

...inven\enable\_pullups\_inven\Debug\enable\_pullups\_inven.lss 1

```
1
2 AVRASM ver. 2.2.7 E:\ESE_280\MyDocuments$\Atmel Studio\7.0\lab_8
  \enable_pullups_inven\enable_pullups_inven\main.asm Tue Oct 27 18:39:56 2020
3
4 E:\ESE_280\MyDocuments$\Atmel Studio\7.0\lab_8\enable_pullups_inven
  \enable_pullups_inven\main.asm(20): 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_8\enable_pullups_inven
  \enable_pullups_inven\main.asm(20): Including file 'C:/Program Files (x86)
  \Atmel\Studio\7.0\Packs\atmel\ATmega_DFP\1.3.300\avrasm\inc\m4809def.inc'
6
7
8      ;*
9      ;* Title: enable_pullups_inven.asm
10     ;* Author: Judah Ben-Eliezer
11     ;* Version: 1.0
12     ;* Last updated: 10/27/2020
13     ;* Target:
14     ;*
15     ;* DESCRIPTION
16     ;*
17     ;*
18     ;*
19     ;*
20     ;* VERSION HISTORY
21     ;* 1.0 Original version
22     ;* *****
    *****
23
24
25     .list
26
27
28     ; Replace with your application code
29     start:
30         ldi r16, $00
31         out VPORTA_DIR, r16
32         ldi XH, HIGH(PORTA_PIN0CTRL)
33         ldi XL, LOW(PORTA_PIN0CTRL)
34         ldi r17, 8
35
36     main_loop:
37         rcall pullups
38         rjmp main_loop
39
40     ;*****
    *****
41
42     ;*
43     ;* "pullups" - title
```

```

43      ;*
44      ;* Description:
45      ;*
46      ;* Author:  Judah Ben-Eliezer
47      ;* Version:   1.0
48      ;* Last updated:  10/27/2020
49      ;* Target:  AtMega4809
50      ;* Number of words:
51      ;* Number of cycles:
52      ;* Low registers modified:
53      ;* High registers modified:
54      ;*
55      ;* Parameters: none
56      ;* Returns:   none
57      ;*
58      ;* Notes:
59      ;*
60      ;*****
        *****
61
62      pullups:
63      000007 910c      ld r16, X
64      000008 6808      ori r16, $88
65      000009 930d      st X+, r16
66      00000a 951a      dec r17
67      00000b f7d9      brne pullups
68      00000c cff8      rjmp main_loop
69
70
71
72
73
74
75  RESOURCE USE INFORMATION
76  -----
77
78  Notice:
79  The register and instruction counts are symbol table hit counts,
80  and hence implicitly used resources are not counted, eg, the
81  'lpm' instruction without operands implicitly uses r0 and z,
82  none of which are counted.
83
84  x,y,z are separate entities in the symbol table and are
85  counted separately from r26..r31 here.
86
87  .dseg memory usage only counts static data declared with .byte
88
89  "ATmega4809" register use summary:
90  x  :   2  y  :   0  z  :   0  r0 :   0  r1 :   0  r2 :   0  r3 :   0  r4 :   0

```

```

91 r5 : 0 r6 : 0 r7 : 0 r8 : 0 r9 : 0 r10: 0 r11: 0 r12: 0
92 r13: 0 r14: 0 r15: 0 r16: 5 r17: 2 r18: 0 r19: 0 r20: 0
93 r21: 0 r22: 0 r23: 0 r24: 0 r25: 0 r26: 1 r27: 1 r28: 0
94 r29: 0 r30: 0 r31: 0

```

95 Registers used: 5 out of 35 (14.3%)

96

97 "ATmega4809" instruction use summary:

```

98 .lds : 0 .sts : 0 adc : 0 add : 0 adiw : 0 and : 0
99 andi : 0 asr : 0 bclr : 0 bld : 0 brbc : 0 brbs : 0
100 brcc : 0 brcs : 0 break : 0 breq : 0 brge : 0 brhc : 0
101 brhs : 0 brid : 0 brie : 0 brlo : 0 brlt : 0 brmi : 0
102 brne : 1 brpl : 0 brsh : 0 brtc : 0 brts : 0 brvc : 0
103 brvs : 0 bset : 0 bst : 0 call : 0 cbi : 0 cbr : 0
104 clc : 0 clh : 0 cli : 0 cln : 0 clr : 0 cls : 0
105 clt : 0 clv : 0 clz : 0 com : 0 cp : 0 cpc : 0
106 cpi : 0 cpse : 0 dec : 1 des : 0 eor : 0 fmul : 0
107 fmul : 0 fmul : 0 icall : 0 ijmp : 0 in : 0 inc : 0
108 jmp : 0 ld : 1 ldd : 0 ldi : 4 lds : 0 lpm : 0
109 lsl : 0 lsr : 0 mov : 0 movw : 0 mul : 0 muls : 0
110 mulsu : 0 neg : 0 nop : 0 or : 0 ori : 1 out : 1
111 pop : 0 push : 0 rcall : 1 ret : 0 reti : 0 rjmp : 2
112 rol : 0 ror : 0 sbc : 0 sbci : 0 sbi : 0 sbic : 0
113 sbis : 0 sbiw : 0 sbr : 0 sbrc : 0 sbrs : 0 sec : 0
114 seh : 0 sei : 0 sen : 0 ser : 0 ses : 0 set : 0
115 sev : 0 sez : 0 sleep : 0 spm : 0 st : 1 std : 0
116 sts : 0 sub : 0 subi : 0 swap : 0 tst : 0 wdr : 0

```

117

118 Instructions used: 9 out of 114 (7.9%)

119

120 "ATmega4809" memory use summary [bytes]:

Segment	Begin	End	Code	Data	Used	Size	Use%
[.cseg]	0x000000	0x00001a	26	0	26	49152	0.1%
[.dseg]	0x002800	0x002800	0	0	0	6144	0.0%
[.eseg]	0x000000	0x000000	0	0	0	256	0.0%

126

127 Assembly complete, 0 errors, 0 warnings

128