

Σχολή Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών

Εθνικό Μετσόβιο Πολυτεχνείο



Εργαστήριο Μικροϋπολογιστών

Πέμπτη εργαστηριακή άσκηση

Ομάδα Δ07

Παπαλεξανδράκης Εμμανουήλ (Α.Μ.: 03114203)

Παπασκαρλάτος Αλέξανδρος (Α.Μ.: 03111097)

Ημερομηνία Υποβολής Αναφοράς: 11 Νοεμβρίου 2018

Θα παρουσιάσουμε μια σύντομη και ουσιαστική ανάλυση των προγραμμάτων που κατασκευάσαμε.

Τα προγράμματα αυτά καθ' αυτά είναι σε ξεχωριστά αρχεία.

Κάποιες λεπτομέρειες θα παραλειφθούν, αλλά εάν ο αναγνώστης ενδιαφέρεται, μπορεί να ανατρέξει στον κώδικα και στα σχόλια αυτού, όπου έχουμε μια πιο αναλυτική παρουσίαση.

Σημειώνουμε δε πως οι ρουτίνες που δίνονται στο pdf της εκφώνησης θεωρούνται γνωστές και δε θα τις αναλύσουμε περαιτέρω.

Άσκηση 5.1

Κατασκευάζουμε το πρόγραμμα συναγερμού που ζητείται.

Θα διαβάζουμε δύο hex αριθμούς από το πληκτρολόγιο και εάν πρόκειται για την αλληλουχία '07' θα ανάβουμε τα led της θύρας A για 4 sec, ενώ αν πρόκειται για οποιαδήποτε άλλη ακολουθία, θα αναβοσβήνουμε τα led της θύρας A ανά 0,25 sec για 4 sec.

Αλγοριθμικά, το πρόγραμμα σε C και σε assembly είναι όμοιο. Απλά, ο κώδικας στη C είναι αρκετά πιο σύντομος καθώς πρόκειται για μια γλώσσα υψηλότερου επιπέδου.

Καταρχάς, συμπεριλαμβάνουμε τις ρουτίνες που δίνονται για διάβασμα από το πληκτρολόγιο.

Έπειτα, έχουμε τις γνωστές κατάλληλες αρχικοποιήσεις.

Δημιουργούμε ένα ατέρμονο loop:

Στην αρχή, θέτουμε ένα καταχωρητή flag τον οποίο θα χρησιμοποιούμε ως σημαία.

Στη συνέχεια, διαβάζουμε από το πληκτρολόγιο. Περιμένουμε μέχρι να πατηθεί κάποιο κουμπί. Αν το κουμπί που πατήθηκε δεν είναι το '0', κάνουμε clear το flag.

Έπειτα, ξαναδιαβάζουμε από το πληκτρολόγιο. Περιμένουμε μέχρι να πατηθεί κάποιο κουμπί. Αν το κουμπί που πατήθηκε δεν είναι το '7', κάνουμε clear το flag.

Τέλος, ελέγχουμε το flag.

Αν το flag είναι ακόμα set, αυτό σημαίνει πως δόθηκαν διαδοχικά τα '0' και '7'. Συνεπώς, ανάβουμε τα led της A για 4 sec.

Διαφορετικά, αν το flag είναι clear, αυτό σημαίνει πως τουλάχιστον ένα ψηφίο που δόθηκε ήταν λάθος. Συνεπώς, αναβοσβήνουμε τα led της A για 4 sec.

Άσκηση 5.2

Κατασκευάζουμε το πρόγραμμα μετασχηματισμού hex σε dec που ζητείται.

Θα διαβάζουμε διψήφιο hex αριθμό από το πληκτρολόγιο και θα προβάλλουμε στην οθόνη τόσο τον hex αριθμό αυτόν καθ' αυτόν, όσο και τη dec του αναπαράσταση.

Καταρχάς, συμπεριλαμβάνουμε τις ρουτίνες που δίνονται για διάβασμα από το πληκτρολόγιο και προβολή στην οθόνη LCD.

Η ρουτίνα keypad_to_hex είναι όμοιας δομής με τη ρουτίνα keypad_to_ascii, με τη διαφορά πως η τιμή που θα επιστρέφεται θα είναι η αντίστοιχη αριθμητική τιμή και όχι η τιμή ascii.

Έπειτα, έχουμε τις γνωστές κατάλληλες αρχικοποιήσεις.

Δημιουργούμε ένα ατέρμονο loop:

Για το πρώτο hex ψηφίο, διαβάζουμε από το πληκτρολόγιο. Περιμένουμε μέχρι να πατηθεί κάποιο κουμπί. Μετατρέπουμε το δοθέν στην αριθμητική τιμή του. Αποθηκεύουμε την τιμή στα 4 MSB ψηφία ενός καταχωρητή value.

Επίσης, μετατρέπουμε τον αριθμό σε ascii αναπαράσταση και τον αποθηκεύουμε σε έναν καταχωρητή data. Προκειμένου να το μετατρέψουμε σε ascii, απλα ελέγχουμε αν ο αριθμός είναι μικρότερος του 0x0A. Εάν είναι πράγματι μικρότερος, προσθέτουμε 0x30 (= '0'), διαφορετικά προσθέτουμε 0x37 (= 'A' - 0x0A).

Όμοια για το δεύτερο hex ψηφίο, διαβάζουμε από το πληκτρολόγιο. Περιμένουμε μέχρι να πατηθεί κάποιο κουμπί. Μετατρέπουμε το δοθέν στην αριθμητική τιμή του. Αποθηκεύουμε την τιμή στα 4 LSB ψηφία του ίδιου καταχωρητή value.

Επίσης, μετατρέπουμε τον αριθμό σε ascii αναπαράσταση.

Προβάλλουμε με τη σειρά που δόθηκαν τα hex ψηφία (των οποίων έχουμε αποθηκεύσει τις ascii αναπαράστασές τους) με τις ρουτίνες οθόνης που δίνονται.

Σε αυτό το σημείο, ο καταχωρητής value περιέχει τη σωστή τιμή του αριθμού μας σε δυαδική μορφή.

Εξετάζουμε το MSB του value. Εάν είναι 1, πρόκειται για αρνητικό αριθμό οπότε γράφουμε '-' στην οθόνη. Διαφορετικά, γράφουμε '+'.

Επίσης, αν το MSB είναι 1, και άρα έχουμε αρνητικό αριθμό, συμπληρώνουμε ως προς 2.

Στη συνέχεια, προκειμένου να μετατρέψουμε τον αριθμό σε δεκαδική μορφή:

- i. Αφαιρούμε συνεχώς 100 μέχρις ότου προκύψει αρνητικός αριθμός. Ο αριθμός των εκατοντάδων αντιστοιχεί στο πλήθος των φορών που επαναλάβαμε αυτό το βήμα, παίρνοντας μη αρνητικό αποτέλεσμα. Διορθώνουμε το αρνητικό υπόλοιπο.
- ii. Από αυτό που έμεινε, αφαιρούμε συνεχώς 10 μέχρις ότου προκύψει αρνητικός αριθμός. Ο αριθμός των δεκάδων αντιστοιχεί στο πλήθος των φορών που επαναλάβαμε αυτό το βήμα, παίρνοντας μη αρνητικό αποτέλεσμα. Διορθώνουμε το αρνητικό υπόλοιπο.
- iii. Ό,τι έμεινε είναι οι μονάδες.

Τέλος, διαδοχικά μετατρέπουμε το πλήθος των εκατοντάδων, των δεκάδων και των μονάδων στην ascii αναπαράστασή τους και προβάλλουμε στην οθόνη.

Άσκηση 5.3

Κατασκευάζουμε το πρόγραμμα χρονομέτρου που ζητείται.

Καταρχάς, συμπεριλαμβάνουμε τις ρουτίνες που δίνονται για προβολή στην οθόνη LCD.

Έπειτα, έχουμε τις γνωστές κατάλληλες αρχικοποιήσεις.

Ως πρόσθετη αρχικοποίηση, γράφουμε το μήνυμα **'00 MIN: 00 SEC'** στην οθόνη.

Επίσης, ορίζουμε (και αρχικοποιούμε) 4 καταχωρητές minh, minl, sech, secl οι οποίοι θα κρατάνε την ascii τιμή των ψηφίων των λεπτών και των δευτερολέπτων ανά πάσα στιγμή.

Για να προβάλλουμε κάθε φορά στην οθόνη (πέρα από το πρώτο μήνυμα **'00 MIN:00 SEC'**) αρκεί να αλλάξουμε τον πρώτο, δεύτερο, όγδοο και ένατο χαρακτήρα (καθώς τα **' MIN:'** **SEC'** μένουν αναλλοίωτα. Λοιπόν, κάθε φορά που θα προβάλλουμε στην οθόνη θα γράφουμε τα minh, minl, sech, secl στις προαναφερθείσες θέσεις, χωρίς να πειράζουμε τα υπόλοιπα δεδομένα.

Δημιουργούμε ένα ατέρμονο loop:

Στην αρχή, ελέγχουμε εάν πατείται το πλήκτρο PA0. Ανν ναι, μηδενίζουμε όλους τους καταχωρητές ψηφίων, προβάλλουμε στην οθόνη τις νέες τιμές και έπειτα, επιστρέφουμε στην αρχή του loop.

Στη συνέχεια, ελέγχουμε εάν πατείται το πλήκτρο PA7. Ανν όχι, επιστρέφουμε στην αρχή του loop.

Έπειτα, καλούμε χρονοκαθυστέρηση του ενός sec.

Σε αυτό το σημείο καλούμε μια ρουτίνα για να αυξήσουμε τις τιμές των καταχωρητών ψηφίων ως εξής:

Αυξάνουμε το secl.

Ανν η νέα τιμή του secl είναι έγκυρη (από 0 έως 9) επιστρέφουμε στο κύριο πρόγραμμα.

Διαφορετικά (αν φτάσαμε το 10), μηδενίζουμε το secl και αυξάνουμε το sech.

Ανν η νέα τιμή του sech είναι έγκυρη (από 0 έως 5) επιστρέφουμε στο κύριο πρόγραμμα.

Διαφορετικά (αν φτάσαμε το 6), μηδενίζουμε το sech και αυξάνουμε το minl.

Ανν η νέα τιμή του minl είναι έγκυρη (από 0 έως 9) επιστρέφουμε στο κύριο πρόγραμμα.

Διαφορετικά (αν φτάσαμε το 10), μηδενίζουμε το secl και αυξάνουμε το minh.

Ανν η νέα τιμή του sech είναι έγκυρη (από 0 έως 5) επιστρέφουμε στο κύριο πρόγραμμα.

Διαφορετικά (αν φτάσαμε το 6), μηδενίζουμε το sech και επιστρέφουμε στο κύριο πρόγραμμα.

Τέλος, προβάλλουμε στην οθόνη τις νέες τιμές των καταχωρητών με τον τρόπο που αναφέρεται στην αρχή.