```
 1
 2  AVRASM ver. 2.2.7  E:\ESE_280\$MyDocuments$\Atmel Studio\7.0\lab_11
      \post_display\post_display\main.asm Tue Nov 17 11:36:27 2020
 3
 4  E:\ESE_280\$MyDocuments$\Atmel Studio\7.0\lab_11\post_display\post_display
      \main.asm(12): 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_11\post_display\post_display
      \main.asm(12): Including file 'C:/Program Files (x86)\Atmel\Studio\7.0\Packs
      \atmel\ATmega_DFP\1.3.300\avrasm\inc\m4809def.inc'
 6
 7
 8                                    ; post_display.asm
 9                                    ;
10                                    ; Created: 11/17/2020 11:13:12 AM
11                                    ; Author : hp
12                                    ;
13
14
15                                    ; Replace with your application code
16
17                                    .list
18
19                                    .equ PERIOD_EXAMPLE_VALUE = 25
20
21                                    reset:
22  000000 940c 0010                     jmp start
23
24                                    .org TCA0_OVF_vect
25  00000e 940c 002e                     jmp toggle_pins_ISR
26
27                                    start:
28                                       ;configure PORTC and PORTD and output FF to
                          both
29  000010 ef0f                          ldi r16, $FF
30  000011 b90c                          out VPORTD_DIR, r16
31  000012 b908                          out VPORTC_DIR, r16
32  000013 b90d                          out VPORTD_OUT, r16
33  000014 b909                          out VPORTC_OUT, r16
34
35                                       ;configure TCA0
36  000015 e000                          ldi r16, TCA_SINGLE_WGMODE_NORMAL_gc
      ;WGMODE normal
37  000016 9300 0a01                     sts TCA0_SINGLE_CTRLB, r16
38
39                                       ;enable overflow interrupt
40  000018 e001                          ldi r16, TCA_SINGLE_OVF_bm
41  000019 9300 0a0a                     sts TCA0_SINGLE_INTCTRL, r16
42
```

```
43                                      ;load period low byte then high byte
44  00001b e109                         ldi r16, LOW(PERIOD_EXAMPLE_VALUE)
45  00001c 9300 0a26                    sts TCA0_SINGLE_PER, r16
46  00001e e000                         ldi r16, HIGH(PERIOD_EXAMPLE_VALUE)
47  00001f 9300 0a27                    sts TCA0_SINGLE_PER + 1, r16
48
49                                      ;set clock and start timer
50  000021 e00d                         ldi r16, TCA_SINGLE_CLKSEL_DIV256_gc |
      TCA_SINGLE_ENABLE_bm
51  000022 9300 0a00                    sts TCA0_SINGLE_CTRLA, r16
52
53  000024 e000                         ldi r16, $00
54  000025 b90d                         out VPORTD_OUT, r16
55
56  000026 9478                         sei     ;enable global interrupts
57
58
59                                  ;*****************************************
                                ******************************
60                                  ;*
61                                  ;* "post_display"
62                                  ;*
63                                  ;* Description: toggles value for all PORTC
                                pins.  Since PORTC is used to multiplex the led display,
                                this will
64                                  ;* turn the LED display on and off
65                                  ;* Author:  Judah Ben-Eliezer
66                                  ;* Version:    1.0
67                                  ;* Last updated:   11/17
68                                  ;* Target:  ATmega4809
69                                  ;* Number of words:    13
70                                  ;* Number of cycles:   6
71                                  ;* Low registers modified:
72                                  ;* High registers modified:
73                                  ;* Parameters:  none
74                                  ;* Returns:        none
75                                  ;*
76                                  ;* Notes:
77                                  ;*
78                                  ;*****************************************
                                ******************************
79                                  post_display:
80  000027 ef1f                         ldi r17, $FF
81  000028 b109                         in r16, VPORTC_OUT
82  000029 2701                         eor r16, r17
83  00002a b909                         out VPORTC_OUT, r16
84  00002b 9508                         ret
85
86                                  main_loop:
```

```
 87  00002c 0000                        nop
 88  00002d cffe                        rjmp main_loop
 89
 90                                     ;*****************************************
                     ****************************
 91                                     ;*
 92                                     ;* "toggle_pins_ISR" - title
 93                                     ;*
 94                                     ;* Description:        ISR to toggle PORTC
                     pins, called whenever timing buffer TCA0 overflows
 95                                     ;*
 96                                     ;* Author: Judah Ben-Eliezer
 97                                     ;* Version:    1.0
 98                                     ;* Last updated:   11/17/2020
 99                                     ;* Target: ATmega4809
100                                     ;* Number of words:    27
101                                     ;* Number of cycles:   12
102                                     ;* Low registers modified:
103                                     ;* High registers modified:
104                                     ;*
105                                     ;* Parameters: none
106                                     ;* Returns:    none
107                                     ;*
108                                     ;* Notes:
109                                     ;*
110                                     ;*****************************************
                     ****************************
111                                    toggle_pins_ISR:
112 00002e 930f                           push r16
113 00002f b70f                           in r16, CPU_SREG
114 000030 930f                           push r16
115 000031 931f                           push r17
116
117 000032 dff4                           rcall post_display  ;call subroutine to
      toggle display
118
119 000033 e001                           ldi r16, TCA_SINGLE_OVF_bm  ;clear OVF flag
120 000034 9300 0a0b                      sts TCA0_SINGLE_INTFLAGS, r16
121
122 000036 911f                           pop r17
123 000037 910f                           pop r16
124 000038 bf0f                           out CPU_SREG, r16
125 000039 910f                           pop r16
126
127 00003a 9518                           reti
128
129
130 RESOURCE USE INFORMATION
131 ------------------------
```

```
132
133 Notice:
134 The register and instruction counts are symbol table hit counts,
135 and hence implicitly used resources are not counted, eg, the
136 'lpm' instruction without operands implicitly uses r0 and z,
137 none of which are counted.
138
139 x,y,z are separate entities in the symbol table and are
140 counted separately from r26..r31 here.
141
142 .dseg memory usage only counts static data declared with .byte
143
144 "ATmega4809" register use summary:
145 x  :   0 y  :   0 z  :   0 r0 :   0 r1 :   0 r2 :   0 r3 :   0 r4 :   0
146 r5 :   0 r6 :   0 r7 :   0 r8 :   0 r9 :   0 r10:   0 r11:   0 r12:   0
147 r13:   0 r14:   0 r15:   0 r16:  28 r17:   4 r18:   0 r19:   0 r20:   0
148 r21:   0 r22:   0 r23:   0 r24:   0 r25:   0 r26:   0 r27:   0 r28:   0
149 r29:   0 r30:   0 r31:   0
150 Registers used: 2 out of 35 (5.7%)
151
152 "ATmega4809" instruction use summary:
153 .lds  :   0 .sts  :   0 adc   :   0 add   :   0 adiw  :   0 and   :   0
154 andi  :   0 asr   :   0 bclr  :   0 bld   :   0 brbc  :   0 brbs  :   0
155 brcc  :   0 brcs  :   0 break :   0 breq  :   0 brge  :   0 brhc  :   0
156 brhs  :   0 brid  :   0 brie  :   0 brlo  :   0 brlt  :   0 brmi  :   0
157 brne  :   0 brpl  :   0 brsh  :   0 brtc  :   0 brts  :   0 brvc  :   0
158 brvs  :   0 bset  :   0 bst   :   0 call  :   0 cbi   :   0 cbr   :   0
159 clc   :   0 clh   :   0 cli   :   0 cln   :   0 clr   :   0 cls   :   0
160 clt   :   0 clv   :   0 clz   :   0 com   :   0 cp    :   0 cpc   :   0
161 cpi   :   0 cpse  :   0 dec   :   0 des   :   0 eor   :   1 fmul  :   0
162 fmuls :   0 fmulsu:   0 icall :   0 ijmp  :   0 in    :   2 inc   :   0
163 jmp   :   2 ld    :   0 ldd   :   0 ldi   :   9 lds   :   0 lpm   :   0
164 lsl   :   0 lsr   :   0 mov   :   0 movw  :   0 mul   :   0 muls  :   0
165 mulsu :   0 neg   :   0 nop   :   1 or    :   0 ori   :   0 out   :   7
166 pop   :   3 push  :   3 rcall :   1 ret   :   1 reti  :   1 rjmp  :   1
167 rol   :   0 ror   :   0 sbc   :   0 sbci  :   0 sbi   :   0 sbic  :   0
168 sbis  :   0 sbiw  :   0 sbr   :   0 sbrc  :   0 sbrs  :   0 sec   :   0
169 seh   :   0 sei   :   1 sen   :   0 ser   :   0 ses   :   0 set   :   0
170 sev   :   0 sez   :   0 sleep :   0 spm   :   0 st    :   0 std   :   0
171 sts   :   6 sub   :   0 subi  :   0 swap  :   0 tst   :   0 wdr   :   0
172
173 Instructions used: 14 out of 114 (12.3%)
174
175 "ATmega4809" memory use summary [bytes]:
176 Segment   Begin    End      Code   Data   Used    Size   Use%
177 ----------------------------------------------------------------
178 [.cseg] 0x000000 0x000076     94      0     94   49152   0.2%
179 [.dseg] 0x002800 0x002800      0      0      0    6144   0.0%
180 [.eseg] 0x000000 0x000000      0      0      0     256   0.0%
```

```
181
182  Assembly complete, 0 errors, 0 warnings
183
```