TX1TO10 PAGE 1

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

2 ;

3 ; Author : ADI - Apps

4 ;

5 ; Date : October 2003

6 ;

7 ; File : Tx1to10.asm

8 ;

9 ; Hardware : ADuC842/ADuC843

10 ;

11 ; Description : This Program transmits the numbers 1-10 in binary

12 ; form continuously down the spi port.

13 ; After the transmission of each byte the incoming

14 ; byte is saved in order between internal RAM

15 ; addresses 40h and 50h.

16 ;

17 ; An SPI slave program can be run on a second ADuC832

18 ; to communicate with this master code.

19 ; The Slave program (spislav.asm in the SPI\SLAVE

20 ; directory) should be started after the master

21 ; program (spimast.asm) but within the time delay

22 ; of 5s in order that the slave program is

23 ; synchronised by the first outputted clock of the

24 ; master.

25 ;

26 ; The clock is outputted at sclock (pin 26)

27 ; The data is outputted at sdata/MOSI (pin 27)

28 ; The data is inputted at MISO (pin 14)

29 ;====================================================================

30 ;

31 $MOD842 ;Use 8052 predefined Symbols

32

00B4 33 LED EQU P3.4

0000 34 FLAG BIT 00H

35

36

37 ;\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

38 ; BEGINNING OF CODE

---- 39 CSEG

0000 40 ORG 0000H

41

0000 020060 42 JMP MAIN

43 ;\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

44 ; SPI INTERRUPT ROUTINE

003B 45 ORG 003BH

003B D2B5 46 SETB P3.5 ; set the SS bit after transmission

47

003D C200 48 CLR FLAG ; Clear flag to leave loop

49

003F A7F7 50 MOV @R1, SPIDAT ; move input into memory

0041 09 51 INC R1 ; increment memory location so new

52 ; data is stored in new address

53

0042 B95002 54 CJNE R1, #50H, CONT ; check if memory location =50h.

55 ; if not continue

0045 7940 56 MOV R1, #40H ; if so reset address to 40h to store

57 ; the next 16 bytes to the same

58 ; memory locations

TX1TO10 PAGE 2

0047 32 59 CONT: RETI

60

61

62 ;====================================================================

63

0060 64 ORG 0060H ; Start code at address above interrupts

65

66

67

0060 68 MAIN: ; Main program

69

0060 75F837 70 MOV SPICON,#037h ; Initialise SPICON to have

71 ; -bitrate=fosc/64

72 ; -CPHA=1

73 ; -CPOL=0, sclk idling low

74 ; -master mode select

75 ; -Enable SPI serial port

76

0063 75A901 77 MOV IEIP2, #01h ; Enable SPI interrrupt

78

0066 D2AF 79 SETB EA ; Enable Global Interrupts

80

0068 7940 81 MOV R1, #40h ; initialise R1 to 40 to store the

82 ; input data from memory location 40

006A 7800 83 MOV R0, #00H ; initialise R0 to 0 to start

84 ; transmissions from 1

85

86 ; Delay the output of data by 2.0s in order that the slave program

87 ; can be easily synchronised with the master program.

88

006C 74C8 89 MOV A, #200

006E 120083 90 CALL DELAY

91

0071 92 TRNSMT:

0071 C2B5 93 CLR P3.5 ; clear the SS bit during transmission

0073 08 94 INC R0

0074 88F7 95 MOV SPIDAT, R0 ; transmit the current value on R0

0076 D200 96 SETB FLAG ; set flag so that we wait here until

97 ; the spi interrupt routine clears

98 ; the FLAG

99

0078 2000FD 100 JB FLAG, $ ; stay here until flag is cleared

101 ; by interrupt

102

103 ; check if R0 is equal to 10. If so the number 10 has been

104 ; transmitted and we should reset R0 to 0 to start transmission

105 ; from 1 again

106

007B E8 107 MOV A, R0

007C B40AF2 108 CJNE A, #0AH, TRNSMT ; if R0 is not 10, jump to TRNSMT

007F 7800 109 MOV R0, #00H ; if R0=10 make R0=0 & jump to TRNSMT

0081 80EE 110 JMP TRNSMT

111

112 ;------------------------------------------------------------------

113

0083 114 DELAY: ; Delays by 100ms \* A

115 ; 10mSec based on 2.097152MHZ Core Clock

116

TX1TO10 PAGE 3

0083 FA 117 MOV R2,A ; Acc holds delay variable

0084 7B1B 118 DLY0: MOV R3,#01Bh ; Set up delay loop0

0086 7CFF 119 DLY1: MOV R4,#0FFh ; Set up delay loop1

0088 DCFE 120 DJNZ R4,$ ; Dec R4 & Jump here until R4 is 0

121 ; wait here for

008A DBFA 122 DJNZ R3,DLY1 ; Dec R3 & Jump DLY1 until R3 is 0

123 ; Wait for

008C DAF6 124 DJNZ R2,DLY0 ; Dec R2 & Jump DLY0 until R2 is 0

125 ; wait for ACC\*100ms

008E 22 126 RET ; Return from subroutine

127

128 ;======================================================================

129

130 END

VERSION 1.2h ASSEMBLY COMPLETE, 0 ERRORS FOUND

TX1TO10 PAGE 4

CONT . . . . . . . . . . . . . . C ADDR 0047H

DELAY. . . . . . . . . . . . . . C ADDR 0083H

DLY0 . . . . . . . . . . . . . . C ADDR 0084H

DLY1 . . . . . . . . . . . . . . C ADDR 0086H

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

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

IEIP2. . . . . . . . . . . . . . D ADDR 00A9H PREDEFINED

LED. . . . . . . . . . . . . . . NUMB 00B4H NOT USED

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

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

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

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

TRNSMT . . . . . . . . . . . . . C ADDR 0071H