

Εργαστήριο Προηγμένοι Μικροεπεξεργαστές

**1η Εργαστηριακή Άσκηση : Φανάρια Κυκλοφορίας**

***Μέλη Ομάδας :***

***Μουσελέ Χριστιάνα ΑΜ: 1090068***

***Μπαλάσης Ιωάννης ΑΜ: 1084631***

***Πίνακας περιεχομένων***

[**Κώδικας** 2](#_Toc161073091)

[**Περιγραφή Κώδικα** 4](#_Toc161073092)

[**Διάγραμμα Ροής** 6](#_Toc161073093)

# **Κώδικας**

#include <avr/io.h>

#include <util/delay.h>

#include <avr/interrupt.h>

volatile *uint8\_t* tramPassed = 0;

volatile *uint8\_t* pedestrianButtonPressed = 0;

volatile *uint8\_t* pedestrianPassing = 0;

volatile *uint8\_t* T2\_started = 0;

volatile *uint8\_t* T3\_started = 0;

#define T2 5000 //time for pedestrians to pass the road

#define T1 20000 // time where the train passes

#define T3 15000 //time where the pedestrian button can be pressed again

int main(void){

PORTD.DIR |= 0b00000001; //PIN0\_bm //PIN is output //pedestrians

PORTD.OUT |= 0b00000001; //PIN0\_bm //LED is off

PORTD.DIR |= 0b00000010; //PIN1\_bm //PIN is output //tram

PORTD.OUT |= 0b00000010; //PIN1\_bm //LED is off

PORTD.DIR |= 0b00000100; //PIN2\_bm //PIN is output //road

PORTD.OUT |= 0b00000100; //PIN2\_bm //LED is off

TCA0.SPLIT.CTRLD = TCA\_SPLIT\_SPLITM\_bm; // SPLIT MODE

// Configure TCA0 for T1 (train passes)

TCA0.SPLIT.CTRLB = TCA\_SPLIT\_LCMP0EN\_bm | TCA\_SPLIT\_HCMP0EN\_bm; // Enable comparators

TCA0.SPLIT.LPER = 20; // Set LPRE for T1 to 20 milliseconds

TCA0.SPLIT.HPER = 0; // Not used for T1

// Configure TCA0 for T2 and T3 (pedestrian button and time to pass)

TCA0.SPLIT.LCMP0 = 15; // Set LCMP0 for T3 to 15 milliseconds

TCA0.SPLIT.CTRLA = TCA\_SPLIT\_CLKSEL\_DIV1024\_gc | TCA\_SPLIT\_ENABLE\_bm; // Set prescaler and enable TCA0

TCA0.SPLIT.INTCTRL |= TCA\_SPLIT\_LCMP1\_bm; // Enable T1

//pull up enable and Interrupt enabled with sense on both edges

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

sei(); //enable interrupts

while (1) {

if (pedestrianButtonPressed && !T3\_started) {

TCA0.SPLIT.CTRLB |= TCA\_SPLIT\_HCMP0EN\_bm; // Enable T2

T2\_started = 1;

T3\_started = 0;

PORTD.OUTCLR = 0b00000001; // LED is on, open only for pedestrians

PORTD.OUT |= 0b00000100; // LED is off

PORTD.OUT |= 0b00000010; // LED is off

TCA0.SPLIT.INTCTRL |= TCA\_SPLIT\_LCMP0\_bm; // Enable T3

pedestrianButtonPressed = 0;

}

else {

PORTD.OUTCLR = 0b00000100; // LED is on, open only for road

PORTD.OUT |= 0b00000010; // LED is off

PORTD.OUT |= 0b00000001; // LED is off

}

}

cli();

}

ISR(PORTF\_PORT\_vect){

int y = PORTF.INTFLAGS; //Procedure to

PORTF.INTFLAGS=y;

pedestrianButtonPressed=1;

}

//T1

ISR(TCA0\_LCMP1\_vect) {

PORTD.OUTCLR= 0b00000001;//PIN0\_bm //LED is on //led is open for pedestrians

PORTD.OUTCLR= 0b00000010;//PIN0\_bm //LED is on //led is open for tram

PORTD.OUT |= 0b00000100; //PIN2\_bm //LED is off // led is off for road

}

// T2

ISR(TCA0\_HCMP0\_vect) {

pedestrianPassing = 1; // Τ2

}

//T3

ISR(TCA0\_LCMP0\_vect) {

pedestrianButtonPressed = 0;

T3\_started = 0;

}

# **Περιγραφή Κώδικα**

Αρχικά δηλώνουμε τις μεταβλητές : volatile *uint8\_t* tramPassed = 0; volatile *uint8\_t* pedestrianButtonPressed = 0; volatile *uint8\_t* pedestrianPassing = 0;

volatile *uint8\_t* T2\_started = 0; volatile *uint8\_t* T3\_started = 0;

ως **volatile *uint8\_t***έτσι ώστε να μπορούν να αλλάξουν τιμές στις διακοπές που χρησιμοποιούμε .

Έπειτα δηλώνουμε τους χρόνους Τ1 , Τ2 , Τ3 όπως ορίζεται από την εκφώνηση.

Στη συνέχεια αρχικοποιούμε τα pin0 , pin1 , pin2 του PORTD ως εξόδους. Μετά ορίζουμε τον χρονοδιακόπτη TCA0 σε λειτουργία SPLIT. Ρυθμίζουμε τον χρονοδιακόπτη έτσι ώστε να μετράει για Τ1 και Τ2,Τ3 αντίστοιχα. Έπειτα ενεργοποιούμε τον Τ1 για την διέλευση του τραμ. Ορίζουμε ως διακόπτη το pin5 του portF για την λειτουργεία του κουμπιού των πεζών και ενεργοποιούμε τις διακοπές . Οι χρονιστές έχουν και τις αντίστοιχες διακοπές τους. Πρώτα απ’ όλα έχουμε την διακοπή που προκαλείται από το πάτημα του κουμπιού των πεζών :

ISR(PORTF\_PORT\_vect){

int y = PORTF.INTFLAGS;

PORTF.INTFLAGS=y; // μηδενίζουμε πάλι το bit 5

pedestrianButtonPressed=1; // Θέτουμε την τιμή 1 στην μεταβλητή

}

Η δεύτερη διακοπή που έχουμε είναι αυτή που προκαλείται από τον χρονιστή T1 , δηλαδή όταν περνάει το τράμ :

ISR(TCA0\_LCMP1\_vect) {

PORTD.OUTCLR= 0b00000001;//PIN0\_bm //LED is on //led is open for pedestrians

PORTD.OUTCLR= 0b00000010;//PIN0\_bm //LED is on //led is open for tram

PORTD.OUT |= 0b00000100; //PIN2\_bm //LED is off // led is off for road

}

Η τρίτη διακοπή που έχουμε είναι αυτή που προκαλείται από τον Τ2 , ο οποίος μετράει τον χρόνο που είναι πράσινο το φανάρι των πεζών και κόκκινο των αυτοκινήτων αντίστοιχα :

ISR(TCA0\_HCMP0\_vect) {

pedestrianPassing = 1; // θετουμε 1 την τιμη

}

Η τέταρτη και τελευταία διακοπή είναι αυτή που προκαλείται από τον χρονιστή Τ3 , ο οποίος μετράει τον ελάχιστο χρόνο που πρέπει να περάσει ετσι ώστε να μπορέσει ο πεζός να ξαναπατήσει το κουμπί :

ISR(TCA0\_LCMP0\_vect) {

pedestrianButtonPressed = 0;

T3\_started = 0;

}

Όπως παρατηρούμε μέσα στις διακοπές έχουμε θέσει κάποιες τιμές σε διαφορετικές μεταβλητές όπου η λειτουργεία τους θα φανεί στην κύρια λούπα του προγράμματος μας .

Αρχικά τα φανάρια είναι όλα κλειστά( από αρχικοποίηση) . Όταν εισέλθει το πρόγραμμα στην λούπα πρώτα πηγαίνει στο if (pedestrianButtonPressed && !T3\_started). Αφού όμως δεν ισχύουν οι προϋποθέσεις εκτελείται το : else {

PORTD.OUTCLR = 0b00000100; // LED is on, open only for road

PORTD.OUT |= 0b00000010; // LED is off

PORTD.OUT |= 0b00000001; // LED is off

}

Όπου έχουμε πράσινο φανάρι για τον δρόμο και κόκκινο για πεζούς και τράμ . Αυτό ισχύει μέχρι να γίνει κάποια διακοπή από τον Τ1 και ανάψει πράσινο για τραμ και πεζούς και κόκκινο για δρόμο . Όταν όμως ο πεζός πατήσει το κουμπί θα προκληθεί η πρώτη διακοπή και θα θέσουμε την τιμή 1 στην μεταβλητή pedestrianButtonPressed=1;. Αυτό σημαίνει ότι θα ισχύουν και οι δύο προϋποθέσεις του if statement που αναφέραμε παραπάνω , άρα θα εκτελεστεί και ο αντίστοιχος κώδικας :

TCA0.SPLIT.CTRLB |= TCA\_SPLIT\_HCMP0EN\_bm; // Enable T2

T2\_started = 1;

T3\_started = 0;

PORTD.OUTCLR = 0b00000001; // LED is on, open only for pedestrians

PORTD.OUT |= 0b00000100; // LED is off

PORTD.OUT |= 0b00000010; // LED is off

TCA0.SPLIT.INTCTRL |= TCA\_SPLIT\_LCMP0\_bm; // Enable T3

pedestrianButtonPressed = 0;

Αρχικά ενεργοποιούμε τον μετρητή Τ2 έτσι ώστε το πράσινο φανάρι για τους πεζούς να μείνει αναμμένο για την κατάλληλη ώρα . Ανάβουμε πράσινο φανάρι για πεζούς και κόκκινο για τραμ και αυτοκίνητα. Ενεργοποιούμε τον Τ3 χρονιστή έτσι ώστε να ελέγχουμε το χρονικό όριο πριν μπορέσει κάποιος να πατήσει πάλι το κουμπί των πεζών. Τέλος αρχικοποιούμε πάλι την τιμή pedestrianButtonPressed στο 0 .

Όσο αφορά την αρχικοποίηση των χρονιστών έχουμε :

Χρονιστής Τ1 :

LPRE = 20000/1024 ~= 20 ( καθώς έχουμε επιλέξει διαίρεση 1024 στην επιλογή CLKSEL)

Χρονιστής Τ2 και Τ3 :

HPER = 5000/1024 ~=5

LCMP0=15000/1024~=15

# **Διάγραμμα Ροής**

