## Άσκηση 7Δ. Ασκήσεις πάνω στην οθόνη LCD(Με χρήση της flex\_lcd.h)

**Να γραφεί πρόγραμμα το οποίο θα μετράει δευτερόλεπτα ξεκινώντας από το 0. Στη θέση 5 της πρώτης γραμμής της οθόνης θα εμφανίζεται το μήνυμα SECONDS: και δίπλα η τιμή των δευτερολέπτων.**

**Θα γίνει χρήση της παρακάτω εντολής της flex\_lcd.h για εμφάνιση ακέραιων αριθμών στην οθόνη:**

**printf(lcd\_putc,"r=%d\n",r);** // Τύπωσε την τιμή της ακέραιης μεταβλητής r

//"r=%d\n" Είναι το format εκτύπωσης

// r= τύπωσε πρώτα r=

//%d τύπωσε υπό μορφή decimal

// \n πήγαινε στη δεύτερη γραμμή



Κύκλωμα άσκησης 7Γ (χρήση της flex\_lcd.h). Αναβοσβήσιμο με τη βοήθεια της ρουτίνας διακοπής

**Προσοχή!!! Στον φάκελο όπου βρίσκεται το πρόγραμμα σας θα πρέπει να τοποθετείστε το αρχείο flex\_lcd.h**

#include <main.h>

#include <flex\_lcd.h> // Το h αρχείο του προγράμματος οδήγησης της οθόνης θα

// πρέπει να είναι στον ίδιο φάκελο όπου θα αποθηκεύσουμε

//το πρόγραμμα μας.

//θα πρέπει να γίνει έλεγχος και πιθανόν τροποποίηση

// στις δηλώσεις #define LCD\_DB4 PIN\_B4 κλπ που

// υπάρχουν στην flex\_lcd.c. Με αυτές τις

// δηλώσεις καθορίζεται σε ποιους ακροδέκτες του

// μικροελεγκτή συνδέεται ο κάθε ακροδέκτης της οθόνης.

// Ανοίξτε το flex\_lcd.h με το Notepad και κάνετε

//έλεγχο στις δηλώσεις σύνδεσης των ακροδεκτών. Αν δεν

//συμφωνούν με τον τρόπο που συνδέσατε τον μικροελεγκτή στην

//οθόνη θα πρέπει να κάνετε τροποποιήσεις.

#byte PORTB=0xf81 // στην θέση 0xf81 είναι ο καταχωρητής δεδομένων

//της πόρτας Β

// Δήλωση μεταβλητών

int counter=20; //Μετρητής διακοπών (μετά από 20 διακοπές ανά 50 ms θα έχουν

// περάσει 20Χ50=1000 ms=1 second.

int seconds=0; // μεταβλητή της οποίας η τιμή θα εμφανίζεται στην οθόνη

// Δήλωση συναρτήσεων

void timer0\_int(void); //Οι συναρτήσεις που θα χρησιμοποιηθούν δηλώνονται πάνω από την main()

void init (void);

void main() {

init(); //κλήση της ρουτίνας αρχικοποίησης

lcd\_init(); // αρχικοποίηση της οθόνης

lcd\_putc("\f"); // καθαρισμός της οθόνης

while(TRUE) //ατέρμων βρόχος(βρόχος που εκτελείται συνεχώς χωρίς καμιά εντολή).

{ }

} // κλείνει η αγκύλη του main

// Ρουτίνα διακοπής από τον timer0.

#INT\_TIMER0 // Directive. Οδηγία προς τον Compiler. Δηλώνει ότι η επόμενη ρουτίνα

// είναι ρουτίνα

// διακοπών από τον timer0

void timer0\_int(void){

set\_timer0(56161); // αρχική τιμή του μετρητή

// για να συμβεί η επόμενη

// διακοπή σε 50ms

counter--; // ελαττώνεται ο μετρητής διακοπών

if (counter==0) {

**Να συμπληρωθούν οι απαραίτητες εντολές**

} // αγκύλη κλεισίματος του if

} //αγκύλη κλεισίματος της ρουτίνας διακοπών

// Ρουτίνα αρχικοποίησης

void init (void){

set\_tris\_b(0x00); // Καθορισμός της πόρτας Β ως εξόδου

set\_tris\_d(0x00); // Καθορισμός της πόρτας D ως εισόδου

seconds=0; // Αρχική τιμή του μετρητή δευτερολέπτων ίση με μηδέν

counter=20; // Αρχική τιμή του counter=20. Μετά από 20 διακοπές ο counter θα γίνει 0

SETUP\_TIMER\_0(T0\_INTERNAL | T0\_DIV\_64 ); //Prescaler=

set\_timer0(56161); // Αρχική τιμή του μετρητή timer0 για διακοπές

//κάθε 50 ms

enable\_interrupts(INT\_TIMER0); // Ενεργοποίηση της

//διακοπής του timer0

enable\_interrupts(GLOBAL); // Ενεργοποίηση του γενικού

// διακόπτη των διακοπών

}