

Αρχιτεκτονική υπολογιστών και ενσωματωμένα συστήματα

Εργασία εξαμήνου

Βασίλειος Ναστος
Παναγιώτης Κορομηλιάς

Msc Τμήμα Πληροφορικής και Τηλεπικοινωνιών
Πανεπιστήμιο Ιωαννίνων
Άρτα 47100, Ελλάδα

Οι μικροελεγκτές

- είναι μικρά τσιπ υπολογιστών που έχουν σχεδιαστεί για να εκτελούν συγκεκριμένες εργασίες και να ελέγχουν ηλεκτρονικές συσκευές.
- είναι συνήθως συσκευές χαμηλής κατανάλωσης ενέργειας, γεγονός που τους καθιστά κατάλληλους για εφαρμογές που τροφοδοτούνται από μπαταρίες.

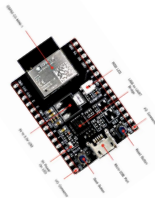
Οι Μικροεπεξεργαστές:

- είναι μικρές συσκευές που ενσωματώνουν μηχανικά και ηλεκτρικά εξαρτήματα σε μικροκλίμακα.
- Συχνά χρησιμοποιούνται σε συνδυασμό με άλλα εξαρτήματα, όπως αισθητήρες και ηλεκτρονικά, για τη δημιουργία ολοκληρωμένων συστημάτων.

Οικογένεια μικροελεγκτών ESP-32



**ESP32-S2 Wifi
Microcontroller**

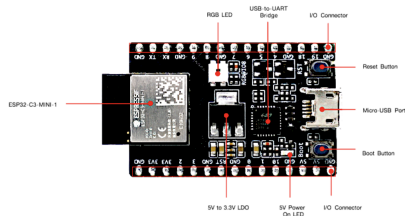


**ESP32-C3 Wifi
Microcontroller**



**ESP32-WROOM Wifi
and Bluetooth module**

- Ο ESP32-C3 είναι ένας μικροελεγκτής χαμηλού κόστους και χαμηλής κατανάλωσης ενέργειας, σχεδιασμένος ειδικά για εφαρμογές IoT, με CPU υψηλής απόδοσης και ενσωματωμένες δυνατότητες Wi-Fi και Bluetooth.
- Ο ESP32-C3 υποστηρίζει πολλαπλές διεπαφές επικοινωνίας και λειτουργίες εξοικονόμησης ενέργειας, καθιστώντας το μια εξαιρετικά ευέλικτη και ενεργειακά αποδοτική λύση για ένα ευρύ φάσμα εφαρμογών IoT, όπως οικιακός αυτοματισμός, έξυπνος φωτισμός και φορητές συσκευές.



Βασικοί ανταγωνιστές



Nordic Semiconductor chip
nRF52840



Texas Instruments chip
CC3220SF



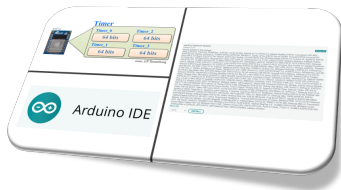
Arduino MKR

- Το Arduino IDE είναι ένα δωρεάν και ανοικτού κώδικα περιβάλλον ανάπτυξης λογισμικού, υλοποιημένο σε java, που χρησιμοποιείται για τη δημιουργία και τον προγραμματισμό κώδικα για πλακέτες μικροελεγκτών Arduino.
- Υποστηρίζει ένα μεγάλο εύρος γλωσσών όπως C, C++, Rust κ.α
- Διαθέτει library manager

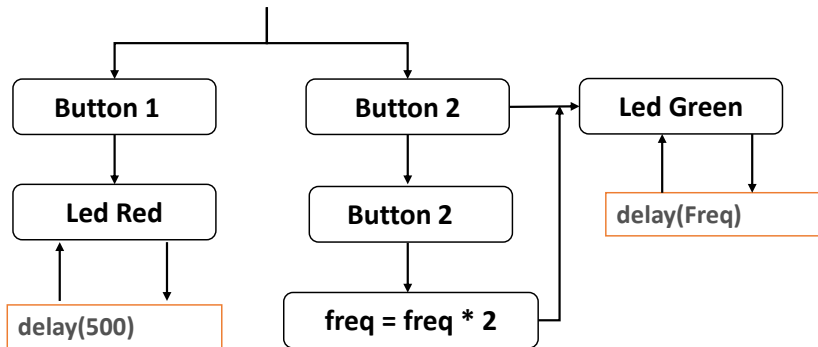


Για την υλοποίηση του πειραματικού μέρους χρησιμοποιήθηκαν:

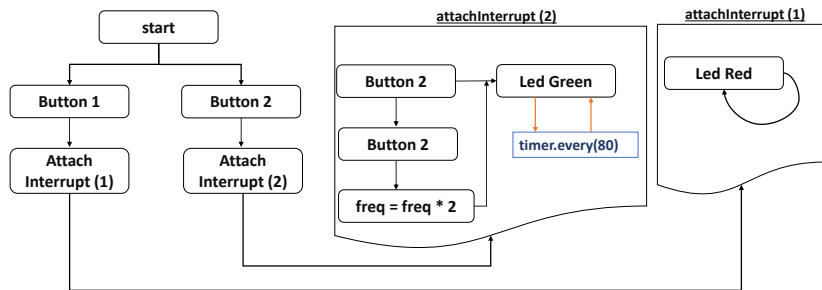
- Arduino IDE
- esp32 by Espressif Systems version 2.0.6 library
- Timers and Interrupts



Πρώτη Εφαρμογή



Δεύτερη Εφαρμογή



Interrupts/Timers (1/2)

Timers:

- Χρησιμοποιείται για τη δημιουργία περιοδικών διακοπών ή ως λειτουργία καθυστέρησης.
- Τέσσερις timers υλικού που είναι διαθέσιμοι στο ESP32, ο καθένας με πολλαπλά κανάλια.
- Μπορούν να λειτουργούν σε διαφορετικούς τρόπους λειτουργίας, όπως συνεχής ή one-shot mode.
- Μπορούν να ενεργοποιήσουν διακοπές σε ένα συγκεκριμένο χρονικό διάστημα ή όταν φτάσουν σε μια συγκεκριμένη τιμή.

Interrupts:

- Σήματα που αναστέλλουν προσωρινά την εκτέλεση του προγράμματος για να χειριστούν ένα συμβάν.
- Δύο τύποι: διακοπές υλικού και διακοπές λογισμικού.
- Διακοπές υλικού που ενεργοποιούνται από εξωτερικά γεγονότα, όπως αλλαγές στην κατάσταση ακροδεκτών ή χρονιστών.
- Διακοπές λογισμικού που ενεργοποιούνται από εντολές λογισμικού.

Interrupts/Timers (2/2)

- Διαμορφώση interrupts για τα button χρησιμοποιώντας τη συνάρτηση `attachInterrupt`, καθορίζοντας την λειτουργία ακμής ως πτωτική.
- Αρχικοποίηση του timer χρησιμοποιώντας τη συνάρτηση `timerBegin`.

```
void setup() {  
  pinMode(GREEN, OUTPUT);  
  pinMode(BLUE, OUTPUT);  
  pinMode(RED, OUTPUT);  
  pinMode(SW1, INPUT_PULLUP);  
  pinMode(SW2, INPUT_PULLUP);  
  
  // Interrupts for buttons  
  attachInterrupt(digitalPinToInterrupt(SW1), onButton1Press, FALLING);  
  attachInterrupt(digitalPinToInterrupt(SW2), onButton2Press, FALLING);  
  
  // Initialize timer  
  timer = timerBegin(0, 80, true);  
  timerAttachInterrupt(timer, &onTimer, true);  
  timerAlarmWrite(timer, green_led_delay, true);  
  timerAlarmEnable(timer);  
}
```

- Ο κώδικας είναι γραμμένος σε C χρησιμοποιώντας το Arduino framework, καθιστώντας εύκολη την τροποποίηση και την αξιοποίηση της ενσωματωμένης υποστήριξης του ESP32 για διακοπές και χρονομετρητές υλικού, η οποία μειώνει την κατανάλωση ενέργειας και βελτιώνει τις επιδόσεις.
- Η υλοποίηση της εφαρμογής αναδεικνύει τις δυνατότητες του ESP32 για την ανάπτυξη προηγμένων εφαρμογών και υπογραμμίζει τη σημασία του αποτελεσματικού προγραμματισμού και των τεχνικών βελτιστοποίησης για την αύξηση της απόδοσής τους. Επιπλέον, η χρήση της γλώσσας προγραμματισμού C παρέχει την ευελιξία και τον έλεγχο που απαιτούνται για την υλοποίηση σύνθετων εργασιών στο ESP32.

Σας ευχαριστούμε