# Άσκηση 1Γ. Χρήση των παράλληλων θυρών του μικροελεγκτή για ανάγνωση και εξαγωγή δεδομένων

**Να γραφεί πρόγραμμα σε γλώσσα προγραμματισμού C για τον μικροελεγκτή PIC 18F4550 που να διαβάζει από την πόρτα D έναν αριθμό, να τον διαιρεί δια 2 και το αποτέλεσμα να το στέλνει στην πόρτα Β.**



Κύκλωμα άσκησης 1Γ

Διάγραμμα ροής:

Αρχή

Η θύρα D γίνεται είσοδος

Η θύρα Β γίνεται έξοδος

Ορισμός μεταβλητής για γράμματα επιθέτου

Ορισμός μεταβλητής για γράμματα ονόματος

Ορισμός μεταβλητής για αποθήκευση αποτελέσματος πράξης

Ανάγνωση από την θύρα D

Εκτέλεση της διαίρεσης δια 2

Εμφάνιση του αποτελέσματος στην θύρα Β

*Το πρόγραμμα θα γραφεί στον CCS Compiler(askisi-1b.c) και θα μεταφραστεί σε γλώσσα μηχανής(askisi-1b.hex). Το αρχείο που περιέχει το πρόγραμμα σε γλώσσα μηχανής(askisi-1b.hex) θα φορτωθεί στη μνήμη προγράμματος του μικροελεγκτή.*

Φάκελος όπου θα αναπτυχθεί το Project της άσκησης 1α

\Έγγραφα\mikro-2\tmima-1\askisi-1\askisi-1b\

**askisi-1a.c**

Το αρχείο σε γλώσσα C

(Το αρχείο του προγράμματος που δημιουργήσαμε)

**18F4550.h**

Αρχείο με πληροφορίες για τον

Μικροελεγκτή που χρησιμοποιούμε.

Ανοίγει με notepad

**askisi-1a.h**

Είναι το αποτέλεσμα της μετάφρασης του προγράμματος από γλώσσα c σε γλώσσα μηχανής. Αυτό είναι το αρχείο που θα φορτωθεί στον μικροελεγκτή

Ανοίγει με το notepad

main.h

Αρχείο με αρχικές ρυθμίσεις.

Ανοίγει με notepad

**add.txt**

Αυτό το αρχείο περιέχει μια γραμμή που θα πρέπει να προσθέτουμε κάθε φορά που δημιουργείται νέο αρχείο askisi-1a.hex στην πρώτη γραμμή (είναι πρόβλημα του προγράμματος.

Ανοίγει με notepad

Πρόγραμμα:

**Προσοχή: Στον φάκελο στον οποίο θα αναπτύξετε το project θα πρέπει να τοποθετήσετε τα αρχεία main.h , 18f4550.h και add.txt.**

#include<main.h> //Το αρχείο <main.h> περιέχει αρχικές ρυθμίσεις

//Πρέπει να τοποθετηθεί οπωσδήποτε στον ίδιο φάκελο στον οποίο θα

//αναπτύξετε το project σας.

#byte PORTA=0xF80 //F80 είναι η θέση τη καταχωρητή δεδομένων της πόρτας Α

// στην μνήμη του μικροελεγκτή

#byte PORTB=0xF81 //F81 είναι η θέση τη καταχωρητή δεδομένων της πόρτας Β

// στην μνήμη του μικροελεγκτή

#byte PORTC=0xF82 //F82 είναι η θέση τη καταχωρητή δεδομένων της πόρτας C

// στην μνήμη του μικροελεγκτή

#byte PORTD=0xF83 //F83 είναι η θέση τη καταχωρητή δεδομένων της πόρτας D

// στην μνήμη του μικροελεγκτή

#byte PORTΕ=0xF84 //F84 είναι η θέση τη καταχωρητή δεδομένων της πόρτας E

// στην μνήμη του μικροελεγκτή

// \*\*\*\*\*\*\*\*\*Από εδώ αρχίζει το κύριο πρόγραμμα\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void main()

{ //άνοιγμα αγκύλης της συνάρτησης main

set\_tris\_b(0x00); //Η θύρα Β γίνεται έξοδος(καταχωρητής κατεύθυνσης=0000 0000)

set\_tris\_d(0xff); //Η θύρα D γίνεται είσοδος(καταχωρητής κατεύθυνσης=1111 1111)

int a; //Ορισμός ακέραιης μεταβλητής a

int onoma=5; //Ορισμός ακέραιης μεταβλητής onoma. Απόδοση τιμής 5

int eponymo=7; //Ορισμός ακέραιης μεταβλητής eponymo. Απόδοση τιμής 7

// Με την παρακάτω δομή while(TRUE){ } εκτελείται για αενάως(για πάντα) το σύνολο των

// εντολών που είναι μέσα στις αγκύλες. Η δεσμευμένη λέξη TRUE στη γλώσσα C αντιστοιχεί

// στην αληθή συνθήκη. Αντί για TRUE θα μπορούσαμε για παράδειγμα να βάλουμε 5>1,

// δηλαδή μια συνθήκη που ισχύει πάντα.

while(TRUE) { //Βρόχος που δεν τελειώνει ποτέ(συνθήκη πάντα αληθής)

a=PORTD // Μεταφορά της τιμής της πόρτας D στην μεταβλητή a

PORTB=a/2; // Η πόρτα Β παίρνει το μισό της μεταβλητής a

} //κλείσιμο της αγκύλης του while

} // κλείσιμο της αγκύλης του main

Συμβουλές:

1. Τοποθετείται σχόλια στα προγράμματα σας. Μόνο έτσι όταν θα τα δείτε μετά από καιρό θα μπορείτε να θυμηθείτε πως λειτουργούν. Και μόνο έτσι θα μπορεί κάποιος άλλος να καταλάβει τι έχετε κάνει.
2. Την αγκύλη που κλείνει βάζετε την στην ίδια θέση με την αντίστοιχη αγκύλη που ανοίγει. Έτσι μπορείτε να ελέγξετε ότι έχουν τοποθετηθεί σωστά τα αντίστοιχα ζευγάρια αγκυλών.
3. Καλό είναι να δίνετε στις μεταβλητές ονόματα που σας θυμίζουν την σημασία των μεταβλητών.