

4^η 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
      rjmp main
```

```
left:   in r25, PINB; έλεγχος εισόδου
      andi r25, 0x01
      cpi r25, 0x01
      breq left
      out PORTA, r26
      inc r24
      lsl r26
      cpi r24,7
      breq right
      rjmp left

      ret
```

```
right:  in r25, PINB; έλεγχος εισόδου
      andi r25, 0x01
      cpi r25, 0x01
      breq right
      out PORTA, r26
      dec r24
      lsr 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
    DDRA=0x00;           //αρχικοποίηση του PORTA ως
    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
        B=B>>1;           //αριστερή ολίσθηση των κουμπιών όσο
        C=C>>2;           //χρειάζεται για να μετακινηθεί η αξία
        D=D>>3;           //τους στο αριστερότερο bit
        F0= !((A & B & (!C)) | (C & D)); //υπολογισμός του F0
        F1= ((A | B) & (C | D)) <<1;    //υπολογισμός του F1
        και δεξιά ολίσθηση κατά μία θέση
        PORTB = (F0 | 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
    led=1;               //αρχικοποίηση του led
    PORTA= led;          //και προβολή του
    while (1){
        SW0=PINC & 0x01;    //απομονώνουμε το SW0
        SW1=PINC & 0x02;    //απομονώνουμε το SW1
        SW2=PINC & 0x04;    //απομονώνουμε το SW2
        SW3=PINC & 0x08;    //απομονώνουμε το SW3
    }
}
```


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.