



ΠΑΝΕΠΙΣΤΗΜΙΟ
ΠΑΤΡΩΝ
UNIVERSITY OF PATRAS

Πολυτεχνική Σχολή
Τμήμα Μηχανικών Η/Υ & Πληροφορικής

ΕΡΓΑΣΤΗΡΙΟ ΜΙΚΡΟΕΠΕΞΕΡΓΑΣΤΩΝ

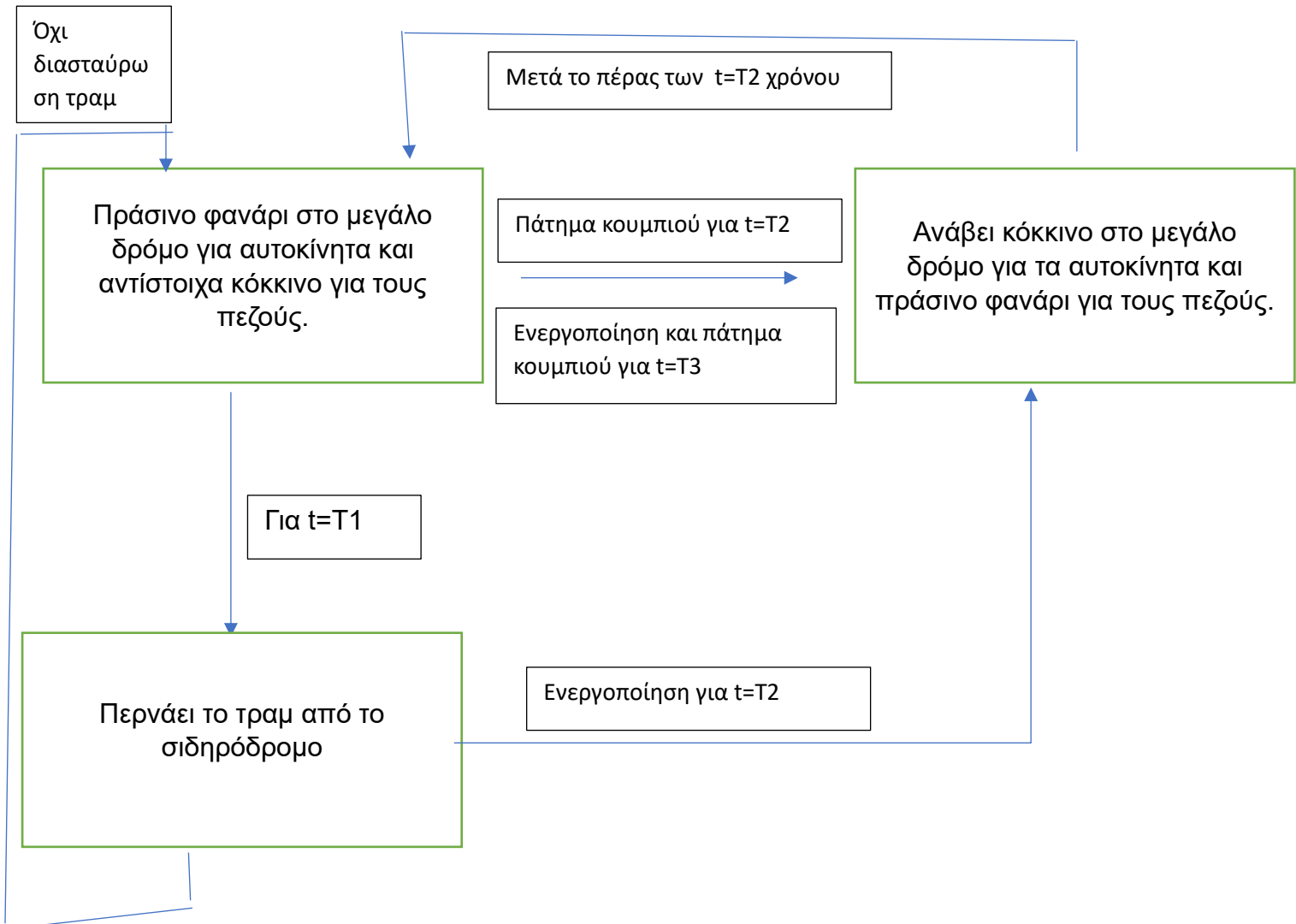
ΟΜΑΔΑ Β1 - ΕΡΓΑΣΤΗΡΙΑΚΗ ΑΣΚΗΣΗ 1

Στοιχεία Φοιτητών:

Αλευράς Ηλίας up1069667@upnet.gr 1069667

Σάββας Στυλιανού up1069661@upnet.gr 1069661

Διάγραμμα Ροής:



Κώδικας:

```
#include <avr/io.h>
#include <util/delay.h>
#include <avr/interrupt.h>
#define T1 30
#define T2 5
#define T3 10
int x=1;
int y=1;
int p=0;
int z=0;
int u=0;
int zero=0;
int main() {
    PORTD.DIR |= 0b0000001;
    PORTD.DIR |= 0b0000010;
    PORTD.DIR |= 0b0000100;
    PORTD.OUT |= 0b00000001; //LED is off
    PORTD.OUT |= 0b00000010; //LED is off
    PORTD.OUT |= 0b00000100; //LED is off
    PORTF.PIN5CTRL |= PORT_PULLUPEN_bm | PORT_ISC_BOTHEDGES_gc;
    //TIMER TRENOU
    /* Load the Compare or Capture register with the timeout value*/
    TCB0.CCMP = T1;
    /* Enable Capture or Timeout interrupt */
    TCB0.INTCTRL = TCB_CAPT_bm;

    TCB0.CTRLB = TCB_CNTMODE_INT_gc;

    //TIMER FANARI PEZON
    /* Load the Compare or Capture register with the timeout value*/
    TCB1.CCMP = T2;

    /* Configure TCB in Periodic Timeout mode */
    TCB1.CTRLB = TCB_CNTMODE_INT_gc;

    /* Enable Capture or Timeout interrupt */
    TCB1.INTCTRL = TCB_CAPT_bm;

    //timer gia koumpi t=T3
    /* Load the Compare or Capture register with the timeout value*/
    TCB2.CCMP = T3;

    /* Configure TCB in Periodic Timeout mode */
    TCB2.CTRLB = TCB_CNTMODE_INT_gc;

    /* Enable Capture or Timeout interrupt */
    TCB2.INTCTRL = TCB_CAPT_bm;

    aftokinita();

    sei();

    while(x==1){
        sei();
    }
}
```

```

        p=0;
        z=0;
        u=0;
        aftokinita();
        TCB0.CTRLA |= TCB_CLKSEL_CLKDIV1_gc ; //ksekina to treno
        TCB0.CTRLA |= TCB_ENABLE_bm;

        while(z==0){ //perimenoume to treno na ftasi
            ;
        }
        // to treno eftase
        pezo(); //pernoun oi pezo
        TCB1.CTRLA |= TCB_CLKSEL_CLKDIV1_gc ;
        TCB1.CTRLA |= TCB_ENABLE_bm;
        while(p==0){ //perimenoume to roloi ton pezon na teliosi
            ;
        }
        //to roloi ton pezon teliose
        aftokinita();
    }

}

void aftokinita(){//pernoun ta aftokinita
    PORTD.OUT |= 0b00000010; //pezo kokkino
    PORTD.OUTCLR= 0b00000100; //aftokinita prasino
}

void pezo(){//otan perna treno pernoun oi pezo
    PORTD.OUTCLR= 0b00000010; //pezo prasino
    PORTD.OUT |= 0b00000100; //aftokinita kokkino
}

ISR(TCB0_INT_vect){
    TCB0.INTFLAGS = TCB_CAPT_bm; /* Clear the interrupt flag */
    TCB0.CTRLA |= zero;
    z=1;
}

ISR(TCB1_INT_vect){
    TCB1.INTFLAGS = TCB_CAPT_bm; /* Clear the interrupt flag */
    TCB1.CTRLA |= zero;
    p=1;
}

ISR(TCB2_INT_vect){
    TCB2.INTFLAGS = TCB_CAPT_bm; /* Clear the interrupt flag */
    TCB2.CTRLA |= zero;
    u=1;
}

ISR(PORTF_PORT_vect){//koumpi pezon
    //clear the interrupt flag
    int z = PORTF.INTFLAGS;
    PORTF.INTFLAGS=z;
    TCB1.CTRLA |= TCB_CLKSEL_CLKDIV1_gc ;//energopiise timer gia fanari
    pezon
    TCB1.CTRLA |= TCB_ENABLE_bm;
    pezo();
    while(p==0){//perimene to timer
        ;
    }
    p=0;
}

```

```

    aftokinita();
    TCB2.CTRLA |= TCB_CLKSEL_CLKDIV1_gc ;//energopiise timer gia interrupts
    TCB2.CTRLA |= TCB_ENABLE_bm;
    cli();
    while(u==0){//perimene to timer
        ;
    }
    u=0;
    sei();
}

```

Περιγραφή:

Μας ζητήθηκε η επίλυση της πρώτης εργαστηριακής άσκησης η οποία αφορά την επίλυση ενός κυκλοφοριακού προβλήματος με 2 φανάρια για τα αυτοκίνητα και πεζούς στον μεγάλο δρόμο και για το σιδηρόδρομο.

Αρχικά, υλοποιήσαμε ένα διάγραμμα ροής το οποίο μας βοήθησε έτσι ώστε να έχουμε ένα πιο ξεκάθαρο πλάνο στη επίλυση της συγκεκριμένης άσκησης.

Όσο αφορά τον κώδικα, δηλώσαμε τις αρχικές καταστάσεις για τα ledακια για τα αυτοκίνητα , για τους πεζούς και για το σιδηρόδρομο, όπου:

- PIN0: τραμ.
- PIN1: διάβαση πεζών.
- PIN2: φώτα αυτοκινητόδρομου.
- PIN5 : διακοπή.

Υλοποιήσαμε το σύστημα με την δημιουργία των 3 ρολογιών TCB:

- TCB0: ρολόι τραμ.
- TCB1: ρολόι όπου το φανάρι για τη διάβαση πεζών είναι πράσινο.
- TCB2: ρολόι όπου είναι απενεργοποιημένα τα interrupts .

Φτιάξαμε μια συνάρτηση `autokinita()` η οποία ανάβει πράσινο το φανάρι των αυτοκινήτων και αυτόματα κόκκινο των πεζών. Επίσης γίνεται η αρχικοποίηση του ρολογιού του τραμ.

Το πρόγραμμα μας εισέρχεται σε ένα `infinite loop` όπου ενεργοποιεί τα `interrupt`. Επιπρόσθετα, ενεργοποιείται ένα ρολόι το οποίο αφορά τον χρόνο για να φτάσει το τραμ. Μόλις φτάσει το τραμ καλείται η συνάρτηση `pezoí()` όπου ανάβει πράσινο το φανάρι για τους πεζούς και κόκκινο για τα φώτα του αυτοκινητόδρομου.

Ξεκινά το ρολόι TCB1 το οποίο χρησιμοποιείται για να μας ενημερώνει το διαθέσιμο χρόνο όπου το φανάρι των πεζών παραμένει πράσινο. Όταν σβήσει το φανάρι των πεζών και γίνει κόκκινο ενεργοποιάται το ρολόι TCB2 όπου χρησιμοποιείται για τον χρόνο στον οποίο τα `interrupts` είναι off.

Τέλος, όταν πατηθεί το κουμπί από τους πεζούς καλείται η ISR του κουμπιού, η οποία με τη σειρά της καλεί την συνάρτηση `pezoí()` και το φανάρι των πεζών γίνεται πράσινο. Ξεκινά το ρολόι TCB1 το οποίο ενημερώνει το διαθέσιμο χρόνο όπου το φανάρι των πεζών παραμένει πράσινο.

Αν τα ρολόγια δεν σταματούν, να γίνει το bit του CTRLA=0 manually.