**ΕΡΓΑΣΤΗΡΙΟ ΜΙΚΡΟΕΠΕΞΕΡΓΑΣΤΩΝ**

**ΠΡΩΤΗ ΑΣΚΗΣΗ**

Στοιχεία μελών ομάδας:

1ο μέλος ομάδας

Ονοματεπώνυμο: Γκουργκούτας Βασίλειος

ΑΜ: 1084667

Έτος: 4ο

2ο μέλος ομάδας

Ονοματεπώνυμο: Κοντογιώργος Αναστάσιος

ΑΜ: 1090084

Έτος: 4ο

**Ερωτήματα Εργαστηριακής Άσκησης 1**

**1.** Ο κώδικας που υλοποιήθηκε για την λειτουργία της διαχείρισης των φαναριών του μεγάλου δρόμου και των πεζών είναι ο εξής:

| #include <avr/io.h> #include <util/delay.h> #include <avr/interrupt.h>  #define T2 20 #define T3 30 int button=0; //Button which will enable the green light for the pedestrians int change=0; //A variable used to wait for an amount of T3 time before the pedestrians can press the button again int tram\_pass=0;  int main(void){    PORTD.DIR |= PIN0\_bm; //PIN0\_bm //PIN is output // Pin Pedestrians  PORTD.OUT |= PIN0\_bm; //PIN0\_bm //LED is off   //PORTD.DIR |= PIN1\_bm; //PIN1\_bm //PIN is output // Pin Tram  //PORTD.OUT |= PIN1\_bm; //PIN1\_bm //LED is off    PORTD.DIR |= PIN2\_bm; //PIN2\_bm //PIN is output // Pin Cars  PORTD.OUT |= PIN2\_bm; //PIN2\_bm //LED is off   //pullup enable and Interrupt enabled with sense on both edges  PORTF.PIN5CTRL |= PORT\_PULLUPEN\_bm | PORT\_ISC\_BOTHEDGES\_gc;    sei(); //begin accepting interrupt signals  while (1) {    if(button==0) // If the button==0, then the cars have green light and the pedestrians have red light  {  PORTD.OUT |= PIN0\_bm; //LED is off // The pedestrians have red light  PORTD.OUTCLR= PIN2\_bm; //Led is on // The cars have green light  }  else // Else if the button==1, then the cars have red light and the pedestrians have green light  {  PORTD.OUT |= PIN2\_bm; //LED is off // Cars have red light  PORTD.OUTCLR= PIN0\_bm; //Led is on // Pedestrians have green light    TCA0.SPLIT.HCNT = 0; //clear counter for high register  TCA0.SPLIT.HCMP0 = T2;//When HCMP0 reaches this value -> interrupt  TCA0.SPLIT.CTRLB = 0; //Normal Mode (TCA\_SINGLE\_WGMODE\_NORMAL\_gc)  TCA0.SPLIT.CTRLD = TCA\_SPLIT\_SPLITM\_bm; //ENABLE TCA SPLIT MODE  TCA0.SPLIT.CTRLB = TCA\_SPLIT\_LCMP0EN\_bm|TCA\_SPLIT\_HCMP0EN\_bm;   //CLOCK\_FREQUENCY/1024  TCA0.SPLIT.CTRLA = TCA\_SPLIT\_CLKSEL\_DIV1024\_gc | TCA\_SPLIT\_ENABLE\_bm; //TCA\_SPLIT\_CLKSEL\_DIV1024\_gc  TCA0.SPLIT.CTRLA |= 1; //Enable    TCA0.SPLIT.INTCTRL =0x02; // HUNF INTERRUPT ENABLE    while (change==0) // We wait for the timer to finish, which means that the time period T2 has passed  {   ;  }    change=0;  PORTD.OUT |= PIN0\_bm; //The Pedestrians have red light  PORTD.OUTCLR= PIN2\_bm; //Led is on // The Cars have green light  TCA0.SPLIT.HCNT = 0; //clear counter for high register  TCA0.SPLIT.HCMP0 = T3;//When HCMP0 reaches this value -> interrupt    TCA0.SPLIT.CTRLD = TCA\_SPLIT\_SPLITM\_bm; //ENABLE TCA SPLIT MODE  TCA0.SPLIT.CTRLB = TCA\_SPLIT\_LCMP0EN\_bm|TCA\_SPLIT\_HCMP0EN\_bm;   //CLOCK\_FREQUENCY/1024  TCA0.SPLIT.CTRLA = TCA\_SPLIT\_CLKSEL\_DIV1024\_gc | TCA\_SPLIT\_ENABLE\_bm; //TCA\_SINGLE\_CLKSEL\_DIV1024\_gc  TCA0.SPLIT.CTRLA |= 1; //Enable  TCA0.SPLIT.INTCTRL =0x02; // HUNF INTERRUPT ENABLE    while (change==0) // The Pedestrians are waiting for a sort amount of time T3 before they can hit the button again  {   ;  }    change=0;  button=0; // We change the button to 0, so that the pedestrians can press the button again  }    }  cli(); }  ISR(PORTF\_PORT\_vect){ // Interrupt occurs with the press of the button  int y = PORTF.INTFLAGS; //Procedure to  PORTF.INTFLAGS=y; //clear the interrupt flag  button=1; //change logic flag to get out of loop }  ISR(TCA0\_LCMP0\_vect){ // Interrupt occurs after a sort amount of time T1  TCA0.SPLIT.CTRLA = 0; //Disable  int intflags = TCA0.SPLIT.INTFLAGS; //Procedure to  TCA0.SPLIT.INTFLAGS=intflags; //clear interrupt flag   }  ISR(TCA0\_HUNF\_vect){ // Interrupt occurs after a sort amount of time T2,T3  TCA0.SPLIT.CTRLA = 0; //Disable  int intflags = TCA0.SPLIT.INTFLAGS;  TCA0.SPLIT.INTFLAGS=intflags;  change=1; //we are changing the variable to change==1 to get out of the loop  } |
| --- |

**2.** Ο κώδικας που υλοποιήθηκε για την λειτουργία της διαχείρισης των φαναριών του μεγάλου δρόμου και των πεζών και την λειτουργία διαχείρισης του τραμ και του χρονικού διαστήματος ανά το οποίο περνάει από το συγκεκριμένο σημείο του σιδηρόδρομου είναι ο εξής:

| #include <avr/io.h> #include <util/delay.h> #include <avr/interrupt.h>  #define T1 40 // Time that takes until the Tram passes  #define T2 20 // Time that the Cars have green light #define T3 30 // Time that the Pedestrians have green light int button=0; //Button which will enable the green light for the pedestrians int change=0; //A variable used to wait for an amount of T3 time before the pedestrians can press the button again int tram\_pass=0; // Variable that is used when the tram passes after a sort amount of time T1  int main(void){    PORTD.DIR |= PIN0\_bm; //PIN0\_bm //PIN is output // Pin Pedestrians  PORTD.OUT |= PIN0\_bm; //PIN0\_bm //LED is off   PORTD.DIR |= PIN1\_bm; //PIN1\_bm //PIN is output // Pin Tram  PORTD.OUT |= PIN1\_bm; //PIN1\_bm //LED is off    PORTD.DIR |= PIN2\_bm; //PIN2\_bm //PIN is output // Pin Cars  PORTD.OUT |= PIN2\_bm; //PIN2\_bm //LED is off   //pullup enable and Interrupt enabled with sense on both edges  PORTF.PIN5CTRL |= PORT\_PULLUPEN\_bm | PORT\_ISC\_BOTHEDGES\_gc;    sei(); //begin accepting interrupt signals  while (1) {    TCA0.SPLIT.CTRLD = TCA\_SPLIT\_SPLITM\_bm; //ENABLE TCA SPLIT MODE  TCA0.SPLIT.CTRLB = TCA\_SPLIT\_LCMP0EN\_bm|TCA\_SPLIT\_HCMP0EN\_bm;  TCA0.SPLIT.LCNT = 0; //clear counter for low register  TCA0.SPLIT.LCMP0 = T1;//When LCMP0 reaches this value -> interrupt  TCA0.SPLIT.CTRLA = TCA\_SPLIT\_CLKSEL\_DIV1024\_gc | TCA\_SPLIT\_ENABLE\_bm;  TCA0.SPLIT.INTCTRL = TCA\_SINGLE\_CMP0\_bm; //Interrupt Enable (=0x10)  sei();  while (tram\_pass==0)   {  sei();  if(button==0) // If the button==0, then the cars have green light and the pedestrians have red light  {  PORTD.OUT |= PIN0\_bm; //LED is off // The pedestrians have red light  PORTD.OUTCLR= PIN2\_bm; //Led is on // The cars have green light  }  else // Else if the button==1, then the cars have red light and the pedestrians have green light  {  PORTD.OUT |= PIN2\_bm; //LED is off // Cars have red light  PORTD.OUTCLR= PIN0\_bm; //Led is on // Pedestrians have green light    TCA0.SPLIT.HCNT = 0; //clear counter for high register  TCA0.SPLIT.HCMP0 = T2;//When HCMP0 reaches this value -> interrupt  TCA0.SPLIT.CTRLB = 0; //Normal Mode (TCA\_SINGLE\_WGMODE\_NORMAL\_gc)  TCA0.SPLIT.CTRLD = TCA\_SPLIT\_SPLITM\_bm; //ENABLE TCA SPLIT MODE  TCA0.SPLIT.CTRLB = TCA\_SPLIT\_LCMP0EN\_bm|TCA\_SPLIT\_HCMP0EN\_bm;   //CLOCK\_FREQUENCY/1024  TCA0.SPLIT.CTRLA = TCA\_SPLIT\_CLKSEL\_DIV1024\_gc | TCA\_SPLIT\_ENABLE\_bm; //TCA\_SINGLE\_CLKSEL\_DIV1024\_gc  TCA0.SPLIT.CTRLA |= 1; //Enable    TCA0.SPLIT.INTCTRL =0x02; // HUNF INTERRUPT ENABLE  sei();  while (change==0) // We wait for the timer to finish, which means that the time period T2 has passed  {  ;  }  cli();  change=0;  PORTD.OUT |= PIN0\_bm; //The Pedestrians have red light  PORTD.OUTCLR= PIN2\_bm; //Led is on // The Cars have green light  TCA0.SPLIT.HCNT = 0; //clear counter for high register  TCA0.SPLIT.HCMP0 = T3;//When HCMP0 reaches this value -> interrupt    TCA0.SPLIT.CTRLD = TCA\_SPLIT\_SPLITM\_bm; //ENABLE TCA SPLIT MODE  TCA0.SPLIT.CTRLB = TCA\_SPLIT\_LCMP0EN\_bm|TCA\_SPLIT\_HCMP0EN\_bm;   //CLOCK\_FREQUENCY/1024  TCA0.SPLIT.CTRLA = TCA\_SPLIT\_CLKSEL\_DIV1024\_gc | TCA\_SPLIT\_ENABLE\_bm; //TCA\_SINGLE\_CLKSEL\_DIV1024\_gc  TCA0.SPLIT.CTRLA |= 1; //Enable  TCA0.SPLIT.INTCTRL =0x02; // HUNF INTERRUPT ENABLE  sei();  while (change==0) // The Pedestrians are waiting for a sort amount of time T3 before they can hit the button again  {  ;  }  cli();  change=0;  button=0; // We change the button to 0, so that the pedestrians can press the button again  }  cli();  }  cli();  if(tram\_pass==1) // This part of the code is executed, when the Tram passes after time T1  {  PORTD.OUT |= PIN2\_bm; // The light will change to red for the cars  PORTD.OUTCLR = PIN0\_bm; // The light will change to green for the pedestrians  PORTD.OUTCLR = PIN1\_bm; // The light will change to green for the Tram    TCA0.SPLIT.HCNT = 0; //clear counter for high register  TCA0.SPLIT.HCMP0 = T2;//When HCMP0 reaches this value -> interrupt  TCA0.SPLIT.CTRLB = 0; //Normal Mode (TCA\_SINGLE\_WGMODE\_NORMAL\_gc)  TCA0.SPLIT.CTRLD = TCA\_SPLIT\_SPLITM\_bm; //ENABLE TCA SPLIT MODE  TCA0.SPLIT.CTRLB = TCA\_SPLIT\_LCMP0EN\_bm|TCA\_SPLIT\_HCMP0EN\_bm;   //CLOCK\_FREQUENCY/1024  TCA0.SPLIT.CTRLA = TCA\_SPLIT\_CLKSEL\_DIV1024\_gc | TCA\_SPLIT\_ENABLE\_bm; //TCA\_SPLIT\_CLKSEL\_DIV1024\_gc  TCA0.SPLIT.CTRLA |= 1; //Enable    TCA0.SPLIT.INTCTRL =0x02; // HUNF INTERRUPT ENABLE  sei();  while (change==0) // We wait for a sort amount of time until both Tram and Pedestrians cross the road for time T2  {  ;  }  cli();  tram\_pass=0;  change=0;  PORTD.OUT |=PIN0\_bm; // The light turns red for the Pedestrians  PORTD.OUT |=PIN1\_bm; // The light turns red for the Tram  PORTD.OUTCLR = PIN2\_bm; // The light turns green for the cars  }  cli();  }     }  ISR(PORTF\_PORT\_vect){ // Interrupt occurs with the press of the button  int y = PORTF.INTFLAGS; //Procedure to  PORTF.INTFLAGS=y; //clear the interrupt flag  button=1; //change logic flag to get out of loop }  ISR(TCA0\_LCMP0\_vect){ // Interrupt occurs after a sort amount of time T1  TCA0.SPLIT.CTRLA = 0; //Disable  int intflags = TCA0.SPLIT.INTFLAGS; //Procedure to  TCA0.SPLIT.INTFLAGS=intflags; //clear interrupt flag  tram\_pass=1; // We change the variable to 1, because the train will pass }  ISR(TCA0\_HUNF\_vect){ // Interrupt occurs after a sort amount of time T2,T3  TCA0.SPLIT.CTRLA = 0; //Disable  int intflags = TCA0.SPLIT.INTFLAGS;  TCA0.SPLIT.INTFLAGS=intflags;  change=1; //we are changing the variable to change==1 to get out of the loop  } |
| --- |

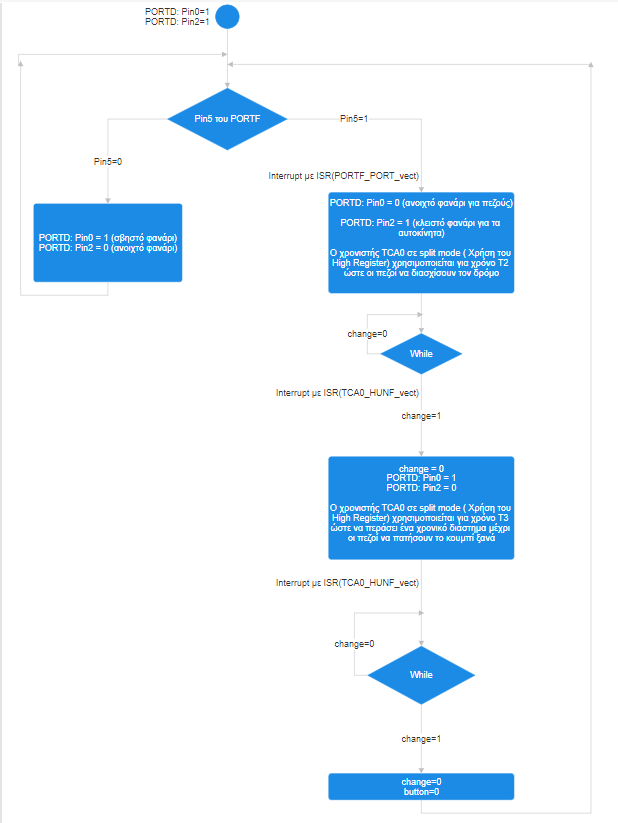
**Επεξήγηση του Κώδικα**

Αρχικά για την υλοποίηση του πρώτου ερωτήματος, ορίζουμε δύο μεταβλητές T2, T3. Η T2 αποτελεί τον χρόνο για τον οποίο οι πεζοί έχουν πράσινο φανάρι και η μεταβλητή T3 αποτελεί το χρονικό διάστημα για το οποίο οι πεζοί δεν έχουν την δυνατότητα να χρησιμοποιήσουν το κουμπί για να ενεργοποιηθεί το πράσινο φανάρι. Έπειτα δηλώνουμε δύο μεταβλητές button και change. Η μεταβλητή button θα χρησιμοποιείται μέσα στην συνθήκη if ενώ η μεταβλητή change θα αξιοποιείται από την ISR διακοπή με χρονιστή. Προχωρώντας στον κώδικα της main, ενεργοποιούμε σαν output pin τα Pin0 και Pin2. Το Pin0 χρησιμοποιείται για τους πεζούς, ενώ το Pin2 χρησιμοποιείται για τα αυτοκίνητα. Επιπρόσθετα κάνουμε pullup enable το pin5 από το PORTF ώστε να το αξιοποιήσουμε σαν κουμπί για το φανάρι των πεζών. Με την χρήση του sei() ξεκινάμε να δεχόμαστε τα σήματα διακοπής. Χρησιμοποιούμε μία συνθήκη if στην οποία όταν το κουμπί δεν είναι πατημένο τότε απενεργοποιούμε το Pin0 των πεζών (Led Off) και ενεργοποιούμε το Pin2 των αυτοκινήτων (Led On). Στην περίπτωση που πατηθεί το Pin5 από τα Intflags του PORTF, τότε ενεργοποιείται η διακοπή και μεταφερόμαστε στην ISR(PORTF\_PORT\_vect). Στην συγκεκριμένη ISR το button γίνεται ίσο με ένα, επομένως έπειτα από την εκτέλεση της ISR μεταφερόμαστε στην εκτέλεση του κώδικα που περικλείεται μέσα στο else. Μέσα στον κώδικα της else αρχικά σβήνουμε το Pin2 των αυτοκινήτων (κόκκινο φανάρι για τα αυτοκίνητα) και ενεργοποιούμε το φανάρι των πεζών, δηλαδή το Pin0. Στην συνέχεια, αξιοποιούμε με το split mode του 16-bit χρονιστή TCA0, σε δύο χρονιστές των 8-bit ο καθένας. Ο ένας εκ των δύο χρονιστών ( High Register) αξιοποιείται ώστε να μετρήσει το χρονικό διάστημα T2 για το οποίο οι πεζοί έχουν πράσινο φανάρι και διασχίζουν τον δρόμο. Όταν γίνεται το compare με την χρονική τιμή T2 και έχει φτάσει ο χρονιστής στην τιμή αυτή, τότε ενεργοποιείται η ISR(TCA0\_HUNF\_vect) η οποία αλλάζει την μεταβλητή change σε άσσο. Αυτό συμβαίνει για να μπορέσουμε να περάσουμε από την while loop η οποία εμποδίζει την συνέχιση του κώδικα εάν δεν περάσει το χρονικό διάστημα T2 και δεν ενεργοποιηθεί η διακοπή. Έπειτα από το πέρας του χρονικού διαστήματος T2, το φανάρι των πεζών ξαναγίνεται κόκκινο και το φανάρι των αυτοκινήτων γίνεται πράσινο. Τότε αξιοποιούμε τον χρονιστή ξανά και συγκεκριμένα με το split mode τον High Register για να μετρήσουμε το χρονικό διάστημα T3. Κατά το χρονικό διάστημα T3 οι πεζοί δεν μπορούν να χρησιμοποιήσουν το κουμπί για να ανάψουν το φανάρι τους σε πράσινο. Έπειτα οι μεταβλητές button και change ξαναγίνονται μηδέν ώστε το φανάρι να μπορεί να ξαναγίνει πράσινο για τα αυτοκίνητα και κόκκινο για τους πεζούς.

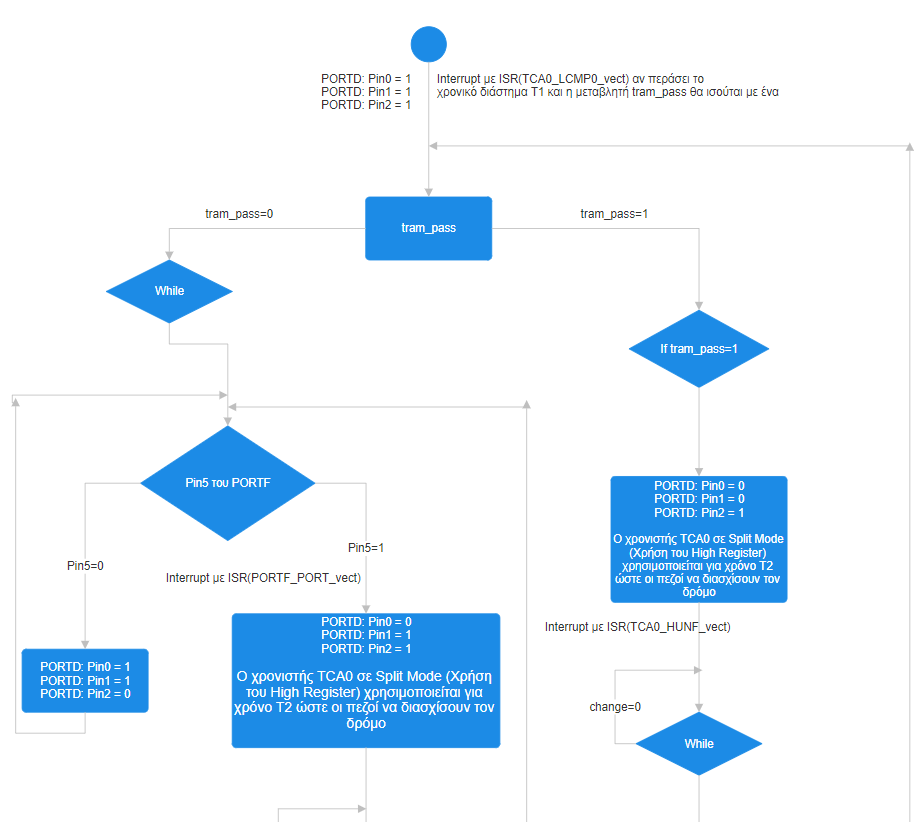
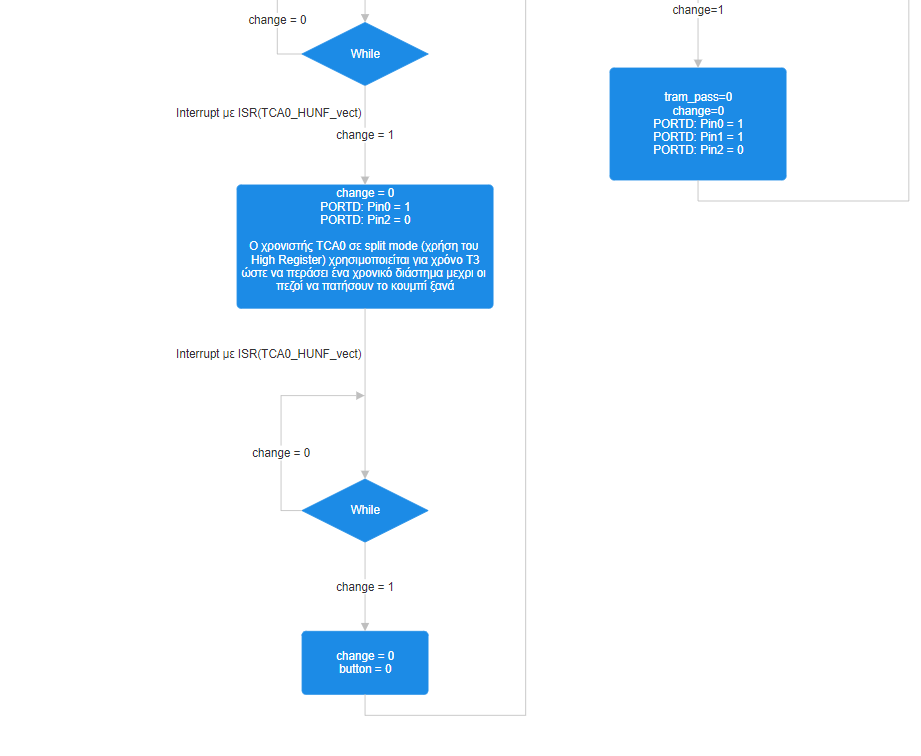
Όσον αφορά τον κώδικα του δεύτερου ερωτήματος για το τράμ, προστέθηκε ένα define το οποίο περιέχει τον χρόνο T1 που ορίζει κάθε πότε διέρχεται το τράμ. Επιπλέον προστέθηκε μία μεταβλητή tram\_pass η οποία χρησιμοποιείται για την εκτέλεση μίας συνθήκης if η οποία αφορά το τραμ. Στην αρχή της main προστέθηκε κώδικας που ορίζει το Pin1 ως output και είναι το φανάρι του τράμ. Στην αρχή της πρώτης while loop ( while(1) ) προστέθηκε κώδικας ο οποίος αφορά τον χρονιστή για το τράμ. Αξιοποιήθηκε ο Low Register και σαν τιμή για τον χρονιστή ορίστηκε το χρονικό διάστημα T1. Μόλις περάσει το χρονικό διάστημα T1, ενεργοποιείται η διακοπή με την ISR (TCA0\_LCMP0\_vect). Μέσα στην συγκεκριμένη ISR η μεταβλητή tram\_pass γίνεται ίση με ένα. Επομένως εκτελείται το κομμάτι της if(tram\_pass==1). Πιο συγκεκριμένα το pin2 με PORTD.OUT μετατρέπεται σε κόκκινο φανάρι για τα αυτοκίνητα ενώ με PORTD.OUTCLR τα pin0 και pin1 μετατρέπονται σε πράσινα φανάρια για τους πεζούς και το τράμ αντίστοιχα. Έπειτα με την χρήση του χρονιστή για χρονικό διάστημα T2 αφήνουμε να παραμείνουν πράσινα τα φανάρια των πεζών και του τράμ. Εφόσο περάσει αυτό το χρονικό διάστημα, ενεργοποιείται το ISR (TCA0\_HUNF\_vect) και αλλάζουμε την μεταβλητή change σε ένα για να μπορέσουμε να προχωρήσουμε στο επόμενο κομμάτι του κώδικά μας. Αλλάζουμε τις μεταβλητές tram\_pass και change σε μηδέν για να μπορούμε να ξαναχρησιμοποιήσουμε τους χρονιστές μας και τις διακοπές τους.Τέλος εφόσον παρέλθει το χρονικό διάστημα T2, τότε αλλάζουμε τις τιμές των pin0 και pin1 με το PORTD.OUT και τα φανάρια γίνονται κόκκινα. Επιπλέον με το PORTD.OUTCLR αλλάζουμε την τιμή του pin2 σε πράσινο φανάρι ξανά αφού πέρασε το χρονικό διάστημα T2.

**Διαγράμματα Ροής**

Το διάγραμμα ροής όσον αφορά το πρώτο ερώτημα για την λειτουργία της διαχείρισης των φαναριών του μεγάλου δρόμου και των πεζών είναι το εξής:

****

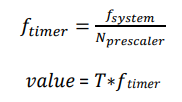
Το διάγραμμα ροής όσον αφορά το δεύτερο ερώτημα με την προσθήκη της λειτουργίας διαχείρισης του τραμ και του χρονικού διαστήματος ανά το οποίο περνάει από το συγκεκριμένο σημείο του σιδηρόδρομου είναι το εξής:

****

**Συ**

**Συχνότητα Ρολογιού**

Με βάση τον τύπο για τον υπολογισμό των τιμών T1, T2 και T3:



και επιπλέον έχουμε ότι

ftimer: Συχνότητα του ρολογιού

value: ορίζει το χρονικό διάστημα μέχρι να κάνει διακοπή (interrupt) ο χρονιστής (timer)

fsystem=20 MHz και Ν = 1024

Από όλα τα παραπάνω οι τιμές που χρησιμοποιήσαμε είναι οι εξής:

T1: 40

T2: 20

T3: 30