**Εργαστήριο Προηγμένοι Μικροεπεξεργαστές**

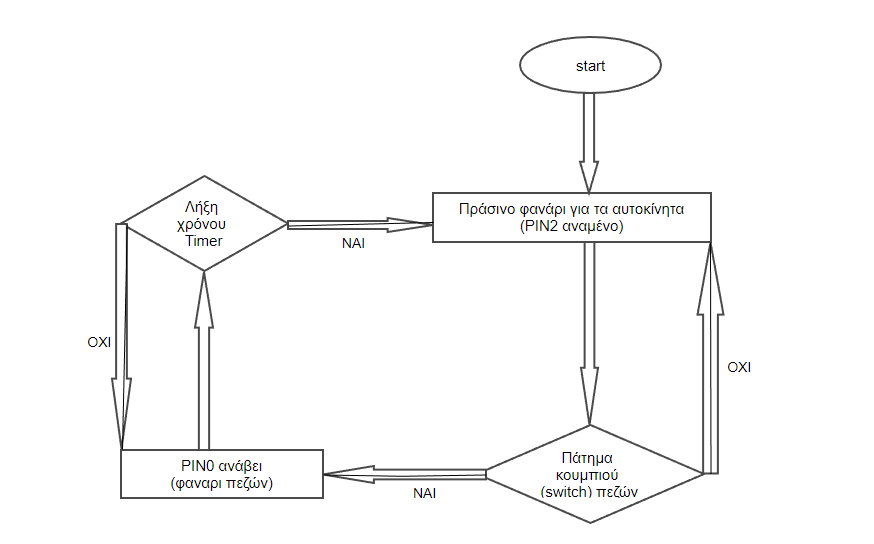
Εργαστηριακή Άσκηση 1

Σκάγκος Ιωάννης 1072611

Στεργίου Γεώργιος 1072503

**Ερώτημα 1**

Διάγραμμα Ροής:



Κώδικας ερωτήματος 1:

#include <avr/io.h>

#include <avr/interrupt.h>

#define T1 10

#define T2 20

#define T3 30

int x=0;

int main(void)

{

PORTD.DIR |= PIN0\_bm;

PORTD.DIR |= PIN2\_bm;

while(1){

PORTF.PIN5CTRL |= PORT\_PULLUPEN\_bm | PORT\_ISC\_BOTHEDGES\_gc;

sei(); //energopoihsh interrupts

PORTD.OUT |=PIN0\_bm; //svhnei fanari pezwn

PORTD.OUTCLR = PIN2\_bm; //anavei fanari amajiwn

while(x==1){

PORTD.OUT |=PIN2\_bm; //svhnei fanari amajiwn

PORTD.OUTCLR = PIN0\_bm; //anavei fanari pezwn

}

cli(); //apenergopoihsh interrupts

}

}

ISR(PORTF\_PORT\_vect)

{

//clear interrupt flag

int y = PORTF.INTFLAGS;

PORTF.INTFLAGS=y;

TCA0.SINGLE.CNT = 0; //clear counter

//Normal Mode

TCA0.SINGLE.CTRLB = 0;

TCA0.SINGLE.CMP0 = T2; //gia na einai gia T2

TCA0.SINGLE.CTRLA = 0x7<<1; //TCA\_SINGLE\_CLKSEL\_DIV1024\_gc σελ 224

TCA0.SINGLE.CTRLA |=1;//Enable

TCA0.SINGLE.INTCTRL = TCA\_SINGLE\_CMP0\_bm; //Interrupt Enable (=0x10)

sei();

x=1;

}

ISR(TCA0\_CMP0\_vect){

TCA0.SINGLE.CTRLA = 0; //Disable

//clear flag

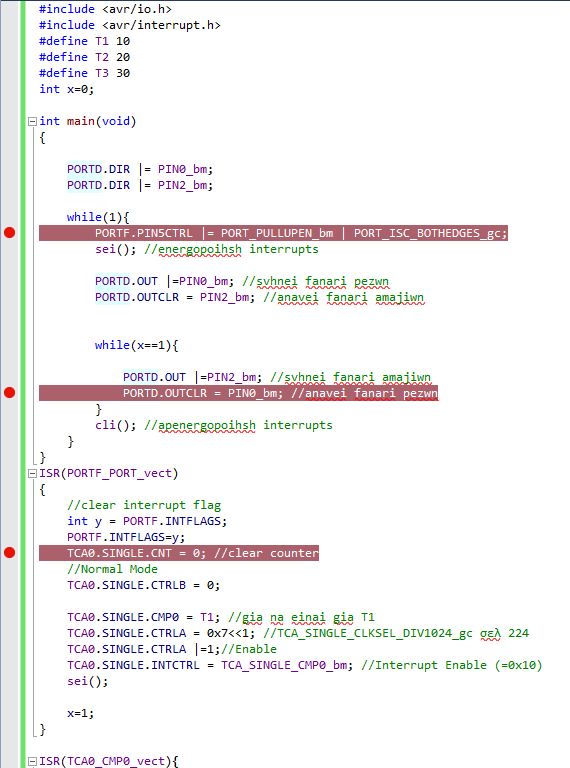
int flags = TCA0.SINGLE.INTFLAGS;

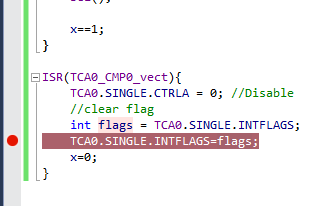
TCA0.SINGLE.INTFLAGS=flags;

x=0;

}

Screenshots με breakout points:





Σχόλια – Επεξήγηση κώδικα:

Αρχικά όπως αναφέρεται στις διαφάνεια παρουσίασης της πρώτης Εργαστηριακής άσκησης θεωρούμε το PIN0 ως το φανάρι των πεζών και το PIN2 ως το φανάρι των αυτοκινήτων και θέτουμε μια μεταβλητή x με αρχική τιμή 0 (x=0).

Στο πρώτο while loop (while(1)) κάνουμε enable το pin5 του portf (που χρησημοποιούμε ως switch, κουμπι πεζών) και τα interrupts. Το PIN2 θεωρούμε ότι είναι αρχικά αναμένο (πρασινο για τα αυτοκίνητα) και γιαυτό το ενεργοποιούμε ενώ απενεργοποιούμε το PIN0 των πεζών ώστε κάθε φορα που βγαίνει από το while(x==1) loop να σβήνει το φανάρι των πεζών.

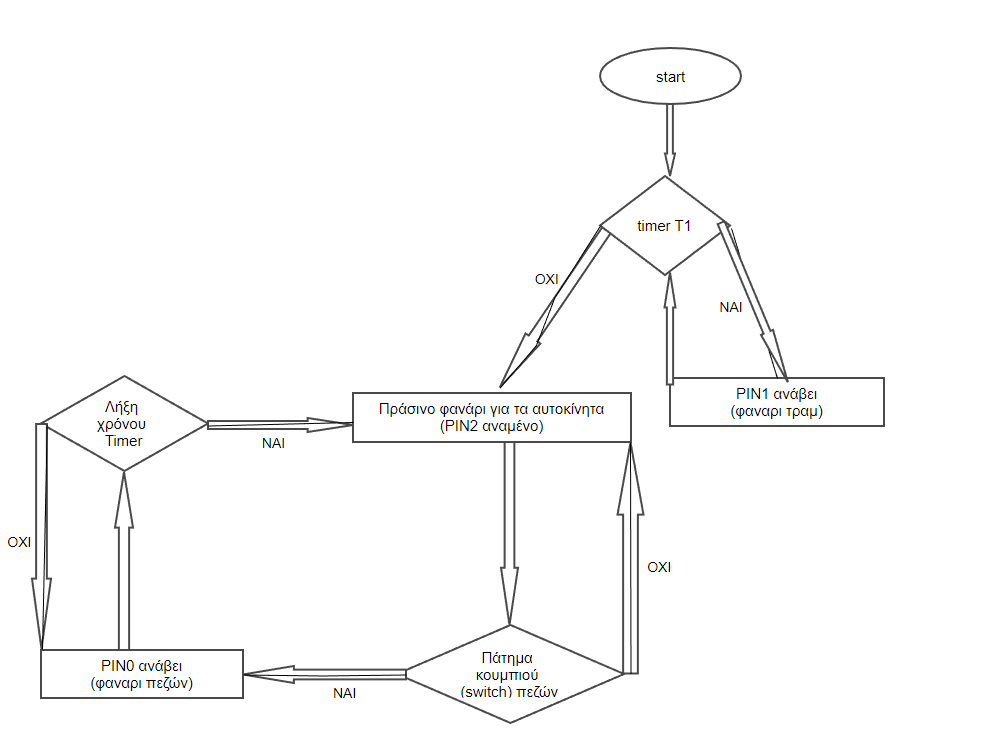
Όταν το κουμπι – switch πατηθεί ενεργοποιήται η πρώτη ISR και θέτει σε λειτουργία τον timer για χρόνο T1 και ύστερα θέτει το x=1.

Αυτό έχει ως συνέπεια το πρόγραμμα να εισέρχεται στην δεύτερη while loop (while(x==1)) όπου εκεί σβήνει το φανάρι των αυτοκινήτων και ανάβει το φανάρι των πεζών.

Όταν ο timer σταματήσει (μετά από T1) ενεργοποιήται η δεύτερη ISR όπου καθαρίζει τις σημαίες του timer και ξαναθέτει το x = 0 και το πρόγραμμα συνεχίζει κανονικά την λειτουργία του εως να ξαναπατηθεί το switch.

**Ερώτημα 2**

Διάγραμμα Ροής:



Κώδικας ερωτήματος 2:

#include <avr/io.h>

#include <avr/interrupt.h>

#define T1 10

#define T2 20

#define T3 30

int x=0;

int t=0;

void init\_timer(void){

TCB0.CCMP = T1; // gia na einai gia T1

TCB0.INTCTRL = TCB\_CAPT\_bm;

TCB0.CTRLA = TCB\_ENABLE\_bm | TCB\_CLKSEL\_CLKDIV1\_gc;

cli(); // disable interrupts

}

int main(void)

{

PORTD.DIR |= PIN0\_bm;

PORTD.DIR |= PIN2\_bm;

PORTD.DIR |= PIN1\_bm;

while(1){

PORTF.PIN5CTRL |= PORT\_PULLUPEN\_bm | PORT\_ISC\_BOTHEDGES\_gc;

sei(); //energopoihsh interrupts

PORTD.OUT |=PIN1\_bm; //svhnei fanari tram

PORTD.OUT |=PIN0\_bm; //svhnei fanari pezwn

PORTD.OUTCLR = PIN2\_bm; //anavei fanari amajiwn

while(x==1){

PORTD.OUT |=PIN2\_bm; //svhnei fanari amajiwn

PORTD.OUTCLR = PIN0\_bm; //anavei fanari pezwn

}

cli(); //apenergopoihsh interrupts

while(t==1){

PORTD.OUT |=PIN2\_bm; //svhnei fanari amajiwn

PORTD.OUT |=PIN2\_bm; //svhnei fanari pezwn

PORTD.OUTCLR = PIN1\_bm; //anavei fanari tram

init\_timer();

}

init\_timer();

}

}

ISR(PORTF\_PORT\_vect)

{

//clear interrupt flag

int y = PORTF.INTFLAGS;

PORTF.INTFLAGS=y;

TCA0.SINGLE.CNT = 0; //clear counter

//Normal Mode

TCA0.SINGLE.CTRLB = 0;

TCA0.SINGLE.CMP0 = T2; //gia na einai gia T2

TCA0.SINGLE.CTRLA = 0x7<<1; //TCA\_SINGLE\_CLKSEL\_DIV1024\_gc σελ 224

TCA0.SINGLE.CTRLA |=1;//Enable

TCA0.SINGLE.INTCTRL = TCA\_SINGLE\_CMP0\_bm; //Interrupt Enable (=0x10)

sei();

x=1;

}

ISR(TCA0\_CMP0\_vect){

TCA0.SINGLE.CTRLA = 0; //Disable

//clear flag

int flags = TCA0.SINGLE.INTFLAGS;

TCA0.SINGLE.INTFLAGS=flags;

x=0;

}

ISR(TCB0\_INT\_vect){

t ^= 1; // toggle t

TCB0.INTFLAGS = TCB\_CAPT\_bm; // katarismos interrupt flag

}

Σχόλια – Επεξήγηση κώδικα:

Η ιδεα πισω από τον κώδικα το ερωτηματος 2 ήταν να προστεθεί ένας ακόμη timer ο οποίος είναι μόνιμα ενεργοποιημένος (υλοποίηση στην συναρτηση timer) και όταν περάσει το χρονικό διάστημα Τ1 δηλαδή είναι η στιγμή που περνάει το τραμ η ISR κάνει toggle την τιμή του t ώστε να το πρόγραμμα να μπεί η να βγεί από την while(t==1) οπου εκεί σβηνουν τα υπολοιπα φαναρια και ανάβει μονο αυτό του τραμ (θεωρήσαμε ότι περναει κάθε T3 για χρονικο διάστημα T3).Η συναρτηση timer καλείτε και μεσα και εξω από την while ώστε να μην «εγκλοβίζεται το πρόγραμμα».Παρόλαυτα γνωρίζουμε ότι δεν είναι σωστή λύση αλλά κάναμε ότι πιο λειτουργικό μπορούσαμε.