

Μικροεπεξεργαστές και Περιφερειακά

Αναφορά Εργαστηρίου Ομάδας 4:

Ονοματεπώνυμο: Φωτεινή Κοκκώνη AEM:10852

Ονοματεπώνυμο: Μπίνος Γεώργιος AEM:9333

Πρώτο Σκέλος: (UART initialization)

Αρχικά για την uart κάναμε define το buffer size μας στα 128 όπως και στα δωσμένα παραδείγματα. Αφού κάναμε include όλα τα libraries που χρειαζόμασταν για την υλοποίηση τότε δημιουργήσαμε μια queue για να αποθηκεύονται οι χαρακτήρες που θα κάναμε input. Δημιουργήσαμε διάφορες μεταβλητές για να μας βοηθήσουν στην υλοποίηση της uart όπως π.χ η Rx_Char. Στον κώδικα υπάρχουν σχόλια για το initialization των led και την χρήση των μεταβλητών (gpio pin sets included)

Δεύτερο Σκέλος: (Ανάλυση Ψηφίων)

Μέσα στο while μας αφού δώσουμε input στην uart αρχίζει η ανάλυση, έχουμε τον παρακάτω κώδικα να δέχεται τους αριθμούς που δώσαμε και να κανονίζει την περίπτωση του backspace καθώς επίσης κάνουμε την χρήση του __WFI() σε περίπτωση button interrupt όταν δίνουμε τους αριθμούς via uart. Αφού κάνουμε τον timer initialize με $\text{clk_freq } 84000000 \text{UL} / (84\text{MHz}) / 100 = 840,000 \text{ cycles}$ άρα $\text{time} = 840,000 / 84,000,000 = 0,01 \text{s}$ σκάει κάθε 100ms. Κάνουμε timer_enable και timer_set_callback(Timer_ISR) όπου η .Timer_ISR έχει έναν counter (Tick) που αυξάνεται κατά 1 κάθε φορά που «σκάει» ο timer. Έτσι περνάμε στο Led_blink activity όπου αναλύουμε των κάθε αριθμό στο queue, μηδενίζουμε στην αρχή το Tick ώστε να είναι 0 κάθε φορά που αναλύεται νέος αριθμός. Έπειτα με ifs εξασφαλίζουμε ότι με νέο input η ανάλυση μας θα σταματήσει στην μέση και θα ξεκινήσει η επόμενη. Με τα επόμενα ifs ξεχωρίζουμε τους odd από τους even αριθμούς ώστε να προχωρήσουν στα while loops που θα κάνουν toggle το led όπως ζητείται αντιστοίχως.

Τρίτο Σκέλος(Interrupt Button):

Κάναμε ένα button pin και ένα isLedFrozen στην αρχή του προγράμματος, μετά γράψαμε το Button_ISR όπου αλλάζει την isLedFrozen από false σε true όταν έχουμε odd αριθμό για το press το οποίο αυξάνεται κατά ένα κάθε φορά που καλείται η Button_ISR. Έτσι μέσα στην Led_Blink συνάρτηση μας έχουμε άλλο ένα if condition για να ελέγχει άμα το led είναι locked ή όχι. Σε even αριθμό του press απλά αλλάζει πάλι σε false (άρα επανέρχονται οι led ενέργειες) και βέβαια με κάθε πάτημα γίνεται κανονικά print και count στην uart.

Τέταρτο Σκέλος(New Number):

Αφού η λειτουργία του input της uart είναι στο while(1) loop όταν ολοκληρώνεται η ανάλυση μας ζητείται νέος αριθμός και όπως αναφέραμε προηγουμένως όταν βάλουμε τον νέο αριθμό με if condition άμα υπάρχει νέο input τότε σταματάει η ανάλυση μας και ξεκινάει η καινούργια

Πέμπτο Σκέλος(Special Case):

Χρησιμοποιούμε μια Boolean μεταβλητή isLoop την οποία ελέγχουμε αν είναι true ή false για ύπαρξη – στο τέλος με αποτέλεσμα να ενεργοποιεί ένα loop των ήδη δωσμένων αριθμών