824CTEST PAGE 1

1 ;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

2 ;

3 ; Author : ADI - Apps

4 ;

5 ; Date : January 2001

6 ;

7 ; File : 824CTest.asm

8 ;

9 ; Description : QuickStart development kit basic test routine.

10 ;

11 ;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

12

13 $MOD824 ; Use predefined Symbols

14

0000 15 FLAG BIT 00h

0001 16 FIRSTGO BIT 01H

00B4 17 LED EQU P3.4

18

19 ;\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

20 ; DEFINE VARIABLES IN INTERNAL RAM

---- 21 DSEG

22

0060 23 ORG 0060h

0060 24 COUNT: DS 1 ; define 1 byte for the count

0061 25 DACOUT: DS 1 ; define 1 byte for the DAC o/p

0062 26 DPH1: DS 1 ; \ define 2 bytes for the

0063 27 DPL1: DS 1 ; / temporary storage of DPTR

0064 28 DATAOUT: DS 1 ; the current data put into mem

29

30 ;\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

---- 31 CSEG ; (beginning of code)

32

0000 33 ORG 0000h

0000 020060 34 JMP MAIN

35

36 ;\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

37

0003 38 ORG 0003h ; (INT0 ISR)

0003 C200 39 CLR FLAG ; clear flag

0005 32 40 RETI

41 ;\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

42

0060 43 ORG 0060h ; (subroutines...)

44

45

46 ;\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

47

0060 48 MAIN: ; (main program)

49

0060 758180 50 MOV SP,#80h

51

52 ; CONFIGURE UART

0063 75CBFF 53 MOV RCAP2H,#0FFh ; config UART for 9830baud

0066 75CAFB 54 MOV RCAP2L,#-5 ; (close enough to 9600baud)

0069 75CDFF 55 MOV TH2,#0FFh

006C 75CCFB 56 MOV TL2,#-5

006F 759852 57 MOV SCON,#52h

0072 75C834 58 MOV T2CON,#34h

824CTEST PAGE 2

59

60 ; CONFIGURE INTERRUPT 0...

61

0075 D288 62 SETB IT0 ; INT0 edge triggered

0077 D2A8 63 SETB EX0 ; enable INT0 interrupt

64

65 ; ENABLE INTERRUPTS & ENTER MAIN LOOP...

66

0079 D2AF 67 SETB EA ; enable inturrupts

68

007B D200 69 SETB FLAG

007D 7401 70 MOV A, #01h

71

007F D299 72 SETB TI ; set TI flag to indicate that the

73 ; buffer is ready to transmit data.

0081 900C8E 74 MOV DPTR, #GOMENU ; send a press INT0 message

0084 120D62 75 CALL SENDSTRING

0087 D201 76 SETB FIRSTGO

77

0089 B2B4 78 BLINK: CPL LED ; blink LED until INT0 button pressed

008B 120D56 79 CALL DELAY

008E 2000F8 80 JB FLAG, BLINK

81

0091 D2B4 82 SETB LED ; turn on LED

83 ;\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

0093 84 MAINMENU:

0093 20010E 85 JB FIRSTGO, SKIP1

0096 900CB4 86 MOV DPTR, #RETMENU ; send a press INT0 message

0099 120D62 87 CALL SENDSTRING

009C D200 88 SETB FLAG

009E 2000FD 89 JB FLAG, $ ; wait for INT0

00A1 0200B8 90 JMP SKIP2

00A4 91 SKIP1:

00A4 C201 92 CLR FIRSTGO

00A6 900403 93 MOV DPTR,#TITLE ; display title

00A9 120D62 94 CALL SENDSTRING

00AC 90051A 95 MOV DPTR,#LINKS1 ; display link options

00AF 120D62 96 CALL SENDSTRING

00B2 9005B1 97 MOV DPTR,#LINKS2 ; display link options

00B5 120D62 98 CALL SENDSTRING

00B8 99 SKIP2:

00B8 900479 100 MOV DPTR,#MENU ; display menu

00BB 120D62 101 CALL SENDSTRING

102

00BE 3098FD 103 JNB RI, $ ; wait here for a key press

00C1 E599 104 MOV A, SBUF ; read input

00C3 C298 105 CLR RI ; clear RI flag for the next input

106

00C5 B43103 107 CJNE A, #'1', $+6 ; if a 1 (31H) is pressed jmp TEST1

00C8 0200E5 108 JMP TEST1

00CB B43203 109 CJNE A, #'2', $+6 ; if a 2 (32H) is pressed jmp TEST2

00CE 0201D4 110 JMP TEST2

00D1 B43303 111 CJNE A, #'3', $+6 ; if a 3 (33H) is pressed jmp TEST3

00D4 0202CD 112 JMP TEST3

00D7 B43403 113 CJNE A, #'4', $+6 ; if a 4 (34H) is pressed jmp TEST4

00DA 02034A 114 JMP TEST4

00DD B43503 115 CJNE A, #'5', $+6 ; if a 5 (35H) is pressed jmp TEST5

00E0 020394 116 JMP TEST5

824CTEST PAGE 3

00E3 80AE 117 JMP MAINMENU

118

119 ;====================================================================

120 ; TEST 1: ADC AND DAC TEST

121 ;====================================================================

122

00E5 123 TEST1:

124

125

00E5 900738 126 MOV DPTR,#PRIMADC ; display ADC title

00E8 120D62 127 CALL SENDSTRING

128

00EB 75FD17 129 MOV DACCON, #17h ; configure DAC for

130 ; 0-Vdd (5V), 12 bits,

00EE 75D247 131 MOV ADC0CON, #47h ; configure primary ADC for

132 ; external ref 2.5V, Ain1-Ain2,

133 ; bipolar, +-2.56V

00F1 D200 134 SETB FLAG

00F3 2000FD 135 JB FLAG, $ ; wait for INT0

136

00F6 75610C 137 MOV DACOUT, #0Ch ; start of prim ADC conv

00F9 756005 138 MOV COUNT, #05h ; # of conv

139

00FC 120135 140 CALL TESTPRIMADC

141

142 ;aux channel Ain3

00FF 90080A 143 MOV DPTR,#AUX3ADC

0102 120D62 144 CALL SENDSTRING

145

0105 75D348 146 MOV ADC1CON,#48h ; AIN3, unipolar, ext ref

0108 75FD13 147 MOV DACCON,#13h ; turn DAC on, 0-2.5V

148

010B D200 149 SETB FLAG

010D 2000FD 150 JB FLAG, $ ; wait for INT0

151

0110 756105 152 MOV DACOUT, #05H

0113 120172 153 CALL TESTAUXCHAN

154

155 ;aux channel Ain5

0116 90086A 156 MOV DPTR,#AUX5ADC

0119 120D62 157 CALL SENDSTRING

158

011C 75D378 159 MOV ADC1CON,#78h ; AIN5, unipolar, ext ref

011F 75FD13 160 MOV DACCON,#13h ; turn DAC on, 0-2.5V

161

0122 D200 162 SETB FLAG

0124 2000FD 163 JB FLAG, $ ; wait for INT0

164

0127 756105 165 MOV DACOUT, #05H

012A 120172 166 CALL TESTAUXCHAN

167

168 ; END OF TEST 1

012D 75D100 169 MOV ADCMODE,#00h ; turn ADCs off

0130 75FD00 170 MOV DACCON, #00h ; turn offthe DAC

0133 0193 171 JMP MAINMENU ; return

172

173

174 ;\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

824CTEST PAGE 4

175

0135 176 TESTPRIMADC:

177 ; output voltage on DAC

0135 8561FC 178 MOV DACH,DACOUT ; change hi-byte only

0138 75FB00 179 MOV DACL,#00h

013B 7810 180 MOV R0,#10h ; allow for settling

013D D8FE 181 DJNZ R0,$

182

183 ; do a single conversion on the DAC voltage. Ain2 biased to

184 ; 2.5V. Hence the ADC conv voltage is the DAC voltage - 2.5V

013F 75D122 185 MOV ADCMODE,#22h ; single conv on prim chan

0142 30DFFD 186 JNB RDY0,$ ; wait for ADC result

0145 C2DF 187 CLR RDY0

188

189 ; display the DAC output on the screen

0147 900CDC 190 MOV DPTR,#DACMSG ; "DAC="

014A 120D62 191 CALL SENDSTRING

014D 7E00 192 MOV R6, #00h

014F AF61 193 MOV R7, DACOUT ; display DAC value

0151 120DBD 194 CALL SEND12BITS

195

196 ; display the ADC conv result on the screen

0154 900CE2 197 MOV DPTR,#ADC0MSG ; " -> ADC0 = "

0157 120D62 198 CALL SENDSTRING

199

015A E5DB 200 MOV A, ADC0H ; send the 24 bit result upo the UART

015C 120D92 201 CALL SENDVAL

015F E5DA 202 MOV A, ADC0M

0161 120D92 203 CALL SENDVAL

0164 E5D9 204 MOV A, ADC0L

0166 120D92 205 CALL SENDVAL

206

0169 1201A8 207 CALL CHECKADC0 ; see if results are "OK" or "ERROR!"

208

016C 1561 209 DEC DACOUT

210

016E D560C4 211 DJNZ COUNT, TESTPRIMADC

0171 22 212 RET

213

214

0172 215 TESTAUXCHAN:

216 ; output voltage on DAC

0172 8561FC 217 MOV DACH,DACOUT ; change hi-byte only

0175 75FB00 218 MOV DACL,#00h ; load DAC

0178 7810 219 MOV R0,#10h ; allow for settling

017A D8FE 220 DJNZ R0, $

221

222 ; do a single conversion on the DAC voltage. The DAC voltage

223 ; appears between Ain3 and ground. Therefore the aux conv

224 ; should equal the DAC o/p

017C 75D112 225 MOV ADCMODE,#12h ; single conv on aux chan

017F 30DEFD 226 JNB RDY1,$ ; wait for ADC result

0182 C2DE 227 CLR RDY1

228

229 ; display the DAC output on the screen

0184 900CDC 230 MOV DPTR,#DACMSG ; "DAC="

0187 120D62 231 CALL SENDSTRING

018A 7E00 232 MOV R6, #00h

824CTEST PAGE 5

018C AF61 233 MOV R7, DACOUT ; display DAC value

018E 120DBD 234 CALL SEND12BITS

235

236 ; display the ADC conv result on the screen

0191 900CF3 237 MOV DPTR,#ADC1MSG ; "ADC1="

0194 120D62 238 CALL SENDSTRING

0197 E5DD 239 MOV A, ADC1H ; display ADC results

0199 120D92 240 CALL SENDVAL

019C E5DC 241 MOV A, ADC1L

019E 120D92 242 CALL SENDVAL

243

01A1 1201BE 244 CALL CHECKADC1 ; "ok" or "ERROR!"

245

01A4 D561CB 246 DJNZ DACOUT, TESTAUXCHAN

01A7 22 247 RET

248 ;\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

01A8 249 CHECKADC0:

01A8 E561 250 MOV A, DACOUT

01AA C4 251 SWAP A ; Swap Daccout from 0Ch to C0h etc

01AB 9408 252 SUBB A, #08h

01AD FA 253 MOV R2, A ; R2 holds B8h (for DACH=0C)

01AE E5DB 254 MOV A, ADC0H ; A holds say C1h

01B0 9A 255 SUBB A, R2

01B1 B41000 256 CJNE A, #10h, $+3 ; C is set if A<10H i.e. OK

01B4 4004 257 JC $+6

01B6 120D4F 258 CALL PRINTERROR

01B9 22 259 RET

01BA 120D48 260 CALL PRINTOK

01BD 22 261 RET

262 ;\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

01BE 263 CHECKADC1:

264 ; check if the ADC result is within 8LSBs of the DAC op

01BE E561 265 MOV A, DACOUT

01C0 C4 266 SWAP A

01C1 9408 267 SUBB A, #08h

01C3 FA 268 MOV R2, A

01C4 E5DD 269 MOV A, ADC1H

01C6 9A 270 SUBB A, R2

01C7 B41000 271 CJNE A, #10h, $+3 ; C is set if A<10H i.e. OK

01CA 4004 272 JC $+6

01CC 120D4F 273 CALL PRINTERROR

01CF 22 274 RET

01D0 120D48 275 CALL PRINTOK

01D3 22 276 RET

277 ;\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

278

279

280

281 ;====================================================================

282 ; TEST 2: EXTERNAL DATA MEMORY TEST

283 ;====================================================================

284

01D4 285 TEST2:

286 ; save particular data into internal RAM location 50h->59h

01D4 287 EXTDATA:

01D4 7850 288 MOV R0, #50h

01D6 76FF 289 MOV @R0, #0FFh ; store FFh into 50H

01D8 08 290 INC R0

824CTEST PAGE 6

01D9 7600 291 MOV @R0, #00h ; store 00h into 51h

01DB 08 292 INC R0

01DC 76AA 293 MOV @R0, #0AAh ; store AAh into 52h

01DE 08 294 INC R0

01DF 7655 295 MOV @R0, #55h ; store 55h into 53h

01E1 08 296 INC R0

01E2 76CC 297 MOV @R0, #0CCh ; store CCh into 54h

01E4 08 298 INC R0

01E5 7633 299 MOV @R0, #33h ; store 33h into 55h

01E7 08 300 INC R0

01E8 763F 301 MOV @R0, #3Fh ; store 3Fh into 56h

01EA 08 302 INC R0

01EB 7666 303 MOV @R0, #66h ; store 66h into 57h

304

01ED 9008C9 305 MOV DPTR,#MEMTITLE ; pick a memory mode

01F0 120D62 306 CALL SENDSTRING

307 ;\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

308

01F3 309 LK9A:

01F3 900936 310 MOV DPTR,#LINK9A ; "LK9 to position A"

01F6 120D62 311 CALL SENDSTRING

01F9 D200 312 SETB FLAG

01FB 2000FD 313 JB FLAG, $ ; wait for INT0

314

01FE 758400 315 MOV DPP,#00h

0201 7850 316 MOV R0, #50h

0203 9005FF 317 MOV DPTR, #05FFH

318

319 ; save data in RAM into the following locations

0206 320 XRAMIN1:

0206 758400 321 MOV DPP,#00h

0209 E6 322 MOV A, @R0

020A 8664 323 MOV DATAOUT, @R0

020C F0 324 MOVX @DPTR, A

020D 858362 325 MOV DPH1, DPH ; DPH1 and DPL1 hold the present

0210 858263 326 MOV DPL1, DPL ; address in XRAM

327

0213 900D2C 328 MOV DPTR, #ADDRMSG

0216 120D62 329 CALL SENDSTRING

330

0219 E584 331 MOV A, DPP

021B 120D92 332 CALL SENDVAL

021E E562 333 MOV A, DPH1

0220 120D92 334 CALL SENDVAL

0223 E563 335 MOV A, DPL1

0225 120D92 336 CALL SENDVAL

337

0228 900D34 338 MOV DPTR, #WRITTENMSG

022B 120D62 339 CALL SENDSTRING

340

022E E564 341 MOV A, DATAOUT

0230 120D92 342 CALL SENDVAL

343

0233 900D40 344 MOV DPTR, #READMSG

0236 120D62 345 CALL SENDSTRING

346

0239 758455 347 MOV DPP, #55H ; change the page number. DPP does

023C 856283 348 MOV DPH, DPH1 ; not hold any address lines in this mode

824CTEST PAGE 7

023F 856382 349 MOV DPL, DPL1

0242 E0 350 MOVX A, @DPTR

0243 120D92 351 CALL SENDVAL

352

0246 B56406 353 CJNE A, DATAOUT, ERROR2A

0249 120D48 354 CALL PRINTOK

024C 020252 355 JMP $+6

356

024F 357 ERROR2A:

024F 120D4F 358 CALL PRINTERROR

359

0252 856283 360 MOV DPH, DPH1

0255 856382 361 MOV DPL, DPL1

0258 08 362 INC R0

0259 E583 363 MOV A, DPH

025B 2420 364 ADD A, #20h

025D F583 365 MOV DPH, A

366

025F B858A4 367 CJNE R0, #58H, XRAMIN1

368 ;\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

369

0262 90096B 370 LK9B: MOV DPTR,#LINK9B ; "LK9 to position B"

0265 120D62 371 CALL SENDSTRING

0268 D200 372 SETB FLAG

026A 2000FD 373 JB FLAG, $ ; wait for INT0

374

026D 758405 375 MOV DPP,#05h

0270 9000FF 376 MOV DPTR, #00FFH

0273 7857 377 MOV R0, #57h ; R0 starts at 57h

378

379 ; save data in RAM into the following locations

0275 380 XRAMIN2:

0275 E6 381 MOV A, @R0

0276 8664 382 MOV DATAOUT, @R0

0278 F0 383 MOVX @DPTR, A

0279 858362 384 MOV DPH1, DPH ; DPH1 --> DPL1 hold the present

027C 858263 385 MOV DPL1, DPL ; address in XRAM

386

387

027F 900D2C 388 MOV DPTR, #ADDRMSG

0282 120D62 389 CALL SENDSTRING

390

0285 E584 391 MOV A, DPP

0287 120D92 392 CALL SENDVAL

028A E562 393 MOV A, DPH1

028C 120D92 394 CALL SENDVAL

028F E563 395 MOV A, DPL1

0291 120D92 396 CALL SENDVAL

397

0294 900D34 398 MOV DPTR, #WRITTENMSG

0297 120D62 399 CALL SENDSTRING

400

029A E564 401 MOV A, DATAOUT

029C 120D92 402 CALL SENDVAL

403

029F 900D40 404 MOV DPTR, #READMSG

02A2 120D62 405 CALL SENDSTRING

406

824CTEST PAGE 8

02A5 758355 407 MOV DPH, #55h ; corrupt DPH as DPP contains the

02A8 856382 408 MOV DPL, DPL1 ; high address lines in this case

02AB E0 409 MOVX A, @DPTR

02AC 120D92 410 CALL SENDVAL

411

02AF B56406 412 CJNE A, DATAOUT, ERROR2B

02B2 120D48 413 CALL PRINTOK

02B5 0202BB 414 JMP $+6

02B8 415 ERROR2B:

02B8 120D4F 416 CALL PRINTERROR

417

02BB 856283 418 MOV DPH, DPH1

02BE 856382 419 MOV DPL, DPL1

02C1 18 420 DEC R0

02C2 E584 421 MOV A, DPP

02C4 2420 422 ADD A, #20h

02C6 F584 423 MOV DPP, A

424

02C8 B84FAA 425 CJNE R0, #4FH, XRAMIN2

02CB 0193 426 JMP MAINMENU

427

428

429 ;====================================================================

430 ; TEST 3: SPI Interface TEST

431 ;====================================================================

432

02CD 433 TEST3:

02CD 9009A0 434 MOV DPTR,#SPITITLE ; display SPI title

02D0 120D62 435 CALL SENDSTRING

436

02D3 D200 437 SETB FLAG

02D5 2000FD 438 JB FLAG, $ ; wait for INT0

439

02D8 75F833 440 MOV SPICON,#33h ; configure SPI port Master Mode

441 ; slowest bit rate

442

443 ; Test SCLOCK

02DB 75F700 444 MOV SPIDAT, #00h ; transmit 00h

02DE 445 WAITFORLOW:

02DE 20B30B 446 JB P3.3, SCLOCKHIGH; wait for SCLOCK to go low

02E1 447 WAITFORHIGH:

02E1 30B30D 448 JNB P3.3, SCLOCKLOW ; wait for SCLOCK to go high

02E4 30FFFD 449 JNB ISPI, $

02E7 C2FF 450 CLR ISPI

02E9 0202F6 451 JMP TESTSDATA

02EC 452 SCLOCKHIGH:

02EC 20FF50 453 JB ISPI, SPIERROR

02EF 80ED 454 JMP WAITFORLOW

02F1 455 SCLOCKLOW:

02F1 20FF4B 456 JB ISPI, SPIERROR

02F4 80EB 457 JMP WAITFORHIGH

458

459

460 ; Test SDATA

02F6 461 TESTSDATA:

02F6 900A55 462 MOV DPTR, #SDATAMSG

02F9 120D62 463 CALL SENDSTRING

02FC D200 464 SETB FLAG

824CTEST PAGE 9

02FE 2000FD 465 JB FLAG, $ ; wait for INT0

466

467 ; ; reinitialise the SPIPORT

468 ; MOV SPICON, #0

469 ; MOV SPICON,#33h ; configure SPI port Master Mode

470 ; slowest bit rate

0301 75F70F 471 MOV SPIDAT, #0Fh ; transmit 0Fh

0304 472 WAITFORLOW1:

0304 20B30B 473 JB P3.3, SDATAHIGH; wait for SCLOCK to go low

0307 474 WAITFORHIGH1:

0307 30B30D 475 JNB P3.3, SDATALOW ; wait for SCLOCK to go high

030A 30FFFD 476 JNB ISPI, $

030D C2FF 477 CLR ISPI

030F 02031C 478 JMP TESTSS

0312 479 SDATAHIGH:

0312 20FF2A 480 JB ISPI, SPIERROR

0315 80ED 481 JMP WAITFORLOW1

0317 482 SDATALOW:

0317 20FF25 483 JB ISPI, SPIERROR

031A 80EB 484 JMP WAITFORHIGH1

485

031C 486 TESTSS:

487 ; connect p3.3 to SS

031C 900A7D 488 MOV DPTR, #SSMSG

031F 120D62 489 CALL SENDSTRING

0322 D200 490 SETB FLAG

0324 2000FD 491 JB FLAG, $ ; wait for INT0

492

0327 75B0FF 493 MOV P3, #0FFh ; set all P3 as inputs

032A D2B5 494 SETB P3.5

032C 30B310 495 JNB P3.3, SPIERROR

032F C2B5 496 CLR P3.5

0331 20B30B 497 JB P3.3, SPIERROR

498

0334 499 SPIOK:

0334 900AA2 500 MOV DPTR, #SPIMSG

0337 120D62 501 CALL SENDSTRING

033A 120D48 502 CALL PRINTOK

033D 0193 503 JMP MAINMENU

504

033F 505 SPIERROR:

033F 900AA2 506 MOV DPTR, #SPIMSG

0342 120D62 507 CALL SENDSTRING

0345 120D4F 508 CALL PRINTERROR

0348 0193 509 JMP MAINMENU

510

511

512 ;====================================================================

513 ; TEST 4: RTD Demo Circuit TEST

514 ;====================================================================

515

034A 516 TEST4:

034A 900AC1 517 MOV DPTR,#RTDTITLE ; display RTD title

034D 120D62 518 CALL SENDSTRING

519

0350 900B28 520 MOV DPTR,#RTDLINKS ; display RTD links

0353 120D62 521 CALL SENDSTRING

522

824CTEST PAGE 10

523 ; Configure Current source

0356 75D501 524 MOV ICON, #01h ; put 200uA out on pin 3

525

526

0359 D200 527 SETB FLAG

035B 2000FD 528 JB FLAG, $ ; wait for INT0

529

530

531 ; Configure ADC

035E 75D24A 532 MOV ADC0CON, #4AH ; use external ref (R11)

533 ; unpolar mode

534 ; fixed 0->80mV range

535

0361 75D122 536 MOV ADCMODE, #22H ; initiate a single prim chan conv

0364 30DFFD 537 JNB RDY0,$ ; Wait for conversion results

538

0367 900D00 539 MOV DPTR, #RTDSENSOR ; Send temp up UART

036A 120D62 540 CALL SENDSTRING

036D E5DB 541 MOV A, ADC0H

036F 120D92 542 CALL SENDVAL

0372 E5DA 543 MOV A, ADC0M

0374 120D92 544 CALL SENDVAL

545

546 ; a value in AD0H of between 88h and A7h represents temps

547 ; between -11degC and 47degC. Only these temps should

548 ; pass the test.

0377 E5DB 549 MOV A, ADC0H

0379 C2D7 550 CLR CY

037B 9408 551 SUBB A, #08h ; A7->9Fh, 88->80h

037D 54F0 552 ANL A, #0F0h ; correct data is now either 80h or 90h

037F B49005 553 CJNE A, #90h, CHECK80 ; if fails 90 check A0

0382 120D48 554 CALL PRINTOK

0385 0193 555 JMP MAINMENU

0387 556 CHECK80:

0387 B48005 557 CJNE A, #80h, ERROR4 ; fails 80 and 90 => fail

038A 120D48 558 CALL PRINTOK

038D 0193 559 JMP MAINMENU

038F 120D4F 560 ERROR4: CALL PRINTERROR

0392 0193 561 JMP MAINMENU

562 ;====================================================================

563 ; TEST 5: Automatic Checks

564 ;====================================================================

0394 565 TEST5:

0394 900BD5 566 MOV DPTR, #AUTO

0397 120D62 567 CALL SENDSTRING

568

569 ;test port 1 for shorts

039A 900C4D 570 MOV DPTR, #PORT1

039D 120D62 571 CALL SENDSTRING

03A0 D291 572 SETB P1.1

03A2 C290 573 CLR P1.0

03A4 E590 574 MOV A, P1

03A6 5403 575 ANL A, #03h ; zero acc.2 -> acc.7

03A8 B40205 576 CJNE A, #02h, $+8 ; p1.1=1, p1.0=0 =>OK

03AB 120D48 577 CALL PRINTOK

03AE 61B3 578 AJMP $+5

03B0 120D4F 579 CALL PRINTERROR

580

824CTEST PAGE 11

581 ;test port 3 for shorts

03B3 900C30 582 MOV DPTR, #PORT3

03B6 120D62 583 CALL SENDSTRING

03B9 D2B2 584 SETB P3.2 ; do not alter TXD and RXD lines

03BB C2B3 585 CLR P3.3

03BD D2B4 586 SETB P3.4

03BF C2B5 587 CLR P3.5

03C1 D2B6 588 SETB P3.6

03C3 C2B7 589 CLR P3.7

03C5 E5B0 590 MOV A, P3

03C7 54FC 591 ANL A, #0FCh ; zero acc.0 and acc.1

03C9 B45405 592 CJNE A, #54h, $+8

03CC 120D48 593 CALL PRINTOK

03CF 61D4 594 AJMP $+5

03D1 120D4F 595 CALL PRINTERROR

596

597 ;use TIC to test if crystal is oscillating

03D4 900C74 598 MOV DPTR, #CRYSTAL

03D7 120D62 599 CALL SENDSTRING

600

03DA 75A60B 601 MOV INTVAL, #0BH ; SET INTVAL FOR 12/128TH =93ms

03DD 75A103 602 MOV TIMECON, #03H ; MEASURE IN 1/128THS SECS

03E0 7401 603 MOV A, #01H

03E2 120D56 604 CALL DELAY ;SOFTWARE 100ms delay => TII set

03E5 E5A1 605 MOV A, TIMECON

03E7 20E205 606 JB ACC.2, $+8

03EA 120D4F 607 CALL PRINTERROR

03ED 61F2 608 AJMP $+5

03EF 120D48 609 CALL PRINTOK

03F2 75A100 610 MOV TIMECON, #00h ; this clears TII and disables TIC

611

612 ;any other automatic tests go here

613

614

03F5 90062F 615 MOV DPTR, #REMOVELINKS1

03F8 120D62 616 CALL SENDSTRING

03FB 9006C2 617 MOV DPTR, #REMOVELINKS2

03FE 120D62 618 CALL SENDSTRING

0401 0193 619 JMP MAINMENU

620

621 ;\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

622

0403 0A0A0A0D 623 TITLE: DB 10,10,10,13,'\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_',10,13

0407 5F5F5F5F

040B 5F5F5F5F

040F 5F5F5F5F

0413 5F5F5F5F

0417 5F5F5F5F

041B 5F5F5F5F

041F 5F5F5F5F

0423 5F5F5F5F

0427 5F5F5F5F

042B 5F5F5F0A

042F 0D

0430 20416E61 624 DB ' Analog Devices MicroConverter ADuC824',10,13

0434 6C6F6720

0438 44657669

043C 63657320

824CTEST PAGE 12

0440 4D696372

0444 6F436F6E

0448 76657274

044C 65722041

0450 44754338

0454 32340A0D

0458 20202045 625 DB ' Evaluation Board Test Routine',0

045C 76616C75

0460 6174696F

0464 6E20426F

0468 61726420

046C 54657374

0470 20526F75

0474 74696E65

0478 00

0479 626 Menu:

0479 0A0A0D4D 627 DB 10,10,13,'Main Menu:', 10, 13

047D 61696E20

0481 4D656E75

0485 3A0A0D

0488 20203120 628 DB ' 1 - Test ADCs and DAC',10,13

048C 2D205465

0490 73742041

0494 44437320

0498 616E6420

049C 4441430A

04A0 0D

04A1 20203220 629 DB ' 2 - Test External Data Memory' ,10,13

04A5 2D205465

04A9 73742045

04AD 78746572

04B1 6E616C20

04B5 44617461

04B9 204D656D

04BD 6F72790A

04C1 0D

04C2 20203320 630 DB ' 3 - Test SPI Interface', 10,13

04C6 2D205465

04CA 73742053

04CE 50492049

04D2 6E746572

04D6 66616365

04DA 0A0D

04DC 20203420 631 DB ' 4 - Test RTD Demo Circuit',10,13

04E0 2D205465

04E4 73742052

04E8 54442044

04EC 656D6F20

04F0 43697263

04F4 7569740A

04F8 0D

04F9 20203520 632 DB ' 5 - Automatic Checks',10,13

04FD 2D204175

0501 746F6D61

0505 74696320

0509 43686563

050D 6B730A0D

0511 53656C65 633 DB 'Select: ',0

824CTEST PAGE 13

0515 63743A20

0519 00

634

051A 0A0A0D45 635 LINKS1: DB 10,10,13,'Ensure that ONLY the following links are inserted'

051E 6E737572

0522 65207468

0526 6174204F

052A 4E4C5920

052E 74686520

0532 666F6C6C

0536 6F77696E

053A 67206C69

053E 6E6B7320

0542 61726520

0546 696E7365

054A 72746564

054E 0A0D202D 636 DB 10,13,' - Slide LK1 to ON Posn'

0552 20536C69

0556 6465204C

055A 4B312074

055E 6F204F4E

0562 20506F73

0566 6E

0567 0A0D202D 637 DB 10,13,' - Slide LK5 to ON Posn'

056B 20536C69

056F 6465204C

0573 4B352074

0577 6F204F4E

057B 20506F73

057F 6E

0580 0A0D202D 638 DB 10,13,' - Slide LK6 to Posn A'

0584 20536C69

0588 6465204C

058C 4B362074

0590 6F20506F

0594 736E2041

0598 0A0D202D 639 DB 10,13,' - Slide LK7 to Posn A',0

059C 20536C69

05A0 6465204C

05A4 4B372074

05A8 6F20506F

05AC 736E2041

05B0 00

05B1 0A0D202D 640 LINKS2: DB 10,13,' - Slide LK8 to Posn A'

05B5 20536C69

05B9 6465204C

05BD 4B382074

05C1 6F20506F

05C5 736E2041

05C9 0A0D202D 641 DB 10,13,' - Slide LK9 to Posn A'

05CD 20536C69

05D1 6465204C

05D5 4B392074

05D9 6F20506F

05DD 736E2041

05E1 0A0D202D 642 DB 10,13,' - Slide LK10 to Posn A'

05E5 20536C69

05E9 6465204C

824CTEST PAGE 14

05ED 4B313020

05F1 746F2050

05F5 6F736E20

05F9 41

05FA 0A0D202D 643 DB 10,13,' - Slide LK11 to ON Posn'

05FE 20536C69

0602 6465204C

0606 4B313120

060A 746F204F

060E 4E20506F

0612 736E

0614 0A0D202D 644 DB 10,13,' - Slide LK12 to ON Posn',0

0618 20536C69

061C 6465204C

0620 4B313220

0624 746F204F

0628 4E20506F

062C 736E00

645

062F 646 REMOVELINKS1:

062F 0A0A0D49 647 DB 10,10,13,'If finished INSERT only the following Links'

0633 66206669

0637 6E697368

063B 65642049

063F 4E534552

0643 54206F6E

0647 6C792074

064B 68652066

064F 6F6C6C6F

0653 77696E67

0657 204C696E

065B 6B73

065D 0A0D202D 648 DB 10,13,' - Slide LK1 to ON Posn'

0661 20536C69

0665 6465204C

0669 4B312074

066D 6F204F4E

0671 20506F73

0675 6E

0676 0A0D202D 649 DB 10,13,' - Slide LK3 to ON Posn'

067A 20536C69

067E 6465204C

0682 4B332074

0686 6F204F4E

068A 20506F73

068E 6E

068F 0A0D202D 650 DB 10,13,' - Slide LK5 to ON Posn'

0693 20536C69

0697 6465204C

069B 4B352074

069F 6F204F4E

06A3 20506F73

06A7 6E

06A8 0A0D202D 651 DB 10,13,' - Slide LK6 to ON Posn',0

06AC 20536C69

06B0 6465204C

06B4 4B362074

06B8 6F204F4E

824CTEST PAGE 15

06BC 20506F73

06C0 6E00

06C2 652 REMOVELINKS2:

06C2 0A0D202D 653 DB 10,13,' - Slide LK7 to Posn A'

06C6 20536C69

06CA 6465204C

06CE 4B372074

06D2 6F20506F

06D6 736E2041

06DA 0A0D202D 654 DB 10,13,' - Slide LK8 to Posn A'

06DE 20536C69

06E2 6465204C

06E6 4B382074

06EA 6F20506F

06EE 736E2041

06F2 0A0D202D 655 DB 10,13,' - Slide LK9 to Posn A'

06F6 20536C69

06FA 6465204C

06FE 4B392074

0702 6F20506F

0706 736E2041

070A 0A0D416C 656 DB 10,13,'All other links should be in the OFF posn',10,13,0

070E 6C206F74

0712 68657220

0716 6C696E6B

071A 73207368

071E 6F756C64

0722 20626520

0726 696E2074

072A 6865204F

072E 46462070

0732 6F736E0A

0736 0D00

657

658

0738 0A0A0D54 659 PRIMADC: DB 10,10,13,'TEST 1: Testing ADCs and DAC in a Feedback Configuration'

073C 45535420

0740 313A2054

0744 65737469

0748 6E672041

074C 44437320

0750 616E6420

0754 44414320

0758 696E2061

075C 20466565

0760 64626163

0764 6B20436F

0768 6E666967

076C 75726174

0770 696F6E

0773 0A0D2D2D 660 DB 10,13,'-------------------------------------------------------',10,13

0777 2D2D2D2D

077B 2D2D2D2D

077F 2D2D2D2D

0783 2D2D2D2D

0787 2D2D2D2D

078B 2D2D2D2D

078F 2D2D2D2D

824CTEST PAGE 16

0793 2D2D2D2D

0797 2D2D2D2D

079B 2D2D2D2D

079F 2D2D2D2D

07A3 2D2D2D2D

07A7 2D2D2D2D

07AB 2D0A0D

07AE 436F6E6E 661 DB 'Connect the DAC (J2-13) to the Primary ADC (Ain1 J2-9).',10,13

07B2 65637420

07B6 74686520

07BA 44414320

07BE 284A322D

07C2 31332920

07C6 746F2074

07CA 68652050

07CE 72696D61

07D2 72792041

07D6 44432028

07DA 41696E31

07DE 204A322D

07E2 39292E0A

07E6 0D

07E7 50726573 662 DB 'Press the INT0 button when ready',10,13,0

07EB 73207468

07EF 6520494E

07F3 54302062

07F7 7574746F

07FB 6E207768

07FF 656E2072

0803 65616479

0807 0A0D00

663

080A 0A0A0D43 664 AUX3ADC: DB 10,10,13,'Connect the DAC (J2-13) to the Auxilary ADC (Ain3 J2-11)',10,13

080E 6F6E6E65

0812 63742074

0816 68652044

081A 41432028

081E 4A322D31

0822 33292074

0826 6F207468

082A 65204175

082E 78696C61

0832 72792041

0836 44432028

083A 41696E33

083E 204A322D

0842 3131290A

0846 0D

0847 50726573 665 DB 'Press the INT0 button when ready',10,13,0

084B 73207468

084F 6520494E

0853 54302062

0857 7574746F

085B 6E207768

085F 656E2072

0863 65616479

0867 0A0D00

666

824CTEST PAGE 17

086A 0A0A0D43 667 AUX5ADC: DB 10,10,13,'Connect the DAC (J2-13) to the Auxilary ADC (Ain5 J2-6)',10,13

086E 6F6E6E65

0872 63742074

0876 68652044

087A 41432028

087E 4A322D31

0882 33292074

0886 6F207468

088A 65204175

088E 78696C61

0892 72792041

0896 44432028

089A 41696E35

089E 204A322D

08A2 36290A0D

08A6 50726573 668 DB 'Press the INT0 button when ready',10,13,0

08AA 73207468

08AE 6520494E

08B2 54302062

08B6 7574746F

08BA 6E207768

08BE 656E2072

08C2 65616479

08C6 0A0D00

669

08C9 0A0A0D20 670 MEMTITLE: DB 10,10,13,' TEST 2: Testing the External Data Memory'

08CD 20202020

08D1 20544553

08D5 5420323A

08D9 20205465

08DD 7374696E

08E1 67207468

08E5 65204578

08E9 7465726E

08ED 616C2044

08F1 61746120

08F5 4D656D6F

08F9 7279

08FB 0A0D2D2D 671 DB 10,13,'------------------------------------------------------',10,13,0

08FF 2D2D2D2D

0903 2D2D2D2D

0907 2D2D2D2D

090B 2D2D2D2D

090F 2D2D2D2D

0913 2D2D2D2D

0917 2D2D2D2D

091B 2D2D2D2D

091F 2D2D2D2D

0923 2D2D2D2D

0927 2D2D2D2D

092B 2D2D2D2D

092F 2D2D2D2D

0933 0A0D00

0936 0A0D536C 672 LINK9A: DB 10,13,'Slide LK9 to position A. Press INT0 when ready.',10,13,0

093A 69646520

093E 4C4B3920

0942 746F2070

0946 6F736974

824CTEST PAGE 18

094A 696F6E20

094E 412E2020

0952 50726573

0956 7320494E

095A 54302077

095E 68656E20

0962 72656164

0966 792E0A0D

096A 00

096B 0A0D536C 673 LINK9B: DB 10,13,'Slide LK9 to position B. Press INT0 when ready.',10,13,0

096F 69646520

0973 4C4B3920

0977 746F2070

097B 6F736974

097F 696F6E20

0983 422E2020

0987 50726573

098B 7320494E

098F 54302077

0993 68656E20

0997 72656164

099B 792E0A0D

099F 00

674

09A0 0A0A0D20 675 SPITITLE: DB 10,10,13,' TEST 3: Testing the SPI/I2C Interface'

09A4 20202020

09A8 20205445

09AC 53542033

09B0 3A205465

09B4 7374696E

09B8 67207468

09BC 65205350

09C0 492F4932

09C4 4320496E

09C8 74657266

09CC 616365

09CF 0A0D2D2D 676 DB 10,13,'------------------------------------------------------',10,13

09D3 2D2D2D2D

09D7 2D2D2D2D

09DB 2D2D2D2D

09DF 2D2D2D2D

09E3 2D2D2D2D

09E7 2D2D2D2D

09EB 2D2D2D2D

09EF 2D2D2D2D

09F3 2D2D2D2D

09F7 2D2D2D2D

09FB 2D2D2D2D

09FF 2D2D2D2D

0A03 2D2D2D2D

0A07 0A0D

0A09 0A0D436F 677 DB 10,13,'Connect SCLOCK (J7-1) to P3.3 (J3-4).'

0A0D 6E6E6563

0A11 74205343

0A15 4C4F434B

0A19 20284A37

0A1D 2D312920

0A21 746F2050

824CTEST PAGE 19

0A25 332E3320

0A29 284A332D

0A2D 34292E

0A30 0A0D5072 678 DB 10,13,'Press the INT0 button when ready',10,13,0

0A34 65737320

0A38 74686520

0A3C 494E5430

0A40 20627574

0A44 746F6E20

0A48 7768656E

0A4C 20726561

0A50 64790A0D

0A54 00

0A55 0A0D436F 679 SDATAMSG: DB 10,13,'Connect SDATA (J7-3) to P3.3 (J3-4)',10,13,0

0A59 6E6E6563

0A5D 74205344

0A61 41544120

0A65 284A372D

0A69 33292074

0A6D 6F205033

0A71 2E332028

0A75 4A332D34

0A79 290A0D00

0A7D 0A0D436F 680 SSMSG: DB 10,13,'Connect SS (J7-7) to P3.3 (J3-4)',10,13,0

0A81 6E6E6563

0A85 74205353

0A89 20284A37

0A8D 2D372920

0A91 746F2050

0A95 332E3320

0A99 284A332D

0A9D 34290A0D

0AA1 00

0AA2 0A0D5465 681 SPIMSG: DB 10,13,'Testing SPI/I2C Interface...',0

0AA6 7374696E

0AAA 67205350

0AAE 492F4932

0AB2 4320496E

0AB6 74657266

0ABA 6163652E

0ABE 2E2E00

682

0AC1 0A0A0D20 683 RTDTITLE: DB 10,10,13,' TEST4: Testing the RTD Demo Circuit'

0AC5 20202020

0AC9 20202054

0ACD 45535434

0AD1 3A205465

0AD5 7374696E

0AD9 67207468

0ADD 65205254

0AE1 44204465

0AE5 6D6F2043

0AE9 69726375

0AED 6974

0AEF 0A0D2D2D 684 DB 10,13,'------------------------------------------------------',0

0AF3 2D2D2D2D

0AF7 2D2D2D2D

0AFB 2D2D2D2D

824CTEST PAGE 20

0AFF 2D2D2D2D

0B03 2D2D2D2D

0B07 2D2D2D2D

0B0B 2D2D2D2D

0B0F 2D2D2D2D

0B13 2D2D2D2D

0B17 2D2D2D2D

0B1B 2D2D2D2D

0B1F 2D2D2D2D

0B23 2D2D2D2D

0B27 00

0B28 685 RTDLINKS:

0B28 0A0D4368 686 DB 10,13,'Change the following Links:'

0B2C 616E6765

0B30 20746865

0B34 20666F6C

0B38 6C6F7769

0B3C 6E67204C

0B40 696E6B73

0B44 3A

0B45 0A0D202D 687 DB 10,13,' - Move LK7 into position B'

0B49 204D6F76

0B4D 65204C4B

0B51 3720696E

0B55 746F2070

0B59 6F736974

0B5D 696F6E20

0B61 42

0B62 0A0D202D 688 DB 10,13,' - Move LK8 into position B'

0B66 204D6F76

0B6A 65204C4B

0B6E 3820696E

0B72 746F2070

0B76 6F736974

0B7A 696F6E20

0B7E 42

0B7F 0A0D202D 689 DB 10,13,' - REMOVE LK5'

0B83 2052454D

0B87 4F564520

0B8B 4C4B35

0B8E 0A0D4E4F 690 DB 10,13,'NOTE: Remove any connector used in test 1'

0B92 54453A20

0B96 52656D6F

0B9A 76652061

0B9E 6E792063

0BA2 6F6E6E65

0BA6 63746F72

0BAA 20757365

0BAE 6420696E

0BB2 20746573

0BB6 742031

0BB9 0A0A0D50 691 DB 10,10,13,'Press INT0 when ready.',10,13,0

0BBD 72657373

0BC1 20494E54

0BC5 30207768

0BC9 656E2072

0BCD 65616479

0BD1 2E0A0D00

824CTEST PAGE 21

692

0BD5 0A0A0D20 693 AUTO: DB 10,10,13,' TEST5: Automatic Checks'

0BD9 20202020

0BDD 20202054

0BE1 45535435

0BE5 3A204175

0BE9 746F6D61

0BED 74696320

0BF1 43686563

0BF5 6B73

0BF7 0A0D2D2D 694 DB 10,13,'------------------------------------------------------',0

0BFB 2D2D2D2D

0BFF 2D2D2D2D

0C03 2D2D2D2D

0C07 2D2D2D2D

0C0B 2D2D2D2D

0C0F 2D2D2D2D

0C13 2D2D2D2D

0C17 2D2D2D2D

0C1B 2D2D2D2D

0C1F 2D2D2D2D

0C23 2D2D2D2D

0C27 2D2D2D2D

0C2B 2D2D2D2D

0C2F 00

0C30 0A0D5465 695 PORT3: DB 10,13,'Testing Port 3 for shorts ', 0

0C34 7374696E

0C38 6720506F

0C3C 72742033

0C40 20666F72

0C44 2073686F

0C48 72747320

0C4C 00

0C4D 0A0D5465 696 PORT1: DB 10,13,'Testing Port1.0 and P1.1 for shorts ', 0

0C51 7374696E

0C55 6720506F

0C59 7274312E

0C5D 3020616E

0C61 64205031

0C65 2E312066

0C69 6F722073

0C6D 686F7274

0C71 732000

0C74 0A0D5465 697 CRYSTAL: DB 10,13,'Testing crystal ciruit ',0

0C78 7374696E

0C7C 67206372

0C80 79737461

0C84 6C206369

0C88 72756974

0C8C 2000

698

0C8E 0A0A0A0D 699 GOMENU: DB 10,10,10,13,'Press INT0 to go to the Main Menu',0

0C92 50726573

0C96 7320494E

0C9A 54302074

0C9E 6F20676F

0CA2 20746F20

0CA6 74686520

824CTEST PAGE 22

0CAA 4D61696E

0CAE 204D656E

0CB2 7500

0CB4 0A0D5072 700 RETMENU: DB 10,13,'Press INT0 to return to the Main Menu',0

0CB8 65737320

0CBC 494E5430

0CC0 20746F20

0CC4 72657475

0CC8 726E2074

0CCC 6F207468

0CD0 65204D61

0CD4 696E204D

0CD8 656E7500

701

0CDC 20444143 702 DACMSG: DB ' DAC=',0

0CE0 3D00

0CE2 202D3E20 703 ADC0MSG: DB ' -> Primary ADC=',0

0CE6 5072696D

0CEA 61727920

0CEE 4144433D

0CF2 00

0CF3 202D3E20 704 ADC1MSG: DB ' -> Aux ADC=',0

0CF7 41757820

0CFB 4144433D

0CFF 00

0D00 2054656D 705 RTDSENSOR: DB ' Temperature Conversion = ',0

0D04 70657261

0D08 74757265

0D0C 20436F6E

0D10 76657273

0D14 696F6E20

0D18 3D2000

706

0D1B 204F4B0A 707 OKMSG: DB ' OK',10,13,0

0D1F 0D00

0D21 20455252 708 ERRORMSG: DB ' ERROR!!',10,13,0

0D25 4F522121

0D29 0A0D00

709

0D2C 0D204144 710 ADDRMSG: DB 13,' ADDR=',0

0D30 44523D00

0D34 3A202057 711 WRITTENMSG: DB ': WRITTEN=',0

0D38 52495454

0D3C 454E3D00

0D40 2C205245 712 READMSG: DB ', READ=',0

0D44 41443D00

713

714

715

716

717

718

719

720 ;====================================================================

721 ; FUNCTIONS

722 ;====================================================================

723

724

824CTEST PAGE 23

725 ;\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

726 ; PRINT OK AND ERROR MESSAGES

0D48 727 PRINTOK:

0D48 900D1B 728 MOV DPTR,#OKMSG

0D4B 120D62 729 CALL SENDSTRING

0D4E 22 730 RET

0D4F 731 PRINTERROR:

0D4F 900D21 732 MOV DPTR,#ERRORMSG

0D52 120D62 733 CALL SENDSTRING

0D55 22 734 RET

735 ;\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

736 ; DELAY

0D56 737 DELAY: ; Delays by 100ms \* A

738 ; 100mSec based on 1.573MHZ Core Clock

739

0D56 FA 740 MOV R2,A ; Acc holds delay variable

0D57 7B32 741 DLY0: MOV R3,#50 ; Set up delay loop0

0D59 7C83 742 DLY1: MOV R4,#131 ; Set up delay loop1

0D5B DCFE 743 DJNZ R4, $ ; Dec R4 & Jump here until R4 is 0

744 ; wait here for 131\*15.3us=2ms

0D5D DBFA 745 DJNZ R3, DLY1 ; Dec R3 & Jump DLY1 until R3 is 0

746 ; Wait for 50\*2ms

0D5F DAF6 747 DJNZ R2,DLY0 ; Dec R2 & Jump DLY0 until R2 is 0

748 ; wait for ACC\*100ms

0D61 22 749 RET ; Return from subroutine

750

751 ;\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

0D62 752 SENDSTRING: ; sends ASCII string to UART starting at location DPTR

753 ; and ending with a null (0) value

754

0D62 C0E0 755 PUSH ACC

0D64 C0F0 756 PUSH B

0D66 E4 757 CLR A

0D67 F5F0 758 MOV B,A

0D69 E5F0 759 SENDNEXT: MOV A,B

0D6B 05F0 760 INC B

0D6D 93 761 MOVC A,@A+DPTR

0D6E 6009 762 JZ SENDDONE

0D70 3099FD 763 JNB TI,$ ; wait til present char gone

0D73 C299 764 CLR TI ; must clear TI

0D75 F599 765 MOV SBUF,A ; transmit byte

0D77 80F0 766 JMP SENDNEXT

0D79 D0F0 767 SENDDONE: POP B

0D7B D0E0 768 POP ACC

769

0D7D 22 770 RET

771

772 ;\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

0D7E 773 SENDCHAR: ; sends ASCII value contained in A to UART

774

0D7E 3099FD 775 JNB TI,$ ; wait til present char gone

0D81 C299 776 CLR TI ; must clear TI

0D83 F599 777 MOV SBUF,A

778

0D85 22 779 RET

780

781 ;\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

0D86 782 HEX2ASCII: ; converts A into the hex character representing the

824CTEST PAGE 24

783 ; previous value of A's least significant nibble

784

0D86 540F 785 ANL A,#00Fh

0D88 B40A00 786 CJNE A,#00Ah,$+3

0D8B 4002 787 JC CONV

0D8D 2407 788 ADD A,#007h

0D8F 2430 789 CONV: ADD A,#'0'

790

0D91 22 791 RET

792

793 ;\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

0D92 794 SENDVAL: ; uses the above two subroutines to....

795 ; 1) convert the hex value of A into two ASCII chars

796 ; 2) spew converted chars out the UART

797

0D92 C0E0 798 PUSH ACC

0D94 C4 799 SWAP A

0D95 540F 800 ANL A,#00Fh

0D97 B186 801 CALL HEX2ASCII

0D99 B17E 802 CALL SENDCHAR

0D9B D0E0 803 POP ACC

0D9D C0E0 804 PUSH ACC

0D9F 540F 805 ANL A,#00Fh

0DA1 B186 806 CALL HEX2ASCII

0DA3 B17E 807 CALL SENDCHAR

0DA5 D0E0 808 POP ACC

809

0DA7 22 810 RET

811 ;\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

812 ; shifts the 12MSBs of R7,R6 to the 12LSBs of R7,R6.

813 ; A zero is read into the 4 MSBs of R7

0DA8 814 SHIFT:

0DA8 C0E0 815 PUSH ACC

0DAA EE 816 MOV A,R6

0DAB 54F0 817 ANL A,#0F0h

0DAD C4 818 SWAP A

0DAE FE 819 MOV R6,A

0DAF EF 820 MOV A,R7

0DB0 C4 821 SWAP A

0DB1 54F0 822 ANL A,#0F0h

0DB3 4E 823 ORL A,R6

0DB4 FE 824 MOV R6,A

0DB5 EF 825 MOV A,R7

0DB6 C4 826 SWAP A

0DB7 540F 827 ANL A,#0Fh

0DB9 FF 828 MOV R7,A

0DBA D0E0 829 POP ACC

830

0DBC 22 831 RET

832 ;\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

833 ; sends the 12 LSBs of R7,R6 up the UART in ASCII form

834

0DBD 835 SEND12BITS:

0DBD C0E0 836 PUSH ACC

0DBF EF 837 MOV A,R7

0DC0 540F 838 ANL A,#0Fh

0DC2 B186 839 CALL HEX2ASCII

0DC4 B17E 840 CALL SENDCHAR

824CTEST PAGE 25

0DC6 EE 841 MOV A,R6

0DC7 C4 842 SWAP A

0DC8 540F 843 ANL A,#0Fh

0DCA B186 844 CALL HEX2ASCII

0DCC B17E 845 CALL SENDCHAR

0DCE EE 846 MOV A,R6

0DCF 540F 847 ANL A,#0Fh

0DD1 B186 848 CALL HEX2ASCII

0DD3 B17E 849 CALL SENDCHAR

0DD5 D0E0 850 POP ACC

0DD7 22 851 RET

852 ;\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

853

854 END

VERSION 1.2h ASSEMBLY COMPLETE, 0 ERRORS FOUND

824CTEST PAGE 26

ACC. . . . . . . . . . . . . . . D ADDR 00E0H PREDEFINED

ADC0CON. . . . . . . . . . . . . D ADDR 00D2H PREDEFINED

ADC0H. . . . . . . . . . . . . . D ADDR 00DBH PREDEFINED

ADC0L. . . . . . . . . . . . . . D ADDR 00D9H PREDEFINED

ADC0M. . . . . . . . . . . . . . D ADDR 00DAH PREDEFINED

ADC0MSG. . . . . . . . . . . . . C ADDR 0CE2H

ADC1CON. . . . . . . . . . . . . D ADDR 00D3H PREDEFINED

ADC1H. . . . . . . . . . . . . . D ADDR 00DDH PREDEFINED

ADC1L. . . . . . . . . . . . . . D ADDR 00DCH PREDEFINED

ADC1MSG. . . . . . . . . . . . . C ADDR 0CF3H

ADCMODE. . . . . . . . . . . . . D ADDR 00D1H PREDEFINED

ADDRMSG. . . . . . . . . . . . . C ADDR 0D2CH

AUTO . . . . . . . . . . . . . . C ADDR 0BD5H

AUX3ADC. . . . . . . . . . . . . C ADDR 080AH

AUX5ADC. . . . . . . . . . . . . C ADDR 086AH

B. . . . . . . . . . . . . . . . D ADDR 00F0H PREDEFINED

BLINK. . . . . . . . . . . . . . C ADDR 0089H

CHECK80. . . . . . . . . . . . . C ADDR 0387H

CHECKADC0. . . . . . . . . . . . C ADDR 01A8H

CHECKADC1. . . . . . . . . . . . C ADDR 01BEH

CONV . . . . . . . . . . . . . . C ADDR 0D8FH

COUNT. . . . . . . . . . . . . . D ADDR 0060H

CRYSTAL. . . . . . . . . . . . . C ADDR 0C74H

CY . . . . . . . . . . . . . . . B ADDR 00D7H PREDEFINED

DACCON . . . . . . . . . . . . . D ADDR 00FDH PREDEFINED

DACH . . . . . . . . . . . . . . D ADDR 00FCH PREDEFINED

DACL . . . . . . . . . . . . . . D ADDR 00FBH PREDEFINED

DACMSG . . . . . . . . . . . . . C ADDR 0CDCH

DACOUT . . . . . . . . . . . . . D ADDR 0061H

DATAOUT. . . . . . . . . . . . . D ADDR 0064H

DELAY. . . . . . . . . . . . . . C ADDR 0D56H

DLY0 . . . . . . . . . . . . . . C ADDR 0D57H

DLY1 . . . . . . . . . . . . . . C ADDR 0D59H

DPH. . . . . . . . . . . . . . . D ADDR 0083H PREDEFINED

DPH1 . . . . . . . . . . . . . . D ADDR 0062H

DPL. . . . . . . . . . . . . . . D ADDR 0082H PREDEFINED

DPL1 . . . . . . . . . . . . . . D ADDR 0063H

DPP. . . . . . . . . . . . . . . D ADDR 0084H PREDEFINED

EA . . . . . . . . . . . . . . . B ADDR 00AFH PREDEFINED

ERROR2A. . . . . . . . . . . . . C ADDR 024FH

ERROR2B. . . . . . . . . . . . . C ADDR 02B8H

ERROR4 . . . . . . . . . . . . . C ADDR 038FH

ERRORMSG . . . . . . . . . . . . C ADDR 0D21H

EX0. . . . . . . . . . . . . . . B ADDR 00A8H PREDEFINED

EXTDATA. . . . . . . . . . . . . C ADDR 01D4H NOT USED

FIRSTGO. . . . . . . . . . . . . B ADDR 0001H

FLAG . . . . . . . . . . . . . . B ADDR 0000H

GOMENU . . . . . . . . . . . . . C ADDR 0C8EH

HEX2ASCII. . . . . . . . . . . . C ADDR 0D86H

ICON . . . . . . . . . . . . . . D ADDR 00D5H PREDEFINED

INTVAL . . . . . . . . . . . . . D ADDR 00A6H PREDEFINED

ISPI . . . . . . . . . . . . . . B ADDR 00FFH PREDEFINED

IT0. . . . . . . . . . . . . . . B ADDR 0088H PREDEFINED

LED. . . . . . . . . . . . . . . NUMB 00B4H

LINK9A . . . . . . . . . . . . . C ADDR 0936H

LINK9B . . . . . . . . . . . . . C ADDR 096BH

LINKS1 . . . . . . . . . . . . . C ADDR 051AH

LINKS2 . . . . . . . . . . . . . C ADDR 05B1H

824CTEST PAGE 27

LK9A . . . . . . . . . . . . . . C ADDR 01F3H NOT USED

LK9B . . . . . . . . . . . . . . C ADDR 0262H NOT USED

MAIN . . . . . . . . . . . . . . C ADDR 0060H

MAINMENU . . . . . . . . . . . . C ADDR 0093H

MEMTITLE . . . . . . . . . . . . C ADDR 08C9H

MENU . . . . . . . . . . . . . . C ADDR 0479H

OKMSG. . . . . . . . . . . . . . C ADDR 0D1BH

P1 . . . . . . . . . . . . . . . D ADDR 0090H PREDEFINED

P3 . . . . . . . . . . . . . . . D ADDR 00B0H PREDEFINED

PORT1. . . . . . . . . . . . . . C ADDR 0C4DH

PORT3. . . . . . . . . . . . . . C ADDR 0C30H

PRIMADC. . . . . . . . . . . . . C ADDR 0738H

PRINTERROR . . . . . . . . . . . C ADDR 0D4FH

PRINTOK. . . . . . . . . . . . . C ADDR 0D48H

RCAP2H . . . . . . . . . . . . . D ADDR 00CBH PREDEFINED

RCAP2L . . . . . . . . . . . . . D ADDR 00CAH PREDEFINED

RDY0 . . . . . . . . . . . . . . B ADDR 00DFH PREDEFINED

RDY1 . . . . . . . . . . . . . . B ADDR 00DEH PREDEFINED

READMSG. . . . . . . . . . . . . C ADDR 0D40H

REMOVELINKS1 . . . . . . . . . . C ADDR 062FH

REMOVELINKS2 . . . . . . . . . . C ADDR 06C2H

RETMENU. . . . . . . . . . . . . C ADDR 0CB4H

RI . . . . . . . . . . . . . . . B ADDR 0098H PREDEFINED

RTDLINKS . . . . . . . . . . . . C ADDR 0B28H

RTDSENSOR. . . . . . . . . . . . C ADDR 0D00H

RTDTITLE . . . . . . . . . . . . C ADDR 0AC1H

SBUF . . . . . . . . . . . . . . D ADDR 0099H PREDEFINED

SCLOCKHIGH . . . . . . . . . . . C ADDR 02ECH

SCLOCKLOW. . . . . . . . . . . . C ADDR 02F1H

SCON . . . . . . . . . . . . . . D ADDR 0098H PREDEFINED

SDATAHIGH. . . . . . . . . . . . C ADDR 0312H

SDATALOW . . . . . . . . . . . . C ADDR 0317H

SDATAMSG . . . . . . . . . . . . C ADDR 0A55H

SEND12BITS . . . . . . . . . . . C ADDR 0DBDH

SENDCHAR . . . . . . . . . . . . C ADDR 0D7EH

SENDDONE . . . . . . . . . . . . C ADDR 0D79H

SENDNEXT . . . . . . . . . . . . C ADDR 0D69H

SENDSTRING . . . . . . . . . . . C ADDR 0D62H

SENDVAL. . . . . . . . . . . . . C ADDR 0D92H

SHIFT. . . . . . . . . . . . . . C ADDR 0DA8H NOT USED

SKIP1. . . . . . . . . . . . . . C ADDR 00A4H

SKIP2. . . . . . . . . . . . . . C ADDR 00B8H

SP . . . . . . . . . . . . . . . D ADDR 0081H PREDEFINED

SPICON . . . . . . . . . . . . . D ADDR 00F8H PREDEFINED

SPIDAT . . . . . . . . . . . . . D ADDR 00F7H PREDEFINED

SPIERROR . . . . . . . . . . . . C ADDR 033FH

SPIMSG . . . . . . . . . . . . . C ADDR 0AA2H

SPIOK. . . . . . . . . . . . . . C ADDR 0334H NOT USED

SPITITLE . . . . . . . . . . . . C ADDR 09A0H

SSMSG. . . . . . . . . . . . . . C ADDR 0A7DH

T2CON. . . . . . . . . . . . . . D ADDR 00C8H PREDEFINED

TEST1. . . . . . . . . . . . . . C ADDR 00E5H

TEST2. . . . . . . . . . . . . . C ADDR 01D4H

TEST3. . . . . . . . . . . . . . C ADDR 02CDH

TEST4. . . . . . . . . . . . . . C ADDR 034AH

TEST5. . . . . . . . . . . . . . C ADDR 0394H

TESTAUXCHAN. . . . . . . . . . . C ADDR 0172H

TESTPRIMADC. . . . . . . . . . . C ADDR 0135H

824CTEST PAGE 28

TESTSDATA. . . . . . . . . . . . C ADDR 02F6H

TESTSS . . . . . . . . . . . . . C ADDR 031CH

TH2. . . . . . . . . . . . . . . D ADDR 00CDH PREDEFINED

TI . . . . . . . . . . . . . . . B ADDR 0099H PREDEFINED

TIMECON. . . . . . . . . . . . . D ADDR 00A1H PREDEFINED

TITLE. . . . . . . . . . . . . . C ADDR 0403H

TL2. . . . . . . . . . . . . . . D ADDR 00CCH PREDEFINED

WAITFORHIGH. . . . . . . . . . . C ADDR 02E1H

WAITFORHIGH1 . . . . . . . . . . C ADDR 0307H

WAITFORLOW . . . . . . . . . . . C ADDR 02DEH

WAITFORLOW1. . . . . . . . . . . C ADDR 0304H

WRITTENMSG . . . . . . . . . . . C ADDR 0D34H

XRAMIN1. . . . . . . . . . . . . C ADDR 0206H

XRAMIN2. . . . . . . . . . . . . C ADDR 0275H