



## Εργαστήριο Μαθήματος: “Μικροεπεξεργαστές & Περιφερειακά”

### 3η Εργασία

30/5/2024

#### **Ομάδα 18:**

**Μαμουγιώργη Μαρία 10533**

**Ξακουστού Αιμιλία 10324**

#### **ΕΙΣΑΓΩΓΗ**

Στα πλαίσια της εργασίας υλοποιήθηκε ένα πρόγραμμα C, στο οποίο αρχικά μέσω σειριακής διεπαφής UART ο χρήστης δίνει έναν αριθμό (ΑΕΜ) και στη συνέχεια λαμβάνοντας υπόψη και την είσοδο του χρήστη πραγματοποιούνται κάποιες διεργασίες που αναλύονται παρακάτω. Σημαντικό ρόλο βέβαια στην εκτέλεση των διεργασιών διαδραματίζουν και οι αισθητήρες αφής και θερμοκρασίας/υγρασίας DHT11. Οι διεργασίες, εκ των οποίων οι περισσότερες προσδιορίζονται ως ISR (Interrupt Service Routine), τροποποιούν την περίοδο, την κατάσταση του led και συμβάλλουν στην εκτύπωση των ζητούμενων.

#### **ΟΡΙΣΜΟΙ ΜΕΤΑΒΛΗΤΩΝ**

Έχοντας πραγματοποιήσει τη συνδεσμολογία στο board, γίνεται αρχικά η αντιστοίχιση των εξωτερικών περιφερειακών στην πλακέτα. Έπειτα ακολουθούν κάποιες global μεταβλητές οι οποίες και αρχικοποιούνται. Μερικές από αυτές είναι η period, ώστε να μπορεί εύκολα να τροποποιείται η περίοδος δειγματοληψίας του αισθητήρα, η counter, που εξασφαλίζει τον έλεγχο και την αποτελεσματικότερη εναλλαγή της περιόδου στη συνάρτηση timer\_isr. Ακόμη, για το touch\_isr του αισθητήρα αφής δηλώνεται η touch\_counter, η οποία ενημερώνεται κάθε φορά που “πατιέται” ο αισθητήρας. Οι μεταβλητές temperature, humidity είναι για τη θερμοκρασία και την υγρασία αντίστοιχα και ορίζονται ως global για να μπορούν να αξιοποιηθούν στις διάφορες συναρτήσεις.

#### **ΣΥΝΑΡΤΗΣΗ TIMER\_ISR**

Η συνάρτηση timer\_isr χειρίζεται τις περιοδικές μετρήσεις του αισθητήρα DHT11 και τον έλεγχο του εξωτερικού LED, το οποίο ρυθμίζεται ανάλογα με το εύρος της θερμοκρασίας. Αρχικά, κάθε φορά που καλείται η εν λόγω ISR αυξάνεται κατά ένα ο counter, μέχρι να φτάσει στο εκάστοτε period, που θα μηδενίζεται. Η υλοποίηση της περιόδου πραγματοποιείται με τον συγκεκριμένο τρόπο ώστε να επιτευχθεί η ελάχιστη καθυστέρηση στην περίοδο. Τότε πραγματοποιεί και την δειγματοληψία από τον DHT11 sensor. Αρχικά, προετοιμάζεται για ανάγνωση των δεδομένων από τον αισθητήρα, θέτοντάς τον σε PullDown για 18ms και σε PullUp για 30μs. Ακολούθως, αλλάζει σε είσοδο και περιμένει για την απόκριση του αισθητήρα. Διαβάζει 40 bits δεδομένων (από τα MSB στα LSB), τα οποία χωρίζει σε ακέραιο και δεκαδικό μέρος της υγρασίας και της θερμοκρασίας. Η υγρασία αντιστοιχεί στα πρώτα 16 bit της ένδειξης του αισθητήρα για αυτό και γίνεται ο διαχωρισμός με αυτόν τον τρόπο. Η θέση του bit μέσα σε κάθε byte καθορίζεται από τον υπολογισμό  $7 - i \% 8$ , διασφαλίζοντας ότι διαβάζεται πρώτα το MSB. Αυτή η μέθοδος εξασφαλίζει ότι τα δεδομένα του αισθητήρα μετατρέπονται με ακρίβεια σε αναγνώσιμες τιμές υγρασίας και θερμοκρασίας. Αν η θερμοκρασία είναι αρνητική, την μετατρέπει αναλόγως. Τέλος, υπολογίζει τις τιμές της θερμοκρασίας και της υγρασίας και εκτυπώνει μέσω UART τα απαραίτητα ζητούμενα σύμφωνα με την εκφώνηση, ανάλογα την τιμή του touch\_counter. Ακόμα ανάλογα με το εύρος των τιμών της θερμοκρασίας καλεί τα αντίστοιχα ISR που χειρίζονται το εξωτερικό led.

#### **ΣΥΝΑΡΤΗΣΗ TEMP\_GT**

Πρόκειται για τη ρουτίνα που καλείται όταν η θερμοκρασία ξεπερνά την τιμή των 25 βαθμών Κελσίου. Καλείται όταν γίνεται ο αντίστοιχος έλεγχος για τη θερμοκρασία. Τότε αν είναι μεγαλύτερη χρησιμοποιείται η συνάρτηση της gpio.h όπου αλλάζει την κατάσταση του led, θέτοντάς το στο 1 δηλαδή αναμμένο.

### ΣΥΝΑΡΤΗΣΗ TEMP\_LT

Πρόκειται για τη ρουτίνα που καλείται όταν η θερμοκρασία είναι μικρότερη από την τιμή των 20 βαθμών Κελσίου. Καλείται όταν γίνεται ο αντίστοιχος έλεγχος για τη θερμοκρασία. Τότε αν είναι μικρότερη χρησιμοποιείται η συνάρτηση της gpio.h όπου αλλάζει την κατάσταση του led , θέτοντας το στο 0 δηλαδή σβησμένο.

### ΣΥΝΑΡΤΗΣΗ TEMP\_MID

Πρόκειται για τη ρουτίνα που καλείται όταν η τιμή της θερμοκρασίας κυμαίνεται μεταξύ των 20 και 25 βαθμών Κελσίου. Καλείται όταν γίνεται ο αντίστοιχος έλεγχος για τη θερμοκρασία. Τότε αν είναι μεγαλύτερη χρησιμοποιείται η συνάρτηση της gpio.h όπου αλλάζει την κατάσταση του led , θέτοντας το στο 1 δηλαδή αναμμένο για 1 δευτερόλεπτο και σε 0,δηλαδή σβησμένο για άλλο 1. Κάνει ουσιαστικά εναλλαγή της κατάστασης του led ανά 1 δευτερόλεπτο.

### ΣΥΝΑΡΤΗΣΗ TOUCH\_ISR

Η συνάρτηση ISR του αισθητήρα αφής αυξάνει έναν μετρητή κάθε φορά που ενεργοποιείται και ανάλογα με την τιμή του touch\_counter, μέσω μιας if-else πραγματοποιεί τα απαιτούμενα ζητούμενα. Αν είναι η πρώτη φορά που ενεργοποιείται, αλλάζει την περίοδο ανάγνωσης της θερμοκρασίας με βάση το sum του τελευταίου και προτελευταίου ψηφίου που δόθηκαν από τον χρήστη. Αν το άθροισμα είναι 2, η περίοδος ορίζεται στα 4 δευτερόλεπτα, αλλιώς ορίζεται ίση με την τιμή της μεταβλητής sum. Αν ο μετρητής αφής είναι περιττός, η περίοδος ορίζεται στα 3 δευτερόλεπτα. Αν είναι άρτιος, η περίοδος παραμένει 2 δευτερόλεπτα και εκτυπώνονται οι τιμές θερμοκρασίας και υγρασίας μέσω UART.

### ΣΥΝΑΡΤΗΣΗ MAIN

Στη συγκεκριμένη συνάρτηση πραγματοποιείται η κλήση των προαναφερθέντων συναρτήσεων καθώς και η αρχικοποίηση, καθορισμός του timer,touch sensor, leds χρησιμοποιώντας τις συναρτήσεις των αντίστοιχων driver. Έχοντας ήδη καθορίσει ότι το AEM θα αποθηκεύεται ως πίνακας στο buffer και ότι το τελευταίο στοιχείο buff[buff\_index -1] ισούται με τον μηδενικό χαρακτήρα, το last\_digit και το prelast\_digit επειδή είναι σε ASCII μορφή (καθώς είναι string το AEM), ορίζονται ως το buff μείον την τιμή του πρώτου αριθμητικού ψηφίου ASCII ώστε να πάρω τον αριθμό και όχι τον χαρακτήρα ASCII του αριθμού ( last\_digit=buff[buff\_index - 2]-'0', prelast\_digit=buff[buff\_index-3]-'0'). Αυτά τα στοιχεία αξιοποιούνται στην συνέχεια μέσα στην touch\_isr, σε περίπτωση που είναι η πρώτη φορά που καλείται.Ο timer\_callback\_isr(timer\_isr) καθορίζει ότι η συνάρτηση timer\_isr θα εκτελείται κάθε φορά που ενεργοποιείται το ISR, επιτρέποντας την περιοδική εκτέλεση λειτουργιών δηλαδή την ανάγνωση του touch αισθητήρα. Από την άλλη, η gpio\_set\_callback(TOUCH\_PIN, touch\_isr) καθορίζει ότι η συνάρτηση touch\_isr θα εκτελείται κάθε φορά που ο ακροδέκτης TOUCH\_PIN ανιχνεύει ένα σήμα, δηλαδή όταν πατηθεί ο αισθητήρας, επιτρέποντας έτσι την άμεση απόκριση. Προκειμένου να παρθεί ορθά η είσοδος του AEM από τον χρήστη γίνεται αρχικά απενεργοποίηση του timer μέσω της timer\_disable και στη συνέχεια ενεργοποιείται μέσω της timer\_enable του timer.h.

### PROBLEMS

Υπήρξαν διάφορα προβλήματα, κυρίως ως προς τη λήψη των τιμών του αισθητήρα DHT11. Η διαδικασία αναγνώρισης των bits του μηνύματος που λαμβάνει ο επεξεργαστής από τον αισθητήρα υπήρξε ιδιαίτερα λεπτό ζήτημα ως προς τον χειρισμό του, καθώς η παραμικρή διαφορά είχε άμεσες επιπτώσεις στην σωστή δειγματοληψία από τον αισθητήρα. Ένα σημαντικό ζήτημα ήταν η άμεση αλλαγή της περιόδου ανάλογα με τις αλλαγές των υπόλοιπων αισθητήρων, το οποίο όμως πρακτικά έχει μια μικρή απόκλιση από τον ακριβή χρόνο λόγω των κλήσεων των isr και τον χρόνο που παίρνει το execution των εντολών. Επίσης, η λήψη δεδομένων από τον αισθητήρα DHT11 οδηγούσε σε ανεπιθύμητα αποτελέσματα όταν δεν ήταν μέσα στο timer\_isr για αυτό υλοποιήθηκε εκεί παρόλο που δεν ενδείκνυται για τον ISR. Κανονικά θέλαμε να υλοποιήσουμε την εκτύπωση των αποτελεσμάτων μέσα στην main, αλλά λόγω ορισμένων περιπλοκών, υλοποιήθηκε μέσα στην timer\_isr, όπως και ο έλεγχος των τιμών της θερμοκρασίας για τον προσδιορισμό της κατάστασης του εξωτερικού led.

### TESTING

Όσον αφορά την διαδικασία επαλήθευσης του κώδικά μας συνδέθηκαν οι αισθητήρες στο nucleo board μας και ελέγχθηκαν σε real time οι τιμές τους, μέσω του Keil και του PUTTY (αισθητήρας στη θερμοκρασία του ψυγείου ή στη θερμοκρασία της αίθουσας). Για τη συνδεσμολογία χρησιμοποιήθηκε αντίσταση 220Ω που συνδέθηκε με το led σε εξωτερικό board και αυτήν με τη σειρά της σε pin της πλακέτας. Ρυθμίστηκε το UART για σειριακή επικοινωνία με 115200 baud rate για τον έλεγχο του κώδικα μέσω PUTTY.