A51 MACRO ASSEMBLER WDT841 17/10/03 16:25:07 PAGE 1

DOS MACRO ASSEMBLER A51 V5.50

OBJECT MODULE PLACED IN WDT841.OBJ

ASSEMBLER INVOKED BY: C:\ADUC\BIN\A51.EXE WDT841.A51 DB EP

LOC OBJ LINE SOURCE

1 ;File: Wdt841.asm

2 ;Author: Eckart Hartmann Date:15/10/2003

3 ;Description of Software: Demonstrates Watchdog timer functions.

4 ;Development progress: Wdt834.df

5 ;

6 extrn CODE (\_WdtCfg) ;<A HREF="/mcc/softw/841/wdt/Wdt841Cfg.html">\_WdtCfg</A> in

ADuC834.lib

7 extrn CODE (\_WdtKk) ;<A HREF="/mcc/softw/841/wdt/Wdt841Kk.html">\_WdtKk</A>)

8 extrn CODE (\_PllDly) ;<A HREF="/mcc/softw/841/pll/Pll841Dly.html">\_PllDly</A>)

9

10 NAME WDT834

11 $NOMOD51

12 $IC(..\kei841.inc) ;<A HREF="/mcc/softw/841/Kei841Inc.html">Parameter passing

registers for Keil</A>.

=1 13 ;Functions: Keil specific definitions.

=1 14 ;File: Keil841.inc

=1 15 ;Author: Eckart Hartmann Date:16/10/2003

=1 16 ;Description of Software: None.

=1 17 ;Development progress: None.

=1 18 ;

=1 19 ;Interface Signals:

=1 20 ;==================

=1 21 ;Naming scheme:

=1 22 ; c = 1 byte, i = 2 byte, l = 4 byte, p = 3 byte pointer.

=1 23 ; First digit = type of variable.

=1 24 ; Second digit = P.

=1 25 ; Third digit = parameter number.

=1 26 ; Fourth digit = byte of parameter.

=1 27 ; Fifth digit = if any then type of first parameter.

=1 28 ; Sixth digit = if any then type of second parameter.

=1 29 ;

REG =1 30 cP1l equ r7 ;Byte parameter1 and return value.

REG =1 31 iP1l equ r7 ;Integer parameter1 and return value low byte.

REG =1 32 iP1h equ r6 ;Integer parameter1 and return value high byte.

REG =1 33 lP1l equ r7 ;Long parameter1 and return value low byte.

REG =1 34 lP1s equ r6 ;Long parameter1 and return value second byte.

REG =1 35 lP1t equ r5 ;Long parameter1 and return value third byte.

REG =1 36 lP1h equ r4 ;Long parameter1 and return value high byte.

REG =1 37 pP1t equ r3 ;Pointer parameter1 type byte.

REG =1 38 pP1h equ r2 ;Pointer parameter1 high byte.

REG =1 39 pP1l equ r1 ;Pointer parameter1 low byte.

=1 40

REG =1 41 cP2li equ r5 ;Byte parameter2 after integer.

REG =1 42 iP2li equ r5 ;Integer parameter2 low byte after integer.

REG =1 43 iP2hi equ r4 ;Integer parameter2 high byte after integer.

REG =1 44 lP2li equ r7 ;Long parameter2 low byte after integer.

REG =1 45 lP2si equ r6 ;Long parameter2 second byte after integer.

REG =1 46 lP2ti equ r5 ;Long parameter2 third byte after integer.

REG =1 47 lP2hi equ r4 ;Long parameter2 high byte after integer.

REG =1 48 pP2ti equ r3 ;Pointer parameter2 type byte after integer.

REG =1 49 pP2hi equ r2 ;Pointer parameter2 high byte after integer.

REG =1 50 pP2li equ r1 ;Pointer parameter2 low byte after integer.

REG =1 51 cP2lc equ r5 ;Byte parameter2 after byte.

REG =1 52 iP2lc equ r5 ;Integer parameter2 low byte after byte.

REG =1 53 iP2hc equ r4 ;Integer parameter2 high byte after byte.

REG =1 54 lP2lc equ r7 ;Long parameter2 low byte after byte.

REG =1 55 lP2sc equ r6 ;Long parameter2 second byte after byte.

REG =1 56 lP2tc equ r5 ;Long parameter2 third byte after byte.

A51 MACRO ASSEMBLER WDT841 17/10/03 16:25:07 PAGE 2

REG =1 57 lP2hc equ r4 ;Long parameter2 high byte after byte.

REG =1 58 pP2tc equ r3 ;Pointer parameter2 type byte after byte.

REG =1 59 pP2hc equ r2 ;Pointer parameter2 high byte after byte.

REG =1 60 pP2lc equ r1 ;Pointer parameter2 low byte after byte.

REG =1 61 cP2lp equ r5 ;Byte parameter2 after pointer.

REG =1 62 iP2lp equ r5 ;Integer parameter2 low byte after pointer.

REG =1 63 iP2hp equ r4 ;Integer parameter2 high byte after pointer.

REG =1 64 lP2lp equ r7 ;Long parameter2 low byte after pointer.

REG =1 65 lP2sp equ r6 ;Long parameter2 second byte after pointer.

REG =1 66 lP2tp equ r5 ;Long parameter2 third byte after pointer.

REG =1 67 lP2hp equ r4 ;Long parameter2 high byte after pointer.

=1 68

REG =1 69 cP3lii equ r3 ;Byte parameter3 after integer and integer.

REG =1 70 iP3lii equ r3 ;Integer parameter3 low byte after integer and integer.

REG =1 71 iP3hii equ r2 ;Integer parameter3 high byte after integer and integer.

REG =1 72 pP3tii equ r3 ;Pointer parameter3 type byte after integer and integer.

REG =1 73 pP3hii equ r2 ;Pointer parameter3 high byte after integer and integer.

REG =1 74 pP3lii equ r1 ;Pointer parameter3 low byte after integer and integer.

REG =1 75 cP3lic equ r3 ;Byte parameter3 after integer and byte.

REG =1 76 iP3lic equ r3 ;Integer parameter3 low byte after integer and byte.

REG =1 77 iP3hic equ r2 ;Integer parameter3 high byte after integer and byte.

REG =1 78 pP3tic equ r3 ;Pointer parameter3 type byte after integer and byte.

REG =1 79 pP3hic equ r2 ;Pointer parameter3 high byte after integer and byte.

REG =1 80 pP3lic equ r1 ;Pointer parameter3 low byte after integer and byte.

REG =1 81 cP3lci equ r3 ;Byte parameter3 after integer and integer.

REG =1 82 iP3lci equ r3 ;Integer parameter3 low byte after byte and integer.

REG =1 83 iP3hci equ r2 ;Integer parameter3 high byte after byte and integer.

REG =1 84 pP3tci equ r3 ;Pointer parameter3 type byte after byte and integer.

REG =1 85 pP3hci equ r2 ;Pointer parameter3 high byte after byte and integer.

REG =1 86 pP3lci equ r1 ;Pointer parameter3 low byte after byte and integer.

REG =1 87 cP3lcc equ r3 ;Byte parameter3 after integer and integer.

REG =1 88 iP3lcc equ r3 ;Integer parameter3 low byte after byte and byte.

REG =1 89 iP3hcc equ r2 ;Integer parameter3 high byte after byte and byte.

REG =1 90 pP3tcc equ r3 ;Pointer parameter3 type byte after byte and byte.

REG =1 91 pP3hcc equ r2 ;Pointer parameter3 high byte after byte and byte.

REG =1 92 pP3lcc equ r1 ;Pointer parameter3 low byte after byte and byte.

=1 93 ;

=1 94 ;Keil841.inc end==============================================Keil841.inc end

=1 95 ;

=1 96

97 $IC(..\kei841.dat) ;<A HREF="/mcc/softw/841/Kei841Dat.html">SFR definition for

Keil</A>.

=1 98 ;File kei841.dat

=1 99 ;SFR definitions for ADuC841 Assebler files.

=1 100 ;ADuC841 Apps, Analog Devices Inc.

=1 101 $NOMOD51

0084 =1 102 DPP DATA 084H ;DATA POINTER - PAGE BYTE

009D =1 103 T3FD DATA 09DH ;Serial baudrate fraction

009E =1 104 T3CON DATA 09EH ;Seral baudrate tap

00A1 =1 105 TIMECON DATA 0A1H ;TIME COUNTER CONTROL REGISTER

00A2 =1 106 HTHSEC DATA 0A2H ;1/128 OF A SECOND COUNTER

00A3 =1 107 SEC DATA 0A3H ;SECONDS COUNTER

00A4 =1 108 MIN DATA 0A4H ;MINUTES COUNTER

00A5 =1 109 HOUR DATA 0A5H ;HOURS COUNTER

00A6 =1 110 INTVAL DATA 0A6H ;TIMER INTERVAL

00A7 =1 111 DPCON DATA 0A7H ;Data Pointer control

00A9 =1 112 IEIP2 DATA 0A9H ;INTERRUPT ENABLE 2

00AE =1 113 PWMCON DATA 0AEH ;PWM control

00AF =1 114 CFG841 DATA 0AFH ;Configure ADuC841.

00B1 =1 115 PWM0L DATA 0B1H ;PWM width low

00B2 =1 116 PWM0H DATA 0B2H ;PWM width high

00B3 =1 117 PWM1L DATA 0B3H ;PWM cycle low

00B4 =1 118 PWM1H DATA 0B4H ;PWM cycle high

00B7 =1 119 SPH DATA 0B7H ;Stack pointer high

00B9 =1 120 ECON DATA 0B9H ;FLASH CONTROL

00BC =1 121 EDATA1 DATA 0BCH ;FLASH DATA1

A51 MACRO ASSEMBLER WDT841 17/10/03 16:25:07 PAGE 3

00BD =1 122 EDATA2 DATA 0BDH ;FLASH DATA2

00BE =1 123 EDATA3 DATA 0BEH ;FLASH DATA3

00BF =1 124 EDATA4 DATA 0BFH ;FLASH DATA4

00C0 =1 125 WDCON DATA 0C0H ;WATCHDOG TIMER CONTROL

00C2 =1 126 CHIPID DATA 0C2H ;CHIP ID REGISTER

00C6 =1 127 EADRL DATA 0C6H ;EEPROM ADDRESS LOW

00C7 =1 128 EADRH DATA 0C7H ;EEPROM ADDRESS HIGH

00C8 =1 129 T2CON DATA 0C8H ;Timer 2 control.

00CA =1 130 RCAP2L DATA 0caH ;Reload/capture low byte.

00CB =1 131 RCAP2H DATA 0cbH ;Reload/capture high byte.

00CC =1 132 TL2 DATA 0CcH ;Timer 2 low byte.

00CD =1 133 TH2 DATA 0CdH ;Timer 2 high byte.

00D2 =1 134 DMAL DATA 0D2H ;DMA low,

00D3 =1 135 DMAH DATA 0D3H ; high

00D4 =1 136 DMAP DATA 0D4H ; and page.

00D7 =1 137 PLLCON DATA 0D7H ;CRYSTAL PLL CONTROL REGISTER

00D8 =1 138 ADCCON2 DATA 0D8H ;ADC control 2

00D9 =1 139 ADCDATAL DATA 0D9H ;ADC DATA REGISTER

00DA =1 140 ADCDATAH DATA 0DAH ;ADC DATA REGISTER

00DF =1 141 PSMCON DATA 0DFH ;POWER SUPPLY MONITOR

00EF =1 142 ADCCON1 DATA 0efH ;ADC control 1

00F1 =1 143 ADCOFSL DATA 0f1H ;ADC offset low

00F2 =1 144 ADCOFSH DATA 0f2H ; and high.

00F3 =1 145 ADCGAINL DATA 0f3H ;ADC gain low

00F4 =1 146 ADCGAINH DATA 0f4H ; and high.

00F5 =1 147 ADCCON3 DATA 0f5H ;ADC control 3.

00F7 =1 148 SPIDAT DATA 0F7H ;SPI DATA REGISTER

00F8 =1 149 SPICON DATA 0F8H ;SPI CONTROL REGISTER

00F9 =1 150 DAC0L DATA 0f9H ;DAC0 LOW BYTE

00FA =1 151 DAC0H DATA 0faH ;DAC0 HIGH BYTE

00FB =1 152 DAC1L DATA 0fbH ;DAC1 LOW BYTE

00FC =1 153 DAC1H DATA 0fcH ;DAC1 HIGH BYTE

00FD =1 154 DACCON DATA 0FDH ;DAC CONTROL REGISTER

=1 155 ;

=1 156 ;Bits.

=1 157 ;

00AE =1 158 EADC BIT 0AEH ;IE.6 - ENABLE ADC INTURRUPT

00BF =1 159 PSI BIT 0BFH ;IP.7 - SPI OR 2-WIRE SERIAL INTERFACE PRIORITY

00C0 =1 160 WDWR BIT 0C0H ;WDCON.0 - WATCHDOG WRITE ENABLE BIT

00C1 =1 161 WDE BIT 0C1H ;WDCON.1 - WATCHDOG ENABLE

00C2 =1 162 WDS BIT 0C2H ;WDCON.2 - WATCHDOG STATUS

00C3 =1 163 WDIR BIT 0C3H ;WDCON.3 - WATCHDOG INTERRUPT RESPONSE BIT

00C5 =1 164 PRE0 BIT 0C5H ;WDCON.4 - WATCHDOG TIMEOUT SELECTION BIT0

00C6 =1 165 PRE1 BIT 0C6H ;WDCON.5 - WATCHDOG TIMEOUT SELECTION BIT1

00C7 =1 166 PRE2 BIT 0C7H ;WDCON.6 - WATCHDOG TIMEOUT SELECTION BIT2

00C8 =1 167 PRE3 BIT 0C8H ;WDCON.7 - WATCHDOG TIMEOUT SELECTION BIT3

00F8 =1 168 SPR0 BIT 0F8H ;SPICON.0 - SPI BITRATE SELECT BIT0

00F9 =1 169 SPR1 BIT 0F9H ;SPICON.1 - SPI BITRATE SELECT BIT1

00FA =1 170 CPHA BIT 0FAH ;SPICON.2 - SPI CLOCK PHASE SELECT

00FB =1 171 CPOL BIT 0FBH ;SPICON.3 - SPI CLOCK POLARITY SELECT

00FC =1 172 SPIM BIT 0FCH ;SPICON.4 - SPI MASTER/SLAVE MODE SELECT

00FD =1 173 SPE BIT 0FDH ;SPICON.5 - SPI INTERFACE ENABLE

00FE =1 174 WCOL BIT 0FEH ;SPICON.6 - SPI WRITE COLLISION ERROR FLAG

00FF =1 175 ISPI BIT 0FFH ;SPICON.7 - SPI INTERRUPT BIT

=1 176

=1 177 ;kei841.dat end================================================kei841.dat end

=1 178 ;

179 ;

---- 180 CSEG at 0000h

0000 02008E 181 rstorg: ljmp amain

182 ;

183 ;Function Start======================================================Function Start

184 ;==========Compiler Specifics:

185 ;

186 ;WdtInt==========Watchdog interrupt entry.

187 ;C Function prototype: interrupt void WdtInt(void);

A51 MACRO ASSEMBLER WDT841 17/10/03 16:25:07 PAGE 4

188 ;Description of Function: On interrupt it calls user watchdog handler.

189 ;User interface: User watchdog handler must be at WdtUInt.

190 ;Robustness: No known problems.

191 ;Side effects: Uses 2 stack levels.

192 ;

---- 193 CSEG at 0005bh ;WDT interrupt vector.

005B 12005F 194 WdtInt: lcall WdtUint ;Jump to user watchdog interrupt handler.

005E 32 195 reti

196 ;

197 ;WdtUint==========Watchdog user handler.

198 ;C Function prototype: char WdtUint(void);

199 ;Description of Function: User defined.

200 ;User interface: User defined.

201 ;Robustness: User defined.

202 ;Side effects: User defined.

203 ;

005F 204 WdtUint:;insert watchdog handler here.

005F B2B3 205 cpl P3.3 ;To show interrupt occured.

206 ;setb WDS

0061 22 207 ret

208 ;

209 ;Function End==========================================================Function End

210 ;Assembler main program.===========================================================

211 ;

212 ; org 00140h

213 ;

0062 1201B4 214 WdWas: lcall flick ;Flicker for 4s to indicate

0065 1201B4 215 lcall flick ; watchdog reset restart.

0068 E5A5 216 mov a,HOUR ;If HOUR=5

006A B4050C 217 cjne a,#5,WdFrc

006D 1201B4 218 WdEnd: lcall flick ; we are done.

0070 7FC8 219 mov ip1l,#200 ;200ms delay should do nothing.

0072 7E00 220 mov ip1h,#0

0074 120000 F 221 lcall \_PllDly

0077 80F4 222 sjmp WdEnd

0079 75A505 223 WdFrc: mov HOUR,#5 ;Else set HOUR as marker immune to reset.

007C C2B4 224 clr P3.4

007E 7FB8 225 mov ip1l,#0b8h ;3s delay should do nothing.

0080 7E0B 226 mov ip1h,#00bh

0082 120000 F 227 lcall \_PllDly

0085 7F82 228 mov cp1l,#082h ;Force reset.

0087 120000 F 229 lcall \_WdtCfg

008A 00 230 nop

008B 0201A0 231 ljmp WdErr ;If no reset occured then error stop.

232 ;

008E C2B3 233 amain: clr P3.3 ;Mark start.

0090 7FB8 234 mov ip1l,#0b8h ;3s delay should do nothing.

0092 7E0B 235 mov ip1h,#00bh

0094 120000 F 236 lcall \_PllDly

0097 C2B4 237 clr P3.4 ;Signal 3s delay over.

0099 20C2C6 238 jb WDS,WdWas ;If reset by watchdog, go to WdWas.

009C 75A500 239 mov HOUR,#0 ;else mark for first boot.

009F 75A400 240 mov MIN,#0

00A2 75A101 241 mov TIMECON,#1 ;Start timer else HOUR is lost by resets.

00A5 7F6A 242 mov cp1l,#06ah ;Start watchdog for 1s interrupt.

00A7 120000 F 243 lcall \_WdtCfg

00AA 7F58 244 mov ip1l,#058h ;Repeat 5 times: wait 600ms

00AC 7E02 245 mov ip1h,#002h

00AE 120000 F 246 lcall \_PllDly

00B1 D2B4 247 setb P3.4 ; flash 100ms

00B3 7F64 248 mov ip1l,#100

00B5 7E00 249 mov ip1h,#00

00B7 120000 F 250 lcall \_PllDly

00BA C2B4 251 clr P3.4

00BC 120000 F 252 lcall \_WdtKk ; kick watchdog before interrupt.

00BF 7F58 253 mov ip1l,#058h

A51 MACRO ASSEMBLER WDT841 17/10/03 16:25:07 PAGE 5

00C1 7E02 254 mov ip1h,#002h

00C3 120000 F 255 lcall \_PllDly

00C6 D2B4 256 setb P3.4

00C8 7F64 257 mov ip1l,#100

00CA 7E00 258 mov ip1h,#00

00CC 120000 F 259 lcall \_PllDly

00CF C2B4 260 clr P3.4

00D1 120000 F 261 lcall \_WdtKk

00D4 7F58 262 mov ip1l,#058h

00D6 7E02 263 mov ip1h,#002h

00D8 120000 F 264 lcall \_PllDly

00DB D2B4 265 setb P3.4

00DD 7F64 266 mov ip1l,#100

00DF 7E00 267 mov ip1h,#00

00E1 120000 F 268 lcall \_PllDly

00E4 C2B4 269 clr P3.4

00E6 120000 F 270 lcall \_WdtKk

00E9 7F58 271 mov ip1l,#058h

00EB 7E02 272 mov ip1h,#002h

00ED 120000 F 273 lcall \_PllDly

00F0 D2B4 274 setb P3.4

00F2 7F64 275 mov ip1l,#100

00F4 7E00 276 mov ip1h,#00

00F6 120000 F 277 lcall \_PllDly

00F9 C2B4 278 clr P3.4

00FB 120000 F 279 lcall \_WdtKk

00FE 7F58 280 mov ip1l,#058h

0100 7E02 281 mov ip1h,#002h

0102 120000 F 282 lcall \_PllDly

0105 D2B4 283 setb P3.4

0107 7F64 284 mov ip1l,#100

0109 7E00 285 mov ip1h,#00

010B 120000 F 286 lcall \_PllDly

010E C2B4 287 clr P3.4

0110 120000 F 288 lcall \_WdtKk

0113 7F4C 289 mov ip1l,#04ch ;If in 1.1s

0115 7E04 290 mov ip1h,#004h

0117 120000 F 291 lcall \_PllDly

292 ; lcall WdtRd ; no interrupt occured

011A AFC0 293 mov cp1l,WDCON

011C EF 294 mov a,cp1l

011D BF6A02 295 cjne cp1l,#06ah,WdEr1 ; then error stop.

296 ;; cjne cp1l,#06eh,WdEr1 ; then error stop.

0120 8003 297 sjmp WdFl1

0122 0201A0 298 WdEr1: ljmp WdErr

0125 1201B4 299 WdFl1: lcall flick ;else flicker.

0128 7F62 300 mov cp1l,#062h ;Start watchdog for 1s reset.

012A 120000 F 301 lcall \_WdtCfg

012D 7F58 302 mov ip1l,#058h ;Repeat 5 times: wait 600ms

012F 7E00 303 mov ip1h,#00h

0131 120000 F 304 lcall \_PllDly

0134 D2B4 305 setb P3.4 ; flash 100ms

0136 7F64 306 mov ip1l,#100

0138 7E00 307 mov ip1h,#00

013A 120000 F 308 lcall \_PllDly

013D C2B4 309 clr P3.4

013F 120000 F 310 lcall \_WdtKk ; kick watchdog before interrupt.

0142 7F58 311 mov ip1l,#058h

0144 7E02 312 mov ip1h,#002h

0146 120000 F 313 lcall \_PllDly

0149 D2B4 314 setb P3.4

014B 7F64 315 mov ip1l,#100

014D 7E00 316 mov ip1h,#00

014F 120000 F 317 lcall \_PllDly

0152 C2B4 318 clr P3.4

0154 120000 F 319 lcall \_WdtKk

A51 MACRO ASSEMBLER WDT841 17/10/03 16:25:07 PAGE 6

0157 7F58 320 mov ip1l,#058h

0159 7E02 321 mov ip1h,#002h

015B 120000 F 322 lcall \_PllDly

015E D2B4 323 setb P3.4

0160 7F64 324 mov ip1l,#100

0162 7E00 325 mov ip1h,#00

0164 120000 F 326 lcall \_PllDly

0167 C2B4 327 clr P3.4

0169 120000 F 328 lcall \_WdtKk

016C 7F58 329 mov ip1l,#058h

016E 7E02 330 mov ip1h,#002h

0170 120000 F 331 lcall \_PllDly

0173 D2B4 332 setb P3.4

0175 7F64 333 mov ip1l,#100

0177 7E00 334 mov ip1h,#00

0179 120000 F 335 lcall \_PllDly

017C C2B4 336 clr P3.4

017E 120000 F 337 lcall \_WdtKk

0181 7F58 338 mov ip1l,#058h

0183 7E02 339 mov ip1h,#002h

0185 120000 F 340 lcall \_PllDly

0188 D2B4 341 setb P3.4

018A 7F64 342 mov ip1l,#100

018C 7E00 343 mov ip1h,#00

018E 120000 F 344 lcall \_PllDly

0191 C2B4 345 clr P3.4

0193 120000 F 346 lcall \_WdtKk

0196 7F4C 347 mov ip1l,#04ch ;If in 1.1s

0198 7E04 348 mov ip1h,#004h

019A 120000 F 349 lcall \_PllDly

019D 0201A0 350 ljmp WdErr ; no reset occured then error stop.

351 ;

01A0 D2B4 352 WdErr: setb P3.4 ;Signal error

01A2 7F1E 353 mov ip1l,#30 ; 100ms delay.

01A4 7E00 354 mov ip1h,#0

01A6 120000 F 355 lcall \_PllDly

01A9 C2B4 356 clr P3.4

01AB 7F00 357 mov ip1l,#0 ; ~2s delay.

01AD 7E08 358 mov ip1h,#8

01AF 120000 F 359 lcall \_PllDly

01B2 80EC 360 sjmp WdErr ; continuously.

361 ;

01B4 7A21 362 flick: mov r2,#33 ;Flicker P3.4 for 2 seconds.

01B6 7F1E 363 flickL: mov ip1l,#30

01B8 7E00 364 mov ip1h,#00

01BA 120000 F 365 lcall \_PllDly

01BD D2B4 366 setb P3.4

01BF 120000 F 367 lcall \_WdtKk

01C2 7F1E 368 mov ip1l,#30

01C4 7E00 369 mov ip1h,#00

01C6 120000 F 370 lcall \_PllDly

01C9 C2B4 371 clr P3.4

01CB DAE9 372 djnz r2,flickL

01CD 22 373 ret

374 END

A51 MACRO ASSEMBLER WDT841 17/10/03 16:25:07 PAGE 7

SYMBOL TABLE LISTING

------ ----- -------

N A M E T Y P E V A L U E ATTRIBUTES

ADCCON1. . . . . . D ADDR 00EFH A

ADCCON2. . . . . . D ADDR 00D8H A

ADCCON3. . . . . . D ADDR 00F5H A

ADCDATAH . . . . . D ADDR 00DAH A

ADCDATAL . . . . . D ADDR 00D9H A

ADCGAINH . . . . . D ADDR 00F4H A

ADCGAINL . . . . . D ADDR 00F3H A

ADCOFSH. . . . . . D ADDR 00F2H A

ADCOFSL. . . . . . D ADDR 00F1H A

AMAIN. . . . . . . C ADDR 008EH A

CFG841 . . . . . . D ADDR 00AFH A

CHIPID . . . . . . D ADDR 00C2H A

CP1L . . . . . . . REG R7

CP2LC. . . . . . . REG R5

CP2LI. . . . . . . REG R5

CP2LP. . . . . . . REG R5

CP3LCC . . . . . . REG R3

CP3LCI . . . . . . REG R3

CP3LIC . . . . . . REG R3

CP3LII . . . . . . REG R3

CPHA . . . . . . . B ADDR 00F8H.2 A

CPOL . . . . . . . B ADDR 00F8H.3 A

DAC0H. . . . . . . D ADDR 00FAH A

DAC0L. . . . . . . D ADDR 00F9H A

DAC1H. . . . . . . D ADDR 00FCH A

DAC1L. . . . . . . D ADDR 00FBH A

DACCON . . . . . . D ADDR 00FDH A

DMAH . . . . . . . D ADDR 00D3H A

DMAL . . . . . . . D ADDR 00D2H A

DMAP . . . . . . . D ADDR 00D4H A

DPCON. . . . . . . D ADDR 00A7H A

DPP. . . . . . . . D ADDR 0084H A

EADC . . . . . . . B ADDR 00A8H.6 A

EADRH. . . . . . . D ADDR 00C7H A

EADRL. . . . . . . D ADDR 00C6H A

ECON . . . . . . . D ADDR 00B9H A

EDATA1 . . . . . . D ADDR 00BCH A

EDATA2 . . . . . . D ADDR 00BDH A

EDATA3 . . . . . . D ADDR 00BEH A

EDATA4 . . . . . . D ADDR 00BFH A

FLICK. . . . . . . C ADDR 01B4H A

FLICKL . . . . . . C ADDR 01B6H A

HOUR . . . . . . . D ADDR 00A5H A

HTHSEC . . . . . . D ADDR 00A2H A

IEIP2. . . . . . . D ADDR 00A9H A

INTVAL . . . . . . D ADDR 00A6H A

IP1H . . . . . . . REG R6

IP1L . . . . . . . REG R7

IP2HC. . . . . . . REG R4

IP2HI. . . . . . . REG R4

IP2HP. . . . . . . REG R4

IP2LC. . . . . . . REG R5

IP2LI. . . . . . . REG R5

IP2LP. . . . . . . REG R5

IP3HCC . . . . . . REG R2

IP3HCI . . . . . . REG R2

IP3HIC . . . . . . REG R2

IP3HII . . . . . . REG R2

IP3LCC . . . . . . REG R3

IP3LCI . . . . . . REG R3

A51 MACRO ASSEMBLER WDT841 17/10/03 16:25:07 PAGE 8

IP3LIC . . . . . . REG R3

IP3LII . . . . . . REG R3

ISPI . . . . . . . B ADDR 00F8H.7 A

LP1H . . . . . . . REG R4

LP1L . . . . . . . REG R7

LP1S . . . . . . . REG R6

LP1T . . . . . . . REG R5

LP2HC. . . . . . . REG R4

LP2HI. . . . . . . REG R4

LP2HP. . . . . . . REG R4

LP2LC. . . . . . . REG R7

LP2LI. . . . . . . REG R7

LP2LP. . . . . . . REG R7

LP2SC. . . . . . . REG R6

LP2SI. . . . . . . REG R6

LP2SP. . . . . . . REG R6

LP2TC. . . . . . . REG R5

LP2TI. . . . . . . REG R5

LP2TP. . . . . . . REG R5

MIN. . . . . . . . D ADDR 00A4H A

P3 . . . . . . . . D ADDR 00B0H A

PLLCON . . . . . . D ADDR 00D7H A

PP1H . . . . . . . REG R2

PP1L . . . . . . . REG R1

PP1T . . . . . . . REG R3

PP2HC. . . . . . . REG R2

PP2HI. . . . . . . REG R2

PP2LC. . . . . . . REG R1

PP2LI. . . . . . . REG R1

PP2TC. . . . . . . REG R3

PP2TI. . . . . . . REG R3

PP3HCC . . . . . . REG R2

PP3HCI . . . . . . REG R2

PP3HIC . . . . . . REG R2

PP3HII . . . . . . REG R2

PP3LCC . . . . . . REG R1

PP3LCI . . . . . . REG R1

PP3LIC . . . . . . REG R1

PP3LII . . . . . . REG R1

PP3TCC . . . . . . REG R3

PP3TCI . . . . . . REG R3

PP3TIC . . . . . . REG R3

PP3TII . . . . . . REG R3

PRE0 . . . . . . . B ADDR 00C0H.5 A

PRE1 . . . . . . . B ADDR 00C0H.6 A

PRE2 . . . . . . . B ADDR 00C0H.7 A

PRE3 . . . . . . . B ADDR 00C8H.0 A

PSI. . . . . . . . B ADDR 00B8H.7 A

PSMCON . . . . . . D ADDR 00DFH A

PWM0H. . . . . . . D ADDR 00B2H A

PWM0L. . . . . . . D ADDR 00B1H A

PWM1H. . . . . . . D ADDR 00B4H A

PWM1L. . . . . . . D ADDR 00B3H A

PWMCON . . . . . . D ADDR 00AEH A

RCAP2H . . . . . . D ADDR 00CBH A

RCAP2L . . . . . . D ADDR 00CAH A

RSTORG . . . . . . C ADDR 0000H A

SEC. . . . . . . . D ADDR 00A3H A

SPE. . . . . . . . B ADDR 00F8H.5 A

SPH. . . . . . . . D ADDR 00B7H A

SPICON . . . . . . D ADDR 00F8H A

SPIDAT . . . . . . D ADDR 00F7H A

SPIM . . . . . . . B ADDR 00F8H.4 A

SPR0 . . . . . . . B ADDR 00F8H.0 A

SPR1 . . . . . . . B ADDR 00F8H.1 A

T2CON. . . . . . . D ADDR 00C8H A

A51 MACRO ASSEMBLER WDT841 17/10/03 16:25:07 PAGE 9

T3CON. . . . . . . D ADDR 009EH A

T3FD . . . . . . . D ADDR 009DH A

TH2. . . . . . . . D ADDR 00CDH A

TIMECON. . . . . . D ADDR 00A1H A

TL2. . . . . . . . D ADDR 00CCH A

WCOL . . . . . . . B ADDR 00F8H.6 A

WDCON. . . . . . . D ADDR 00C0H A

WDE. . . . . . . . B ADDR 00C0H.1 A

WDEND. . . . . . . C ADDR 006DH A

WDER1. . . . . . . C ADDR 0122H A

WDERR. . . . . . . C ADDR 01A0H A

WDFL1. . . . . . . C ADDR 0125H A

WDFRC. . . . . . . C ADDR 0079H A

WDIR . . . . . . . B ADDR 00C0H.3 A

WDS. . . . . . . . B ADDR 00C0H.2 A

WDT834 . . . . . . N NUMB -----

WDTINT . . . . . . C ADDR 005BH A

WDTUINT. . . . . . C ADDR 005FH A

WDWAS. . . . . . . C ADDR 0062H A

WDWR . . . . . . . B ADDR 00C0H.0 A

\_PLLDLY. . . . . . C ADDR ----- EXT

\_WDTCFG. . . . . . C ADDR ----- EXT

\_WDTKK . . . . . . C ADDR ----- EXT

REGISTER BANK(S) USED: 0

ASSEMBLY COMPLETE. 0 WARNING(S), 0 ERROR(S)