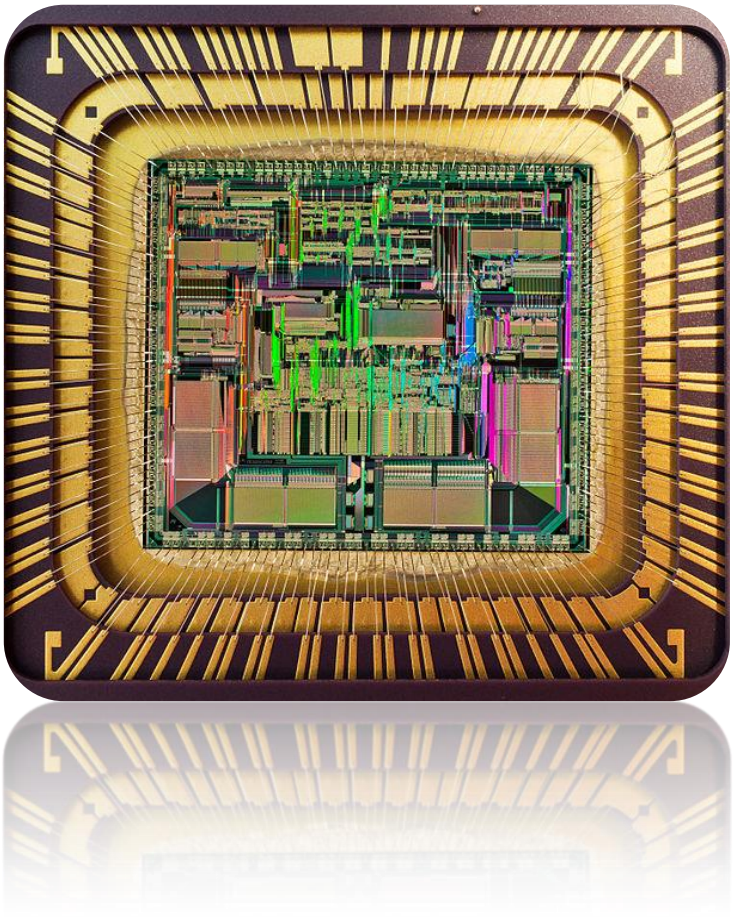
ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝΤΟΜΕΑΣ ΤΕΧΝΟΛΟΓΙΑΣ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΥΠΟΛΟΓΙΣΤΩΝΣΥΣΤΗΜΑΤΑ ΜΙΚΡΟΥΠΟΛΟΓΙΣΤΩΝ

6η Ομάδα Ασκήσεων

Βαβουλιώτης Γεώργιος
ΑΜ : 03112083
6ο Εξάμηνο

Γιαννόπουλος Αναστάσης
Α.Μ. : 03112176
6ο Εξάμηνο



ΑΣΚΗΣΗ 1

Ο κώδικας σε assembly AVR για την πρώτη άσκηση φαίνεται παρακάτω :

```
.INCLUDE "m16def.inc"  ;dhlwsh mikroelegkth
.def temp=r16
.def reg=r17
.def Delay=r18          ;kataxwrhths metablhths ka8ysterhshs

      clr temp
      out DDRD,temp    ;PORTD 8yra eisodou
      ser temp
      out DDRB,temp    ;PORTB 8yra eksodou
      out PORTD,temp


CHECK_POINT:
      in temp,PIND
      andi temp,0x80   ;MSB
      cpi temp,0       ;an to MSB einai 0
      breq LABEL0

LABEL1:
      ldi reg,0xFF     ;anavw ta led gia 0.5 sec giati MSB=1
      out PORTB,reg
      rcall DELAY1
      ldi reg,0x00     ;svhnw ta led gia 1.5 sec giati MSB=1
      out PORTB,reg
      rcall DELAY2
      rjmp CHECK_POINT

LABEL0:
      ldi reg,0xFF     ;anavw ta led gia 1.5 sec giati MSB=0
      out PORTB,reg
      rcall DELAY2
      ldi reg,0x00     ;svhnw ta led gia 0.5 sec giati MSB=0
      out PORTB,reg
      rcall DELAY1
      rjmp CHECK_POINT

;0.5sec=500msec ==> Delay10 50 fores
DELAY1:
      ldi Delay,50
      CHECKDELAY1:
            rcall Delay10
            dec Delay
            brne CHECKDELAY1
      ret

;1.5sec=1500msec ==> Delay10 150 fores
DELAY2:
      ldi Delay,150
      CHECKDELAY2:
            rcall Delay10
            dec Delay
            brne CHECKDELAY2
      ret
```

ΑΣΚΗΣΗ 2

Ο κώδικας σε assembly AVR για την δεύτερη άσκηση φαίνεται παρακάτω :

```
.INCLUDE "m16def.inc" ;dhlwsh mikroelegkth
.def A=r16            ;dhlwsh kataxwrhtwn
.def B=r17            ;kai antistoixhsh me  metavlhtes
.def C=r18
.def D=r19
.def E=r20
.def F=r21
.def temp=r22
.def tempA=r23
.cseg
.org0 ;diey8ynsh ekkinhshs


      clr temp
      out DDRC,temp    ;C 8yra eisodou
      ser temp
      out PORTC,temp
      out DDRA,temp    ;A 8yra eksodou
loop_exer2:
      in temp,PINC     ;diavazw thn eisodo
      lsr temp
      lsr temp
      mov A,temp        ;to bit2 ths eisodou ston A
      move tempA,temp
      lsr temp
      mov B,temp       ;to bit3 ths eisodou ston B
      lsr temp
      mov C,temp       ;to bit4 ths eisodou ston C
      lsr temp
      mov D,temp       ;to bit5 ths eisodou ston D
```

```
        lsr temp
        mov E,temp      ;to bit6 ths eisodou ston E
        lsr temp
        mov F,temp      ;to bit 7 ths eisodou ston F

myfun21:
        and A,B
        mov temp,C
        com D           ;D'
        com E           ;E'
        and temp,D
        and temp,E      ;C*D'*E'
        or temp,A       ;A*B + C*D'*E'
        andi temp,0x01  ;to LSB
        out PORTA,temp  ;outpout sto 1o bit ths 8yras A

        ;epanafora twn A,D,E
        com D
        com E
        mov A,tempA
myfun22:
        com C
        and A,B
        and A,C
        and A,D
        com D
        com F
        and D,E
        and D,F
        or A,D          ;sto LSB tou A to apotelesma ths X1
        mov tempA,A     ;to apotelesma ths X1 ston tempA
        andi A,0x01     ;krataw to LSB
```

```
        lsl A
        out PORTA,A      ;outpout sto 2o bit ths 8yras A

myfun23:
        or temp,tempA
        lsl temp
        lsl temp
        out PORTA,temp   ;outpout sto 2o bit ths 8yras A
        rjmp loop_exer2  ;synexomenh leitourgia
```

ΑΣΚΗΣΗ 3

Ο κώδικας σε assembly AVR για την τρίτη άσκηση φαίνεται παρακάτω :

```
.INCLUDE "m16def.inc"   ;dhlwsh mikroelegkth
.def reg=r16            ;orismos kataxwrhth
.def temp=r17
.def reg1=r18
allstarts:
      clr temp
      out DDRD,temp    ;port D 8yra eisodou
      set temp
      out DDRB,temp    ;port B 8yra eksodou
      out PORTD,temp
      ldi temp,0x01
      out PORTB,temp   ;anavw to LSB led

check_point:
      in reg,PIND      ;anagnwsh ths eisodou
      mov reg1,reg
      andi reg,0x0F    ;00001111 an ola pathmena
      cpi reg,0x0F     ;an ola pathmena ==> arxikh katastash
      breq keepsame

      mov reg,reg1     ;epanaferw th swsth timh
      andi reg,0x03
      cpi reg,0x03     ;an einai pathmena ta sw0 sw1
      breq goleft

      mov reg,reg1     ;epanaferw th swsth timh
      andi reg,0x05
      cpi reg,0x05     ;an einai pathmena ta sw0 sw2
      breq goright
```

```
	mov reg,reg1	;epanaferw th swsth timh
	andi reg,0x06
	cpi reg,0x06	;an einai pathmena ta sw1 sw2
	breq goleft2times

	mov reg,reg1	;epanaferw th swsth timh
	andi reg,0x09
	cpi reg,0x09	;an einai pathmena ta sw0 sw3
	breq goright2times

	mov reg,reg1	;epanaferw th swsth timh
	andi reg,0x0A
	cpi reg,0x0A	;an einai pathmena ta sw1 sw3
	breq turn0_1

	mov reg,reg1	;epanaferw th swsth timh
	andi reg,0x0C
	cpi reg,0x0C	;an einai pathmena ta sw2 sw3
	breq keepsame

keepsame:
	ldi temp,0x01	;temp=00000001
	out PORTB,temp	;anavw to LSB led
	rjmp check_point

goleft:
	rol temp
	out PORTB,temp
	rjmp check_point
```

```
goright:
      ror temp
      out PORTB,temp
      rjmp check_point

goleft2times:
      rol temp
      rol temp
      out PORTB,temp
      rjmp check_point

goright2times:
      ror temp
      ror temp
      out PORTB,temp
      rjmp check_point

turn0_1:
      ldi temp,0x03    ;temp=00000011
      out PORTB,temp   ;anavw ta led 0 kai 1
      rjmp check_point
```