PSMON PAGE 1

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

2 ;

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

4 ;

5 ; Date : October 2003

6 ;

7 ; File : PSMon.asm

8 ;

9 ; Hardware : ADuC841

10 ;

11 ; Description : Demonstrates use of on-chip power supply monitor.

12 ; In normal operation, this code flashes the LED at

13 ; approximately 5Hz. When Vdd drops below the user

14 ; specified trip-point (here 4.63V) the PSM interrupt

15 ; is executed. once inside this interrupt service

16 ; routine, this code waits until the PSM interrupt

17 ; bit becomes zero again, indicating that the power

18 ; supply is again above the trip point and has been

19 ; there for at least 256ms. at this point, a RETI

20 ; instruction is executed, and normal code execution

21 ; is resumed, indicated by the flashing LED.

22 ;

23 ;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

24

25 $MOD841 ; Use 8052&ADuC841 predefined symbols

26

00B4 27 LED EQU P3.4 ; P3.4 drives red LED on eval board

28

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

30 ; BEGINNING OF CODE

---- 31 CSEG

32

0000 33 ORG 0000h

34

0000 020060 35 JMP MAIN ; jump to main program

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

37 ; INTERRUPT VECTOR SPACE

0043 38 ORG 0043h ; (PSM ISR)

39

0043 D2B4 40 SETB LED ; turn off the LED to indicate fault

41

42 ; most often, a routine would here be called to store critical values

43 ; in user Flash/EE space and wait in a "safe" state of code execution

44 ; until the PSM interrupt bit becomes zero indicating that adequate

45 ; power supply voltage has returned.

46

0045 E5DF 47 CHECK: MOV A,PSMCON ; PSMCON.5 is the PSM interrupt bit..

0047 20E5FB 48 JB ACC.5,CHECK ; ..it is cleared only when Vdd has

49 ; remained above the trip point for

50 ; 256ms or more.

004A C2B4 51 CLR LED

004C 32 52 RETI ; return only when "all's well"

53

54 ;====================================================================

55 ; MAIN PROGRAM

0060 56 ORG 0060h ; start program above interrupts

57

0060 58 MAIN:

PSMON PAGE 2

59

0060 75DFC1 60 MOV PSMCON, #0C1H ; enable PSM with

61 ; AVdd 4.63V threshold

62 ; DVdd 4.63V threshold

0063 75A922 63 MOV IEIP2, #22H ; enable PSM interrupt

64 ; high priority for PSM interrupt

0066 D2AF 65 SETB EA ; enable interrupts

0068 740A 66 MOV A,#10

006A 67 FLASH: ; Main Routine would go here

006A B2B4 68 CPL LED ; blink LED indicating norm operation

006C 120071 69 CALL DELAY ; delay 100ms

006F 80F9 70 JMP FLASH ; loop

71

72 ;\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

73 ; SUBROUTINE

74

0071 75 DELAY: ; delay 10ms

0071 FD 76 MOV R5,A

0072 7FC8 77 DLY0: MOV R7,#200 ;

0074 7EFF 78 DLY1: MOV R6,#0FFH ;

0076 DEFE 79 DJNZ R6,$ ;

0078 DFFA 80 DJNZ R7,DLY1 ;

007A DDF6 81 DJNZ R5,DLY0

007C 22 82 RET

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

84

85 END

VERSION 1.2h ASSEMBLY COMPLETE, 0 ERRORS FOUND

PSMON PAGE 3

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

CHECK. . . . . . . . . . . . . . C ADDR 0045H

DELAY. . . . . . . . . . . . . . C ADDR 0071H

DLY0 . . . . . . . . . . . . . . C ADDR 0072H

DLY1 . . . . . . . . . . . . . . C ADDR 0074H

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

FLASH. . . . . . . . . . . . . . C ADDR 006AH

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

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

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

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

PSMCON . . . . . . . . . . . . . D ADDR 00DFH PREDEFINED