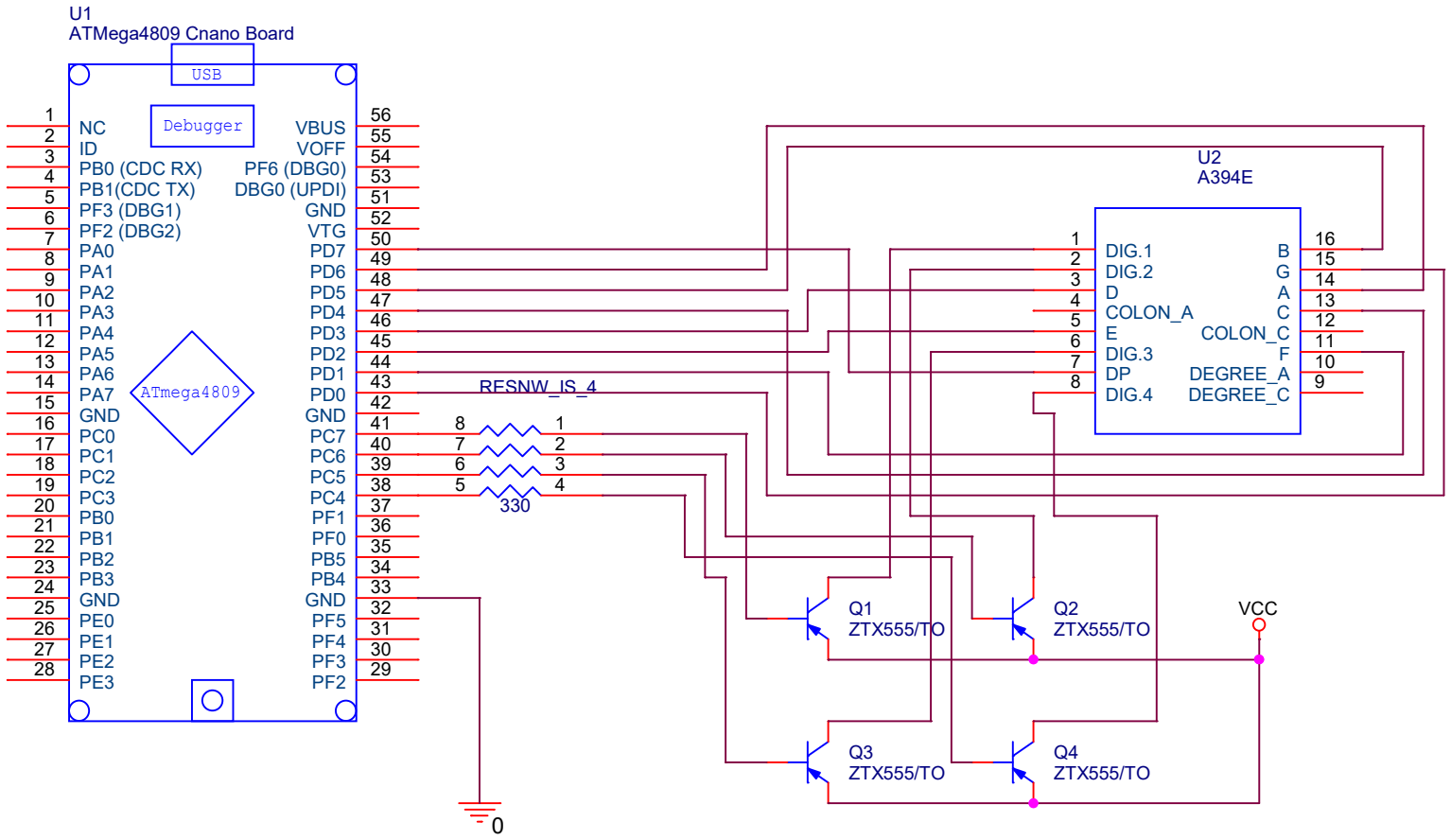
Judah Ben-Eliezer

112352727
10/20/2020

Prelab 7:

Hardware Switch Debouncing and Conditional I/O



Title	Segment and Digit Test	
Size A	Document Number <Doc>	Rev <RevCo
Date: Tuesday, October 20, 2020	Sheet 1 of 1	

```
 1
 2  AVRASM ver. 2.2.7  E:\ESE_280\$MyDocuments$\Atmel Studio\7.0\lab_7\lab_7\lab_7
      \main.asm Tue Oct 20 19:46:25 2020
 3
 4  E:\ESE_280\$MyDocuments$\Atmel Studio\7.0\lab_7\lab_7\lab_7\main.asm(9):
      Including file 'C:/Program Files (x86)\Atmel\Studio\7.0\Packs\atmel
      \ATmega_DFP\1.3.300\avrasm\inc\m4809def.inc'
 5  E:\ESE_280\$MyDocuments$\Atmel Studio\7.0\lab_7\lab_7\lab_7\main.asm(9):
      Including file 'C:/Program Files (x86)\Atmel\Studio\7.0\Packs\atmel
      \ATmega_DFP\1.3.300\avrasm\inc\m4809def.inc'
 6
 7
 8                                  ; segment_and_digit_test.asm
 9                                  ;
10                                  ; Created: 10/20/2020 6:49:26 PM
11                                  ; Author : Judah Ben-Eliezer
12                                  ;
13
14                                  .list
15
16                                  #define F_CPU 8000000UL
17
18                                  start:
19  000000 ef0f                         ldi r16, $FF
20  000001 b90c                         out VPORTD_DIR, r16
21  000002 b908                         out VPORTC_DIR, r16
22  000003 e000                         ldi r16, $00
23  000004 e810                         ldi r17, $80
24  000005 e220                         ldi r18, $20
25
26                                  main_loop:
27  000006 fd13                         sbrc r17, 3     ; sets mask to first bit 7
      only if bit 3 is set
28  000007 e810                         ldi r17, $80
29  000008 b919                         out VPORTC_OUT, r17
30  000009 b90d                         out VPORTD_OUT, r16
31
32                                  delay_1_s:      ; not sure yet if 1 s
33                                  outer_loop:
34  00000a ef3a                         ldi r19, $FA
35                                  inner_loop:
36  00000b 953a                         dec r19
37  00000c f7f1                         brne inner_loop
38  00000d 952a                         dec r18
39  00000e f7d9                         brne outer_loop
40
41                                  ror_r17:
42  00000f 9517                         ror r17
43  000010 cff5                         rjmp main_loop
```

```
44
45
46
47
48  RESOURCE USE INFORMATION
49  ------------------------
50
51  Notice:
52  The register and instruction counts are symbol table hit counts,
53  and hence implicitly used resources are not counted, eg, the
54  'lpm' instruction without operands implicitly uses r0 and z,
55  none of which are counted.
56
57  x,y,z are separate entities in the symbol table and are
58  counted separately from r26..r31 here.
59
60  .dseg memory usage only counts static data declared with .byte
61
62  "ATmega4809" register use summary:
63  x  :   0 y  :   0 z  :   0 r0 :   0 r1 :   0 r2 :   0 r3 :   0 r4 :   0
64  r5 :   0 r6 :   0 r7 :   0 r8 :   0 r9 :   0 r10:   0 r11:   0 r12:   0
65  r13:   0 r14:   0 r15:   0 r16:   5 r17:   5 r18:   2 r19:   2 r20:   0
66  r21:   0 r22:   0 r23:   0 r24:   0 r25:   0 r26:   0 r27:   0 r28:   0
67  r29:   0 r30:   0 r31:   0
68  Registers used: 4 out of 35 (11.4%)
69
70  "ATmega4809" instruction use summary:
71  .lds  :   0 .sts  :   0 adc   :   0 add   :   0 adiw  :   0 and   :   0
72  andi  :   0 asr   :   0 bclr  :   0 bld   :   0 brbc  :   0 brbs  :   0
73  brcc  :   0 brcs  :   0 break :   0 breq  :   0 brge  :   0 brhc  :   0
74  brhs  :   0 brid  :   0 brie  :   0 brlo  :   0 brlt  :   0 brmi  :   0
75  brne  :   2 brpl  :   0 brsh  :   0 brtc  :   0 brts  :   0 brvc  :   0
76  brvs  :   0 bset  :   0 bst   :   0 call  :   0 cbi   :   0 cbr   :   0
77  clc   :   0 clh   :   0 cli   :   0 cln   :   0 clr   :   0 cls   :   0
78  clt   :   0 clv   :   0 clz   :   0 com   :   0 cp    :   0 cpc   :   0
79  cpi   :   0 cpse  :   0 dec   :   2 des   :   0 eor   :   0 fmul  :   0
80  fmuls :   0 fmulsu:   0 icall :   0 ijmp  :   0 in    :   0 inc   :   0
81  jmp   :   0 ld    :   0 ldd   :   0 ldi   :   6 lds   :   0 lpm   :   0
82  lsl   :   0 lsr   :   0 mov   :   0 movw  :   0 mul   :   0 muls  :   0
83  mulsu :   0 neg   :   0 nop   :   0 or    :   0 ori   :   0 out   :   4
84  pop   :   0 push  :   0 rcall :   0 ret   :   0 reti  :   0 rjmp  :   1
85  rol   :   0 ror   :   1 sbc   :   0 sbci  :   0 sbi   :   0 sbic  :   0
86  sbis  :   0 sbiw  :   0 sbr   :   0 sbrc  :   1 sbrs  :   0 sec   :   0
87  seh   :   0 sei   :   0 sen   :   0 ser   :   0 ses   :   0 set   :   0
88  sev   :   0 sez   :   0 sleep :   0 spm   :   0 st    :   0 std   :   0
89  sts   :   0 sub   :   0 subi  :   0 swap  :   0 tst   :   0 wdr   :   0
90
91  Instructions used: 7 out of 114 (6.1%)
92
```

```
 93  "ATmega4809" memory use summary [bytes]:
 94  Segment   Begin    End      Code   Data   Used    Size   Use%
 95  ----------------------------------------------------------------
 96  [.cseg] 0x000000 0x000022     34      0     34   49152   0.1%
 97  [.dseg] 0x002800 0x002800      0      0      0    6144   0.0%
 98  [.eseg] 0x000000 0x000000      0      0      0     256   0.0%
 99
100  Assembly complete, 0 errors, 0 warnings
101
```

```
 1
 2  AVRASM ver. 2.2.7  E:\ESE_280\$MyDocuments$\Atmel Studio\7.0\lab_7
      \table_lookup_seg_check\table_lookup_seg_check\main.asm Tue Oct 20 20:27:01
      2020
 3
 4  E:\ESE_280\$MyDocuments$\Atmel Studio\7.0\lab_7\table_lookup_seg_check
      \table_lookup_seg_check\main.asm(9): Including file 'C:/Program Files (x86)
      \Atmel\Studio\7.0\Packs\atmel\ATmega_DFP\1.3.300\avrasm\inc\m4809def.inc'
 5  E:\ESE_280\$MyDocuments$\Atmel Studio\7.0\lab_7\table_lookup_seg_check
      \table_lookup_seg_check\main.asm(9): Including file 'C:/Program Files (x86)
      \Atmel\Studio\7.0\Packs\atmel\ATmega_DFP\1.3.300\avrasm\inc\m4809def.inc'
 6
 7
 8                                    ; table_lookup_seg_check.asm
 9                                    ;
10                                    ; Created: 10/20/2020 7:48:20 PM
11                                    ; Author : hp
12                                    ;
13
14                                    .list
15
16                                    start:
17  000000 e000                          ldi r16, $00
18  000001 ef1f                          ldi r17, $FF
19  000002 b900                          out VPORTA_DIR, r16
20  000003 b91c                          out VPORTD_DIR, r17
21  000004 9a47                          sbi VPORTC_DIR, 7
22  000005 9a4f                          sbi VPORTC_OUT, 7
23
24                                    main_loop:
25  000006 b102                          in r16, VPORTA_IN
26  000007 b91d                          out VPORTD_OUT, r17
27  000008 e010                          ldi r17, $00
28  000009 e028                          ldi r18, $08
29
30                                    reverse_bits:
31  00000a 9507                          ror r16
32  00000b 1f11                          rol r17
33  00000c 952a                          dec r18
34  00000d f7e1                          brne reverse_bits
35
36                                    hex_to_7seg:
37  00000e 701f                          andi r17, 0x0F              ;clear ms
      nibble
38  00000f e0f0                           ldi ZH, HIGH(hextable * 2)  ;set Z to
      point to start of table
39  000010 e2ec                           ldi ZL, LOW(hextable * 2)
40  000011 e000                           ldi r16, $00                ;add offset to
       Z pointer
```

```
000012 0fe1                         add ZL, r17
000013 1ff0                         adc ZH, r16
000014 9114                         lpm r17, Z                  ;load byte from table pointed to by Z
000015 cff0                        rjmp main_loop

                                    ;Table of segment values to display digits 0 - F
                                    ;!!! seven values must be added - verify all values
000016 4f01
000017 0612
000018 244c
000019 0f20
E:\ESE_280\$MyDocuments$\Atmel Studio\7.0\lab_7\table_lookup_seg_check\table_lookup_seg_check\main.asm(44): warning: .cseg .db misalignment - padding zero byte


RESOURCE USE INFORMATION
------------------------

Notice:
The register and instruction counts are symbol table hit counts,
and hence implicitly used resources are not counted, eg, the
'lpm' instruction without operands implicitly uses r0 and z,
none of which are counted.

x,y,z are separate entities in the symbol table and are
counted separately from r26..r31 here.

.dseg memory usage only counts static data declared with .byte

"ATmega4809" register use summary:
x  :   0 y  :   0 z  :   1 r0 :   0 r1 :   0 r2 :   0 r3 :   0 r4 :   0
r5 :   0 r6 :   0 r7 :   0 r8 :   0 r9 :   0 r10:   0 r11:   0 r12:   0
r13:   0 r14:   0 r15:   0 r16:   6 r17:   8 r18:   2 r19:   0 r20:   0
r21:   0 r22:   0 r23:   0 r24:   0 r25:   0 r26:   0 r27:   0 r28:   0
r29:   0 r30:   2 r31:   2
Registers used: 6 out of 35 (17.1%)

"ATmega4809" instruction use summary:
.lds  :   0 .sts  :   0 adc   :   1 add   :   1 adiw  :   0 and   :   0
andi  :   1 asr   :   0 bclr  :   0 bld   :   0 brbc  :   0 brbs  :   0
brcc  :   0 brcs  :   0 break :   0 breq  :   0 brge  :   0 brhc  :   0
brhs  :   0 brid  :   0 brie  :   0 brlo  :   0 brlt  :   0 brmi  :   0
brne  :   1 brpl  :   0 brsh  :   0 brtc  :   0 brts  :   0 brvc  :   0
brvs  :   0 bset  :   0 bst   :   0 call  :   0 cbi   :   0 cbr   :   0
clc   :   0 clh   :   0 cli   :   0 cln   :   0 clr   :   0 cls   :   0
```

```
 85  clt   :   0 clv   :   0 clz   :   0 com   :   0 cp    :   0 cpc   :   0
 86  cpi   :   0 cpse  :   0 dec   :   1 des   :   0 eor   :   0 fmul  :   0
 87  fmuls :   0 fmulsu:   0 icall :   0 ijmp  :   0 in    :   1 inc   :   0
 88  jmp   :   0 ld    :   0 ldd   :   0 ldi   :   7 lds   :   0 lpm   :   2
 89  lsl   :   0 lsr   :   0 mov   :   0 movw  :   0 mul   :   0 muls  :   0
 90  mulsu :   0 neg   :   0 nop   :   0 or    :   0 ori   :   0 out   :   3
 91  pop   :   0 push  :   0 rcall :   0 ret   :   0 reti  :   0 rjmp  :   1
 92  rol   :   1 ror   :   1 sbc   :   0 sbci  :   0 sbi   :   2 sbic  :   0
 93  sbis  :   0 sbiw  :   0 sbr   :   0 sbrc  :   0 sbrs  :   0 sec   :   0
 94  seh   :   0 sei   :   0 sen   :   0 ser   :   0 ses   :   0 set   :   0
 95  sev   :   0 sez   :   0 sleep :   0 spm   :   0 st    :   0 std   :   0
 96  sts   :   0 sub   :   0 subi  :   0 swap  :   0 tst   :   0 wdr   :   0
 97
 98  Instructions used: 13 out of 114 (11.4%)
 99
100  "ATmega4809" memory use summary [bytes]:
101  Segment   Begin    End      Code   Data   Used    Size   Use%
102  ----------------------------------------------------------------
103  [.cseg] 0x000000 0x000036     44     10     54   49152   0.1%
104  [.dseg] 0x002800 0x002800      0      0      0    6144   0.0%
105  [.eseg] 0x000000 0x000000      0      0      0     256   0.0%
106
107  Assembly complete, 0 errors, 1 warnings
108
```

```
 1
 2  AVRASM ver. 2.2.7  E:\ESE_280\$MyDocuments$\Atmel Studio\7.0\lab_7
      \display_hex_digit_at_pos\display_hex_digit_at_pos\main.asm Tue Oct 20
      20:48:20 2020
 3
 4  E:\ESE_280\$MyDocuments$\Atmel Studio\7.0\lab_7\display_hex_digit_at_pos
      \display_hex_digit_at_pos\main.asm(9): Including file 'C:/Program Files (x86)
      \Atmel\Studio\7.0\Packs\atmel\ATmega_DFP\1.3.300\avrasm\inc\m4809def.inc'
 5  E:\ESE_280\$MyDocuments$\Atmel Studio\7.0\lab_7\display_hex_digit_at_pos
      \display_hex_digit_at_pos\main.asm(9): Including file 'C:/Program Files (x86)
      \Atmel\Studio\7.0\Packs\atmel\ATmega_DFP\1.3.300\avrasm\inc\m4809def.inc'
 6
 7
 8                                    ; display_hex_digit_at_pos.asm
 9                                    ;
10                                    ; Created: 10/20/2020 8:30:20 PM
11                                    ; Author : hp
12                                    ;
13
14                                    .list
15
16                                    start:
17  000000 e000                          ldi r16, $00
18  000001 ef1f                          ldi r17, $FF
19  000002 b900                          out VPORTA_DIR, r16
20  000003 b91c                          out VPORTD_DIR, r17
21  000004 b918                          out VPORTC_DIR, r17
22  000005 b91d                          out VPORTD_OUT, r17
23
24                                    main_loop:
25  000006 b102                          in r16, VPORTA_IN
26  000007 1710                          cp r17, r16
27  000008 1720                          cp r18, r16
28  000009 7c10                          andi r17, $C0
29  00000a 700f                          andi r16, $0F
30  00000b 3010                          cpi r17, $00
31  00000c f039                          breq zero
32  00000d 3410                          cpi r17, $40
33  00000e f039                          breq one
34  00000f 3810                          cpi r17, $80
35  000010 f039                          breq two
36  000011 3c10                          cpi r17, $C0
37  000012 f039                          breq three
38  000013 cff2                          rjmp main_loop
39
40                                    zero:
41  000014 984f                          cbi VPORTC_OUT, 7
42  000015 c005                          rjmp hex_to_7seg
43
```

```
                                one:
000016 984e                        cbi VPORTC_OUT, 6
000017 c003                        rjmp hex_to_7seg

                                two:
000018 984d                        cbi VPORTC_OUT, 5
000019 c001                        rjmp hex_to_7seg

                                three:
00001a 984c                        cbi VPORTC_OUT, 4




                                ;****************************************************************************
                                ;*
                                ;* "hex_to_7seg" - Hexadecimal to Seven Segment Conversion
                                ;*
                                ;* Description: Converts a right justified hexadecimal digit to the seven
                                ;* segment pattern required to display it. Pattern is right justified a
                                ;* through g. Pattern uses 0s to turn segments on ON.
                                ;*
                                ;* Author:                      Ken Short
                                ;* Version:                         1.0
                                ;* Last updated:                101620
                                ;* Target:                      ATmega4809
                                ;* Number of words:                 8
                                ;* Number of cycles:            13
                                ;* Low registers modified:      none
                                ;* High registers modified:         r16, r18, ZL, ZH
                                ;*
                                ;* Parameters: r18: right justified hex digit, high nibble 0
                                ;* Returns: r18: segment values a through g right justified
                                ;*
                                ;* Notes:
                                ;*
                                ;****************************************************************************
                                hex_to_7seg:
00001b 702f                        andi r18, 0x0F              ;clear ms nibble
```

```
 82  00001c e0f0                          ldi ZH, HIGH(hextable * 2)  ;set Z to
       point to start of table
 83  00001d e4e6                          ldi ZL, LOW(hextable * 2)
 84  00001e e000                          ldi r16, $00                ;add offset to
        Z pointer
 85  00001f 0fe2                          add ZL, r18
 86  000020 1ff0                          adc ZH, r16
 87  000021 9124                          lpm r18, Z                  ;load byte
       from table pointed to by Z
 88  000022 c005                         rjmp output
 89
 90                                       ;Table of segment values to display digits
                       0 - F
 91                                       ;!!! seven values must be added - verify
                      all values
 92  000023 4f01
 93  000024 0612
 94  000025 244c
 95  000026 0f20
 96  E:\ESE_280\$MyDocuments$\Atmel Studio\7.0\lab_7\display_hex_digit_at_pos
       \display_hex_digit_at_pos\main.asm(88): warning: .cseg .db misalignment -
       padding zero byte
 97  000027 0000                     hextable: .db $01, $4F, $12, $06, $4C, $24,
       $20, $0F, $00
 98
 99                                   output:
100  000028 9520                        com r18
101  000029 b92d                        out VPORTD_OUT, r18
102
103
104  RESOURCE USE INFORMATION
105  ------------------------
106
107  Notice:
108  The register and instruction counts are symbol table hit counts,
109  and hence implicitly used resources are not counted, eg, the
110  'lpm' instruction without operands implicitly uses r0 and z,
111  none of which are counted.
112
113  x,y,z are separate entities in the symbol table and are
114  counted separately from r26..r31 here.
115
116  .dseg memory usage only counts static data declared with .byte
117
118  "ATmega4809" register use summary:
119  x  :   0 y  :   0 z  :   1 r0 :   0 r1 :   0 r2 :   0 r3 :   0 r4 :   0
120  r5 :   0 r6 :   0 r7 :   0 r8 :   0 r9 :   0 r10:   0 r11:   0 r12:   0
121  r13:   0 r14:   0 r15:   0 r16:   8 r17:  10 r18:   6 r19:   0 r20:   0
122  r21:   0 r22:   0 r23:   0 r24:   0 r25:   0 r26:   0 r27:   0 r28:   0
```

```
123 r29:   0 r30:   2 r31:   2
124 Registers used: 6 out of 35 (17.1%)
125
126 "ATmega4809" instruction use summary:
127 .lds  :   0 .sts  :   0 adc   :   1 add   :   1 adiw  :   0 and   :   0
128 andi  :   3 asr   :   0 bclr  :   0 bld   :   0 brbc  :   0 brbs  :   0
129 brcc  :   0 brcs  :   0 break :   0 breq  :   4 brge  :   0 brhc  :   0
130 brhs  :   0 brid  :   0 brie  :   0 brlo  :   0 brlt  :   0 brmi  :   0
131 brne  :   0 brpl  :   0 brsh  :   0 brtc  :   0 brts  :   0 brvc  :   0
132 brvs  :   0 bset  :   0 bst   :   0 call  :   0 cbi   :   4 cbr   :   0
133 clc   :   0 clh   :   0 cli   :   0 cln   :   0 clr   :   0 cls   :   0
134 clt   :   0 clv   :   0 clz   :   0 com   :   1 cp    :   2 cpc   :   0
135 cpi   :   4 cpse  :   0 dec   :   0 des   :   0 eor   :   0 fmul  :   0
136 fmuls :   0 fmulsu:   0 icall :   0 ijmp  :   0 in    :   1 inc   :   0
137 jmp   :   0 ld    :   0 ldd   :   0 ldi   :   5 lds   :   0 lpm   :   2
138 lsl   :   0 lsr   :   0 mov   :   0 movw  :   0 mul   :   0 muls  :   0
139 mulsu :   0 neg   :   0 nop   :   0 or    :   0 ori   :   0 out   :   5
140 pop   :   0 push  :   0 rcall :   0 ret   :   0 reti  :   0 rjmp  :   6
141 rol   :   0 ror   :   0 sbc   :   0 sbci  :   0 sbi   :   0 sbic  :   0
142 sbis  :   0 sbiw  :   0 sbr   :   0 sbrc  :   0 sbrs  :   0 sec   :   0
143 seh   :   0 sei   :   0 sen   :   0 ser   :   0 ses   :   0 set   :   0
144 sev   :   0 sez   :   0 sleep :   0 spm   :   0 st    :   0 std   :   0
145 sts   :   0 sub   :   0 subi  :   0 swap  :   0 tst   :   0 wdr   :   0
146
147 Instructions used: 13 out of 114 (11.4%)
148
149 "ATmega4809" memory use summary [bytes]:
150 Segment   Begin    End      Code   Data   Used    Size   Use%
151 ----------------------------------------------------------------
152 [.cseg] 0x000000 0x000056     76     10     86   49152   0.2%
153 [.dseg] 0x002800 0x002800      0      0      0    6144   0.0%
154 [.eseg] 0x000000 0x000000      0      0      0     256   0.0%
155
156 Assembly complete, 0 errors, 1 warnings
157
```