## 4<sup>η</sup> Ομάδα ασκήσεων- Εργαστηριακή Άσκηση

Ανθοπούλου Φαίδρα-Αναστασία 031118818

Αργυροπούλου Ζωή 03118904

## Ζήτημα 4.1

```
reset: ldi r24 , low(RAMEND) ; αρχικοποίηση stack pointer
       out SPL , r24
       ldi r24 , high(RAMEND)
       out SPH , r24
       ser r26; αρχικοποίηση της PORTA
       out DDRA , r26 ; για έξοδο
        clr r24 ; initialize PORTB
       out DDRB, r24; for input
main: clr r24; μετρητής για την θέση των led
       ldi r26,0x01 ;initialize r26
       rcall left
        nop
        rcall right
        rimp main
left:
         in r25, PINB; έλεγχος εισόδου
         andi r25, 0x01
            cpi r25, 0x01
            breq left
            out PORTA, r26
            inc r24
            1s1 r26
            cpi r24,7
            breq right
            rimp left
            ret
         in r25, PINB; έλεγχος εισόδου
right:
         andi r25, 0x01
            cpi r25, 0x01
            breq right
            out PORTA, r26
            dec r24
            1sr r26
            cpi r24,0
            breq left
```

```
rjmp right
             ret
Ζήτημα 4.2
#include <avr/io.h>
char A, B, C, D, F0, F1;
int main(void)
{
      DDRB=0xFF;
                                 //αρχικοποίηση του PORTB ως έξοδο
                                 //αρχικοποίηση του PORTA ως
      DDRA=0x00;
είσοδο
      while (1)
      {
            A=PINA & 0x01; //απομονώνουμε το A

B=PINA & 0x02; //απομονώνουμε το B

C=PINA & 0x04; //απομονώνουμε το C

D=PINA & 0x08; //απομονώνουμε το D
            D=PINA & 0x08; //απομονωνουμε το \nu R=B>>1; //αριστερή ολίσθηση των κουμπιών όσο
            C=C>>2; //χρειάζεται για να μετακινηθεί η αξία
                                   //τους στο αριστερότερο bit
            D=D>>3;
            F0= !((A & B & (!C)) | (C & D));//υπολογισμός του F0
            F1=((A \mid B) & (C \mid D)) <<1; //υπολογισμός του F1
και δεξιά ολίσθηση κατά μία θέση
            PORTB = (F0 \mid F1);
                                     //ένωση των F0 και F1 σε ένα
κοινό αποτέλεσμα που εμφανίζεται στο PORTB
      }
}
Ζήτημα 4.3
#include <avr/io.h>
char SW0, SW1, SW2, SW3, led;
int main(void){
      DDRA=0xFF;
                                //αρχικοποίηση του PORTA ως έξοδο
      DDRC=0x00;
                                //αρχικοποίηση του PORTC ως είσοδο
                                //αρχικοποίηση του led
      led=1;
                                //και προβολή του
      PORTA= led;
      while (1){
            SW0=PINC & 0x01;
                                               //απομονώνουμε το SW0
            SW1=PINC & 0x02;
                                               //απομονώνουμε το SW1
                                               //απομονώνουμε το SW2
            SW2=PINC & 0x04;
            SW3=PINC & 0x08;
                                               //απομονώνουμε το SW3
```

```
if (SW0 == 1){ //\epsilon \lambda \dot{\epsilon} \gamma \chi o u \mu \epsilon \alpha v \pi \alpha \tau \dot{\eta} \theta \eta \kappa \epsilon \tau o SW0}
                                                                                                                           while (SW0 == 1){ //\pi \epsilon \rho \iota \mu \dot{\epsilon} \nu \sigma u \alpha \phi \epsilon \theta \epsilon \dot{\iota} \tau \sigma
SW0
                                                                                                                                                                    SW0=PINC & 0x01;
                                                                                                                            if (led== 128){ //\alpha v \varepsilon i \mu \alpha \sigma \tau \varepsilon \sigma \tau \eta v \delta \varepsilon \xi i \delta \tau \varepsilon \rho \eta
θέση
                                                                                                                                                                    led=1; //πηγαίνουμε στην αριστερότερη
                                                                                                                            }
                                                                                                                            else{
                                                                                                                                                                    led =led << 1; //\alpha\lambda\lambda\iota\dot{\omega}\varsigma αριστερή ολίσθηση
                                                                                                                            }
                                                                                  if (SW1 == 2){ //\epsilon\lambda\epsilon\gamma\chiουμε αν πατήθηκε το SW1
                                                                                                                           while (SW1 == 2){ //\pi \epsilon \rho \iota \mu \dot{\epsilon} v \circ \iota \mu \dot{\epsilon} v \circ \iota \dot{\epsilon} v \circ \iota
SW1
                                                                                                                                                                    SW1=PINC & 0x02;
                                                                                                                            }
                                                                                                                            if (led== 1){ //αν είμαστε στην αριστερότερη
θέση
                                                                                                                                                                     led=128; //πηγαίνουμε στην δεξιότερη
                                                                                                                            }
                                                                                                                            else{
                                                                                                                                                                    led =led \Rightarrow 1; //\alpha\lambda\lambda\iota\dot{\omega}\varsigma \delta\epsilon\xi\iota\dot{\alpha} \delta\lambda\iota\sigma\theta\eta\sigma\eta
                                                                                                                            }
                                                                                  if (SW2 == 4){ //\epsilon\lambda\epsilon\gamma\chiουμε αν πατήθηκε το SW2
                                                                                                                           while (SW2 == 4){ //\pi\epsilon\rho\iota\mu\dot{\epsilon}\nu\sigma\nu\mu\epsilon \nu\alpha \alpha\phi\epsilon\theta\epsilon\dot{\epsilon} to
SW2
                                                                                                                                                                    SW2=PINC & 0x04;
                                                                                                                            led=128; //πηγαίνουμε στην αριστερότερη θέση
                                                                                  if (SW3 == 8){ //\epsilon\lambda\epsilon\gamma\chiουμε αν πατήθηκε το SW3
                                                                                                                           while (SW3 == 8){ //\pi \epsilon \rho \iota \mu \epsilon v \circ \nu \alpha \alpha \phi \epsilon \theta \epsilon \iota
το SW3
                                                                                                                                                                    SW3=PINC & 0x08;
                                                                                                                            led=1; //πηγαίνουμε στην δεξιότερη θέση
                                                                                  PORTA= led; //το led παίρνει τη θέση που έχει
καθοριστεί από τους διακόπτες
                                                                                  }
                                         }
```

Αρκετά σχόλια κατέληξαν να καταλαμβάνουν 2 σειρές εξαιτίας της μεταφοράς τους (η σελίδα του word δεν είναι αρκετά πλατιά). Στον κώδικα καταλαμβάνουν μία μόνο σειρά, αλλιώς θα υπήρχε και ένα ακόμη // στην δεύτερη σειρά που καταλαμβάνουν.