WDTIMER PAGE 1

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

2 ;

3 ; Author : ADI - Apps www.analog.com/MicroConverter

4 ;

5 ; Date : 28 May 1999

6 ;

7 ; File : WDtimer.asm

8 ;

9 ; Hardware : ADuC812

10 ;

11 ; Description : Demonstrates use of the on-chip watchdog timer.

12 ; In normal operation, WD timer is refreshed by code

13 ; every 100ms, as indicated by a flashing LED. in a

14 ; runaway code condition (which can be simulated here

15 ; by pressing the INT0 button on the eval board) code

16 ; fails to refresh WD bits, and WD timer generates a

17 ; reset after a user selected time-out period (from

18 ; 16ms to 2048ms) has elapsed. the time-out period

19 ; in this routine is 2.048ms for visual clarity.

20 ;

21 ; note: be sure to remove the PSEN pull-down (LK3)

22 ; before allowing watchdog to time-out, or

23 ; you'll end up in serial download mode again

24 ; rather than recovering normal code execution.

25 ;

26 ;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

27

28 $MOD812 ; Use 8052&ADuC812 predefined symbols

29

00B4 30 LED EQU P3.4 ; P3.4 drives red LED on eval board

00D5 31 ERROR EQU F0 ; the 'ERROR' flag is used here to

32 ; simulate an erroneous command that

33 ; sends code into an unknown state

34

35 ;\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

36 ; BEGINNING OF CODE

---- 37 CSEG

38

0000 39 ORG 0000h

40

0000 02004B 41 JMP MAIN ; jump to main program

42 ;\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

43 ; INTERRUPT VECTOR SPACE

0003 44 ORG 0003h ; (INT0 ISR)

45

0003 D2D5 46 SETB ERROR ; simulate an error condition..

47 ; ..when INT0 button is pressed

0005 32 48 RETI

49

50 ;====================================================================

51 ; MAIN PROGRAM

004B 52 ORG 004Bh

53

004B 54 MAIN:

55

56 ; ENABLE INTERRUPT TO TRIGGER SIMULATED ERROR CONDITION...

57

004B D288 58 SETB IT0 ; make INT0 edge triggered

WDTIMER PAGE 2

004D D2AF 59 SETB EA ; enable interrupts

004F D2A8 60 SETB EX0 ; enable INT0 (button on eval board)

0051 C2D5 61 CLR ERROR ; simulate error free operation

62

63 ; SET UP WATCHDOG TIMER...

64

0053 75C0E0 65 MOV WDCON,#0E0h ; 2.048 second timeout period

0056 D2C0 66 SETB WDE ; enable watchdog timer

67

68 ; from this point forward, watchdog bits must be refreshed every

69 ; 2.048 seconds or less. if they are not, watchdog timer will

70 ; generate a reset.

71

72 ; THE BELOW LOOP REPRESENTS NORMAL CODE EXECUTION...

73

0058 B2B4 74 FLASH: CPL LED ; blink (complement) the red LED

005A 120066 75 CALL DELAY ; delay 100ms

005D D2C3 76 SETB WDR1 ; refresh watchdog timer..

005F D2C2 77 SETB WDR2 ; ..bits must be set in this order

0061 30D5F4 78 JNB ERROR,FLASH ; jump if 'ERROR' flag is not set

79

80 ; THE BELOW ENDLESS LOOP REPRESENTS RUN-AWAY CODE EXECUTION...

81

0064 80FE 82 JMP $ ; this endless loop is used to

83 ; represent an unknown state of

84 ; program execution

85

86 ; program will sit in the above endless loop until the watchdog

87 ; period (2048ms) has elapsed, at which time a reset will be

88 ; generated by the watchdog timer, thereby recovering the chip to

89 ; resume normal code execution.

90

91 ;\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

92 ; SUBROUTINE

93

0066 94 DELAY: ; delay 100ms

95

0066 7FC8 96 MOV R7,#200 ; 200 \* 500us = 100ms

0068 7EE5 97 DLY1: MOV R6,#229 ; 229 \* 2.17us = 500us

006A DEFE 98 DJNZ R6,$ ; sit here for 500us

006C DFFA 99 DJNZ R7,DLY1 ; repeat 200 times (100ms total)

006E 22 100 RET

101

102 ;\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

103

104 END

VERSION 1.2h ASSEMBLY COMPLETE, 0 ERRORS FOUND

WDTIMER PAGE 3

DELAY. . . . . . . . . . . . . . C ADDR 0066H

DLY1 . . . . . . . . . . . . . . C ADDR 0068H

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

ERROR. . . . . . . . . . . . . . NUMB 00D5H

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

F0 . . . . . . . . . . . . . . . B ADDR 00D5H PREDEFINED

FLASH. . . . . . . . . . . . . . C ADDR 0058H

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

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

MAIN . . . . . . . . . . . . . . C ADDR 004BH

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

WDCON. . . . . . . . . . . . . . D ADDR 00C0H PREDEFINED

WDE. . . . . . . . . . . . . . . B ADDR 00C0H PREDEFINED

WDR1 . . . . . . . . . . . . . . B ADDR 00C3H PREDEFINED

WDR2 . . . . . . . . . . . . . . B ADDR 00C2H PREDEFINED