθηκε η ενσωματωμένη βιβλιοθήκη network
- Για την υλοποίηση του πρωτοκόλλου MQTT χρησιμοποιήθηκε η βιβλιοθήκη mqtt (<https://github.com/pycom/pycom-libraries/blob/master/examples/mqtt/mqtt.py>)

Χρήσιμοι Σύνδεσμοι

- <https://www.espressif.com/en>
- <https://www.espressif.com/en/products/hardware/esp32/overview>
- <https://lastminuteengineers.com/esp32-arduino-ide-tutorial/>
- <http://docs.micropython.org/en/latest/esp32/quickref.html>
- <https://micropython.org/download/esp32/>