UART1 PAGE 1

1 ;====================================================================

2 ;

3 ; Author : ADI - Apps

4 ;

5 ; Date : October 2003

6 ;

7 ; File : UART.asm

8 ;

9 ; Hardware : ADuC842/ADuC843

10 ;

11 ; Description : This Program saves 16 numbers in order initially

12 ; starting with 0 into memory locations 40h to 50h.

13 ; When finished the values in these locations are

14 ; transmitted down the UART in ASCII form to the PC

15 ; where they can be viewed using the preconfigured

16 ; Hyperterminal program. (c:\ADuC\_Beta84x\9600com1.ht)

17 ;

18 ; After the transmission of the 16 bytes a 2 second

19 ; delay is called and the process is repeated, this

20 ; time starting with the saving of 10h to location

21 ; 40h.

22 ;

23 ;====================================================================

24 ;

25 $MOD842 ;Use 8052 predefined Symbols

26

00B4 27 LED EQU P3.4

28

29 ;\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

30 ; BEGINNING OF CODE

---- 31 CSEG

0000 32 ORG 0000H

33

0000 020060 34 JMP MAIN

35

0060 36 ORG 0060H ; Start code at address above interrupts

37

38

0060 39 MAIN: ; Main program

40

0060 75D703 41 MOV PLLCON,#03H ; core clk = 2.097152MHz

42 ;Configure uart for 9600 baud at core clk = 2.097152MHz

0063 759E83 43 MOV T3CON,#83h

0066 759D2D 44 MOV T3FD,#2Dh

0069 759852 45 MOV SCON,#52h

46

006C 7800 47 MOV R0, #00 ; start output data at 0

006E 7940 48 MOV R1, #40h ; initialise R1 to 40 to store the

49 ; input data from memory location 40

0070 50 SAVENOS:

0070 E8 51 MOV A,R0

0071 F7 52 MOV @R1, A ; move R0 into memory location R1

0072 09 53 INC R1 ; increment memory location and data so

54 ; new data is stored in new address

0073 08 55 INC R0

0074 B950F9 56 CJNE R1, #50H, SAVENOS ; reset memory location to 40h

57 ; when memory location reaches 50h

58 ; saving 16 bytes of data

UART1 PAGE 2

59

60 ; Transmit the values in locations 40h->50h up the UART wait for

61 ; 5 seconds and then repeat

62

63

0077 B2B4 64 START: CPL LED ;CPL LED with each transmission

0079 9000F0 65 MOV DPTR, #TITLE

007C 1200A4 66 CALL SENDSTRING ; write title block on screen

67

007F 7940 68 MOV R1, #40h ; move value at address 40 into R2

0081 E7 69 MOV A, @R1

0082 FA 70 MOV R2, A

71

0083 72 NEXT: ; Put new value on a new line

0083 740A 73 MOV A, #10 ; Transmit a linefeed (= ASCII 10)

0085 1200BC 74 CALL SENDCHAR

0088 740D 75 MOV A, #13 ;Transmit a carriage return (=ASCII 13)

008A 1200BC 76 CALL SENDCHAR

77

008D EA 78 MOV A, R2 ; Transmit R2 i.e. value @ address R1

008E 1200C4 79 CALL SENDVAL

0091 09 80 INC R1 ; Increment address

0092 E7 81 MOV A, @R1

0093 FA 82 MOV R2, A ; R2 holds the value @ addrR1

83

0094 E9 84 MOV A, R1 ; Check if at address 50h

0095 B450EB 85 CJNE A, #50h, NEXT ; if not jump to Next

0098 02009B 86 JMP WAIT5S ; if so wait 5s and repeat

87

009B 74C8 88 WAIT5S: MOV A, #200

009D 1200E4 89 CALL DELAY ; Wait 2 seconds

00A0 7940 90 MOV R1, #40h

00A2 80CC 91 JMP SAVENOS ; Resave new numbers to same addresses

92

93

94 ;\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

95 ; SENDSTRING

96

00A4 97 SENDSTRING: ; sends ASCII string to UART starting at location

98 ; DPTR and ending with a null (0) value

99

00A4 C0E0 100 PUSH ACC

00A6 C0F0 101 PUSH B

00A8 E4 102 CLR A

00A9 F5F0 103 MOV B,A

00AB E5F0 104 IO0010: MOV A,B

00AD 05F0 105 INC B

00AF 93 106 MOVC A,@A+DPTR

00B0 6005 107 JZ IO0020

00B2 1200BC 108 CALL SENDCHAR

00B5 80F4 109 JMP IO0010

00B7 D0F0 110 IO0020: POP B

00B9 D0E0 111 POP ACC

112

00BB 22 113 RET

114

115 ;\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

116 ; SENDCHAR

UART1 PAGE 3

117

00BC 118 SENDCHAR: ; sends ASCII value contained in A to UART

119

00BC 3099FD 120 JNB TI,$ ; wait til present char gone

00BF C299 121 CLR TI ; must clear TI

00C1 F599 122 MOV SBUF,A

123

00C3 22 124 RET

125

126 ;\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

127 ; SENDVAL

128

00C4 129 SENDVAL: ; converts the hex value of A into two ASCII chars,

130 ; and then spits these two characters up the UART.

131 ; does not change the value of A.

132

00C4 C0E0 133 PUSH ACC

00C6 C4 134 SWAP A

00C7 1200D8 135 CALL HEX2ASCII

00CA 11BC 136 CALL SENDCHAR ; send high nibble

00CC D0E0 137 POP ACC

00CE C0E0 138 PUSH ACC

00D0 1200D8 139 CALL HEX2ASCII

00D3 11BC 140 CALL SENDCHAR ; send low nibble

00D5 D0E0 141 POP ACC

142

00D7 22 143 RET

144

145

146 ;\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

147 ; HEX2ASCII

148

00D8 149 HEX2ASCII: ; converts A into the hex character representing the

150 ; value of A's least significant nibble

151

00D8 540F 152 ANL A,#00Fh

00DA B40A00 153 CJNE A,#00Ah,$+3

00DD 4002 154 JC IO0030

00DF 2407 155 ADD A,#007h

00E1 2430 156 IO0030: ADD A,#'0'

157

00E3 22 158 RET

159

160 ;\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

161 ; DELAY

00E4 162 DELAY: ; Delays by 10ms \* A

163 ; 10mSec based on 2.097152MHZ

164 ; Core Clock

165 ;

166

00E4 FD 167 MOV R5,A ; Acc holds delay variable (1 clock)

00E5 7E1B 168 DLY0: MOV R6,#01Bh ; Set up delay loop0 (2 clocks)

00E7 7FFF 169 DLY1: MOV R7,#0FFh ; Set up delay loop1 (2 clocks)

00E9 DFFE 170 DJNZ R7,$ ; Dec R3 & Jump here until R7 is 0 (3 clocks)

00EB DEFA 171 DJNZ R6,DLY1 ; Dec R2 & Jump DLY1 until R6 is 0 (3 clocks)

00ED DDF6 172 DJNZ R5,DLY0 ; Dec R1 & Jump DLY0 until R5 is 0 (3 clocks)

00EF 22 173 RET ; Return from subroutine

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

UART1 PAGE 4

175

176

00F0 0A0A0D5F 177 TITLE: DB 10,10,13,'\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_',10,13

00F4 5F5F5F5F

00F8 5F5F5F5F

00FC 5F5F5F5F

0100 5F5F5F5F

0104 5F5F5F5F

0108 5F5F5F5F

010C 5F5F5F5F

0110 5F5F5F5F

0114 5F5F5F0A

0118 0D

0119 416E616C 178 DB 'Analog Devices MicroConverter ADuC842',10,13

011D 6F672044

0121 65766963

0125 6573204D

0129 6963726F

012D 436F6E76

0131 65727465

0135 72204144

0139 75433834

013D 320A0D

0140 20202020 179 DB ' UART Demo Routine',10,13

0144 20202020

0148 20554152

014C 54204465

0150 6D6F2052

0154 6F757469

0158 6E650A0D

015C 20204461 180 DB ' Data Stored in Memory in Hex Form',10,13,0

0160 74612053

0164 746F7265

0168 6420696E

016C 204D656D

0170 6F727920

0174 696E2048

0178 65782046

017C 6F726D0A

0180 0D00

181

182

183 END

VERSION 1.2h ASSEMBLY COMPLETE, 0 ERRORS FOUND

UART1 PAGE 5

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

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

DELAY. . . . . . . . . . . . . . C ADDR 00E4H

DLY0 . . . . . . . . . . . . . . C ADDR 00E5H

DLY1 . . . . . . . . . . . . . . C ADDR 00E7H

HEX2ASCII. . . . . . . . . . . . C ADDR 00D8H

IO0010 . . . . . . . . . . . . . C ADDR 00ABH

IO0020 . . . . . . . . . . . . . C ADDR 00B7H

IO0030 . . . . . . . . . . . . . C ADDR 00E1H

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

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

NEXT . . . . . . . . . . . . . . C ADDR 0083H

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

PLLCON . . . . . . . . . . . . . D ADDR 00D7H PREDEFINED

SAVENOS. . . . . . . . . . . . . C ADDR 0070H

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

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

SENDCHAR . . . . . . . . . . . . C ADDR 00BCH

SENDSTRING . . . . . . . . . . . C ADDR 00A4H

SENDVAL. . . . . . . . . . . . . C ADDR 00C4H

START. . . . . . . . . . . . . . C ADDR 0077H NOT USED

T3CON. . . . . . . . . . . . . . D ADDR 009EH PREDEFINED

T3FD . . . . . . . . . . . . . . D ADDR 009DH PREDEFINED

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

TITLE. . . . . . . . . . . . . . C ADDR 00F0H

WAIT5S . . . . . . . . . . . . . C ADDR 009BH