



# ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

ΣΧΟΛΗ ΗΜ&ΜΥ  
Εργαστήριο Μικροϋπολογιστών

7<sup>η</sup> Εργαστηριακή Άσκηση  
Ακ. έτος 2011-2012

*Ομάδα C07:*

Ελένη Ευαγγελάτου	A.M.: 03108050
Γρηγόρης Λύρας	A.M.: 03109687
Βασιλεία Φραγκιαδάκη	A.M.: 03108026

14 Φεβρουαρίου 2012

## Άσκηση (i)

Κυρίως κώδικας:

```
1  /*Includes for compatibility with GNU toolchain*/
2  #define __SFR_OFFSET 0
3  #include <avr/io.h>
4  #include <avr/interrupt.h>
5  .global main
6  /*
7   * AVR_3h_ask1.asm
8   *
9   * Created: 10/2/2012 10:01:52 ??
10  * Author: Valia
11  */
12
13
14  /*
15  * AVR_ask1.asm
16  *
17  * Created: 10/2/2012 9:24:48 ??
18  * Author: Valia
19  */
20
21  ;//.def r24 = r24
22  ;//.def r28 = r28
23  ;//.def r27 = r27
24  ;//.def r25 = r25
25  ;//.def r29 = r29
26  ;//.def r26 = r26
27  ;//.def r30 = r30
28
29  main:
30      ldi r24, hi8(RAMEND)
31      out SPH, r24
32      ldi r24, lo8(RAMEND)
33      out SPL, r24
34
35      ser r24
36      out DDRB, r24
37
38      clr r24
39      out DDRA, r24
40
41      clr r29
42      out DDRC, r29
43  start:
44      clr r30
45      in r29, PINC
46      lsr r29
47      brcc syn1
48  etik1:    ;//perimenei mexri na ginei 0 (button)
49      in r26, PINC
50      lsr r26
51      brcs etik1
52      ori r30, 1
53
54  syn1:
55      lsr r29
56      brcc syn2
57  etik2:    ;//perimenei mexri na ginei 0 (button)
58      in r26, PINC
59      lsr r26
60      lsr r26
61      brcs etik2
62      ori r30, 2
63
64  syn2:
65      lsr r29
66      brcc syn3
67  etik3:    ;//perimenei mexri na ginei 0 (button)
68      in r26, PINC
69      lsr r26
70      lsr r26
71      lsr r26
72      brcs etik3
```

```

73     ori r30, 4
74
75     syn3:
76         lsr r29
77         brcc syn4
78     etik4:        ;//perimenei meaxri na ginei 0 (button)
79         in r26, PINC
80         lsr r26
81         lsr r26
82         lsr r26
83         lsr r26
84         brcs etik4
85         ori r30, 8
86
87     syn4:
88         lsr r29
89         brcc syn5
90     etik5:        ;//perimenei meaxri na ginei 0 (button)
91         in r26, PINC
92         lsl r26
93         lsl r26
94         lsl r26
95         lsl r26
96         brcs etik5
97         ori r30, 0x010
98
99     syn5:
100         lsr r29
101         brcc syn6
102     etik6:        ;//perimenei meaxri na ginei 0 (button)
103         in r26, PINC
104         lsl r26
105         lsl r26
106         lsl r26
107         brcs etik6
108         ori r30, 0x020
109
110     syn6:
111         lsr r29
112         brcc syn7
113     etik7:        ;//perimenei meaxri na ginei 0 (button)
114         in r26, PINC
115         lsl r26
116         lsl r26
117         brcs etik7
118         ori r30, 0x040
119
120     syn7:
121         lsr r29
122         brcc synexise
123     etik8:        ;//perimenei meaxri na ginei 0 (button)
124         in r26, PINC
125         lsl r26
126         brcs etik8
127         ori r30, 0x080
128
129     synexise:
130
131         in r28, PINA
132         clr r25    ;//mhdenismos exodun
133         clr r27
134
135     gate_1:
136         lsr r28
137         brcc exit_gate1 ;//an to lsb == 0 feugoume
138         lsr r28
139         brcc exit_gate1 ;//an to 2o lsb == 0 feugoume
140         inc r25
141
142     exit_gate1:
143
144     gate_2:
145         lsr r28
146         brcc exit_gate2 ;//an to 3o lsb == 0 feugoume
147         lsr r28

```

```

148     brcc exit_gate2 ;//an to 4o lsb == 0 feugoume
149     ori r27, 2
150
151 exit_gate2:
152     or r27, r25 ;//exodos sta 2 prwta lsb bit
153
154 gate_3:
155     lsr r28
156     brcs exit_gate3 ;//an to 5o lsb == 1 feugoume
157     lsr r28
158     brcs exit_gate3 ;//an to 6o lsb == 1 feugoume
159     ori r27, 4 ;// or me 100
160
161 exit_gate3:
162
163 gate_4:
164     lsr r28
165     brcs one ;//an einai 1 psaxnoumt an kai to allo einai 0
166     lsr r28 ;//an einai to 1o 0, tsekaroyme an kai to 2o einai 1
167     brcc exit_gate4
168     rjmp setting
169
170 one:
171     lsr r28
172     brcs exit_gate4 ;// an to allo einai 0 feugoyme
173
174 setting:
175     ori r27, 8
176
177 exit_gate4:
178     eor r27, r30 ;//xor
179     out PORTB, r27
180     rjmp start

```

## Άσκηση (ii)

Κυρίως κώδικας:

```
1  /* .....  
2  
3  * File Name : part2.c  
4  
5  * Purpose :  
6  
7  * Creation Date : 07-02-2012  
8  
9  * Last Modified : Wed 08 Feb 2012 12:09:50 AM EET  
10  
11 * Created By : Greg Liras <gregliras@gmail.com>  
12  
13 .....*/  
14  
15  
16 #include<avr/io.h>  
17  
18 int f0 ( int c )  
19 {  
20     for( ; c > 0 ; c >=1 )  
21         if ( ( c & 3 ) == 3 )  
22             return 0;  
23     return 1;  
24 }  
25 int f1 ( int c )  
26 {  
27     if ( c <= 15 || c == 31 )  
28         return 1;  
29     return 0;  
30 }  
31 int main(void)  
32 {  
33     DDRA = 0xff;  
34     DDRC = 0x00;  
35     int f_0;  
36     int f_1;  
37     int c;  
38     for( ;; )  
39     {  
40         c = PINC & 31;  
41         f_0 = f0( c );  
42         f_1 = f1( c );  
43         PORTA = f_0 | ( f_1 << 1 ) | ( ( f_0 | f_1 ) << 2 );  
44     }  
45 }
```

### Άσκηση (iii)



Σχήμα 1: Διάγραμμα ροής

Κυρίως κώδικας:

```
1  #define __SFR_OFFSET 0
2  #include <avr/io.h>
3
4  .data
5  _tmp_: .byte 2
6  .text
7  .global main
8  main:
9  reset:
10     ldi r24,lo8(RAMEND)
11     out SPL,r24
12     ldi r24,hi8(RAMEND)
13     out SPH,r24
14     ser r24
15     out DDRA,r24
16     ;// PORTA output
17     ldi r24,(1 << PC7) | (1 << PC6) | (1 << PC5) | (1 << PC4)
18     out DDRC,r24
19     ;// 4x4 pad input
20     rcall scan_keypad_rising_edge
21     ;// initialize
22 lock:
23     ;//reset the memory
24     ldi r24,0x00
25     ldi r25,0x00
26     ldi r26,lo8(_tmp_)
27     ldi r27,hi8(_tmp_)
28     st X+,r24
```

```

29     st X ,r25
30     ldi r24, 0x14
31     rcall scan_keypad_rising_edge
32     cpi r24, 0x00
33     ;//first test failed
34     brne lock
35     ldi r24, 0x14
36     rcall scan_keypad_rising_edge
37     ;//second test failed
38     cpi r24, 0x12
39     brne lock
40     rjmp pass
41
42     pass:
43         ldi r24,0xff
44         out PORTA, r24
45         ldi r24,lo8(2000)
46         ldi r25,hi8(2000)
47         rcall wait_msec
48         ;//reset the leds
49         ldi r24,0x00
50         out PORTA, r24
51         rjmp lock
52
53
54     scan_row:
55         ldi r25 ,0x08
56     back_: lsl r25
57         dec r24
58         brne back_
59         out PORTC ,r25
60         nop
61         nop
62         in r24 ,PINC
63         andi r24 ,0x0f
64         ret
65
66
67
68     scan_keypad:
69         ldi r24 ,0x01
70         rcall scan_row
71         swap r24
72         mov r27 ,r24
73         ldi r24 ,0x02
74         rcall scan_row
75         add r27 ,r24
76         ldi r24 ,0x03
77         rcall scan_row
78         swap r24
79         mov r26 ,r24
80         ldi r24 ,0x04
81         rcall scan_row
82         add r26 ,r24
83         movw r24 ,r26
84         ret
85
86
87     scan_keypad_rising_edge:
88         mov r22 ,r24
89         rcall scan_keypad
90         push r24
91         push r25
92         mov r24 ,r22
93         ldi r25 ,0
94         rcall wait_msec
95         rcall scan_keypad
96         pop r23
97         pop r22
98         and r24 ,r22
99         and r25 ,r23
100        ldi r26 ,lo8(_tmp_)
101        ldi r27 ,hi8(_tmp_)
102        ld r23 ,X+
103        ld r22 ,X

```

```

104     st X ,r24
105     st -X ,r25
106     com r23
107     com r22
108     and r24 ,r22
109     and r25 ,r23
110     ret
111
112
113
114     wait_usec:
115         sbiw r24,1
116         nop
117         brne wait_usec
118         ret
119
120     wait_msec:
121         in r27,PINB
122         ror r27
123         brcs wait_msec
124         push r24
125         push r25
126         ldi r24,lo8(998)
127         ldi r25,hi8(998)
128         rcall wait_usec
129         pop r25
130         pop r24
131         sbiw r24,1
132         brne wait_msec
133         ret

```



## Άσκηση (iv)

Κυρίως κώδικας:

```
1  /*Includes for compatibility with GNU toolchain*/
2  #define __SFR_OFFSET 0
3  #include <avr/io.h>
4  #include <avr/interrupt.h>
5  .global main
6  /*
7   * AVRAsk4.asm
8   *
9   * Created: 12/2/2012 10:31:36 ??
10  * Author: Valia
11  */
12  .data
13  _tmp_ :.byte 2
14
15  .text
16  #define temp1 r24
17  #define temp2 r25
18
19  main:
20      rcall lcd_init ;//initialize
21
22      ldi r24,0x045 ;// 'T'
23      rcall lcd_data
24      ldi r24,0x05B ;// 'E'
25      rcall lcd_data
26      ldi r24,0x014 ;// 'A'
27      rcall lcd_data
28      ldi r24,0x0b3 ;// 'M'
29      rcall lcd_data
30      ldi r24,0x00c ;// ' '
31      rcall lcd_data
32      ldi r24,0x07c ;// 'O'
33      rcall lcd_data
34      ldi r24,0x073 ;// '7'
35      rcall lcd_data
36
37  start:
38      rcall scan_keypad_rising_edge ;//apotelesma sto r24:r25
39      rcall keypad_to_ascii
40      cpi r24,0
41      breq start
42
43      push r24
44      ldi r24,0x01 ;//freskarisma
45      rcall lcd_command
46      ldi r24,lo8(1530)
47      ldi r25,hi8(1530)
48      rcall wait_usec
49      pop r24
50
51      rcall lcd_data ;//emfanish sthn o8onh
52
53      ldi r26,lo8(_tmp_) ;//ka8arisma mmhmhs
54      ldi r27,hi8(_tmp_)
55      ldi r24,0x0
56      st X+,r24
57      st X,r24
58
59      rjmp start
60
61  keypad_to_ascii:
62      movw r26,r24
63      ldi r24,'*'
64      sbrc r26,0
65      ret
66      ldi r24,'O'
67      sbrc r26,1
68      ret
69      ldi r24,'#'
70      sbrc r26,2
71      ret
72      ldi r24,'D'
```

```

73     sbrlc r26,3
74     ret
75     ldi r24,'7'
76     sbrlc r26,4
77     ret
78     ldi r24,'8'
79     sbrlc r26,5
80     ret
81     ldi r24,'9'
82     sbrlc r26,6
83     ret
84     ldi r24,'C'
85     sbrlc r26,7
86     ret
87     ldi r24,'4'
88     sbrlc r27,0
89     ret
90     ldi r24,'5'
91     sbrlc r27,1
92     ret
93     ldi r24,'6'
94     sbrlc r27,2
95     ret
96     ldi r24,'B'
97     sbrlc r27,3
98     ret
99     ldi r24,'1'
100    sbrlc r27,4
101    ret
102    ldi r24,'2'
103    sbrlc r27,5
104    ret
105    ldi r24,'3'
106    sbrlc r27,6
107    ret
108    ldi r24,'A'
109    sbrlc r27,7
110    ret
111    clr r24
112    ret
113
114    scan_keypad_rising_edge:
115    mov r22,r24
116    rcall scan_keypad
117
118    push r24 ;//apo8ikeush apotelesmatos
119    push r25
120    mov r24,r22 ;//ka8ysterhsh r22msec (10-20 msec)
121    ldi r25,0
122    rcall wait_msec
123
124    rcall scan_keypad ;//aporrispe osa plhktra emfanizoyrn spin8hrismo
125    pop r23
126    pop r22
127    and r24,r22
128    and r25,r23
129    ldi r26,lo8(_tmp_)
130    ldi r27,hi8(_tmp_)
131    ld r23,X+
132    ld r22,X
133    st X,r24
134    st -X,r25
135    com r23
136    com r22
137    and r24,r22
138    and r25,r22
139    ret
140
141    lcd_init:
142    ldi r24,40
143    ldi r25,0
144    ;//rcall wait_msec
145
146    ldi r24,0x30
147    out PORTD,r24

```

```

148     sbi PORTD,PD3
149     cbi PORTD,PD3
150     ldi r24,39
151     ldi r25,0
152     ;//rcall wait_usec
153
154     ldi r24,0x30
155     out PORTD,r24
156     sbi PORTD,PD3
157     cbi PORTD,PD3
158     ldi r24,39
159     ldi r25,0
160     ;//rcall wait_usec
161
162     ldi r24,0x20
163     out PORTD,r24
164     sbi PORTD,PD3
165     cbi PORTD,PD3
166     ldi r24,39
167     ldi r25,0
168     ;//rcall wait_usec
169
170     ldi r24,0x28
171     rcall lcd_command
172
173     ldi r24,0x0c
174     rcall lcd_command
175
176     ldi r24,0x01
177     rcall lcd_command
178     ldi r24,lo8(1530)
179     ldi r25,hi8(1530)
180     ;//rcall wait_usec
181
182     ldi r24,0x06
183     rcall lcd_command
184
185     ret
186
187     lcd_data:
188     sbi PORTD,PD2
189     rcall write_2_nibbles
190     ldi r24,43
191     ldi r25,0
192     ;//rcall wait_usec
193     ret
194
195     lcd_command:
196     cbi PORTD,PD2
197     rcall write_2_nibbles
198     ldi r24,39
199     ldi r25,0
200     ;//rcall wait_usec
201     ret
202
203     write_2_nibbles:
204     push r24
205     in r25,PIND
206     andi r25,0x0f
207     andi r24,0xf0
208     add r24,r25
209     out PORTD,r24
210     sbi PORTD,PD3
211     cbi PORTD,PD3
212     pop r24
213     swap r24
214     andi r24,0xf0
215     add r24,r25
216     out PORTD,r24
217     sbi PORTD,PD3
218     cbi PORTD,PD3
219     ret
220
221     scan_keypad:
222     ldi r24,0x01

```

```

223     rcall scan_row
224     swap r24
225     mov r27,r24
226     ldi r24,0x02
227     rcall scan_row
228     add r27,r24      ;//1h kai 2h grammh
229     ldi r24,0x03
230     rcall scan_row
231     swap r24
232     mov r26,r24
233     ldi r24,0x04
234     rcall scan_row
235     add r26,r24      ;//3h kai 4h grammh
236     movw r24,r26     ;//r25:r24 to apotelesma
237     ret
238
239     scan_row:
240     ldi r25,0x08
241     back_:
242     lsl r25
243     dec r24
244     brne back_
245     out PORTC,r25
246     nop
247     nop
248     in r24,PINC
249     andi r24,0x0f
250     ret
251
252     wait_msec:
253     push temp1
254     push temp2
255     ldi temp1, lo8(998)
256     ldi temp2, hi8(998)
257     rcall wait_usec
258     pop temp2
259     pop temp1
260     sbiw temp1, 1
261     brne wait_msec
262
263     ret
264
265     wait_usec:
266     sbiw temp1,1
267     nop
268     nop
269     nop
270     nop
271     brne wait_usec
272
273     ret

```

## Άσκηση (v)

Κυρίως κώδικας:

```
1  #define __SFR_OFFSET 0
2  #include <avr/io.h>
3
4  .global main
5  main:
6  reset:
7      ldi r24,lo8(RAMEND)
8      out SPL,r24
9      ldi r24,hi8(RAMEND)
10     out SPH,r24
11     ser r24
12     out DDRA,r24
13     clr r24
14     out DDRB,r24
15
16  loop:
17     rjmp loop
18
19
20
21  alarm:
22     push r24
23     ldi r24,0xff
24     out PORTA,r24
25     pop r24
26     ret
27
28  set_timer:
29     push r24
30     ldi r24,hi8(2048)
31     out
```

## Άσκηση (vi)

Κυρίως κώδικας:

1