;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

;

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

;

; Date : October 2003

;

; File : PDown.asm

;

; Hardware : ADuC842/ADuC843

;

; Description : Demonstrates a use of a timer interval counter to

; wake the ADuC842 out of Power down mode after a user

; specified Power down time.

;

; The LED will, on power up, flash at 10Hz. After 5s

; the ADuC842 will enter power down mode (the LED will

; stop flashing in the off position).

;

; By pressing the external interrupt 0 button (INT0) or

; when the user specified time runs out (20s in this

; example) the ADuC842 will wake up and continue

; blinking as before for 5s before entering power down

; mode again.

;

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

$MOD842 ; Use 8052&ADuC832 predefined symbols

LED EQU P3.4 ; P3.4 drives red LED on eval board

;\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

; BEGINNING OF CODE

CSEG

ORG 0000h

JMP MAIN ; jump to main program

;\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

; EXTERNAL INTERRUPT VECTOR SPACE

ORG 0003h

RETI

;\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

; TIC INTERRUPT VECTOR SPACE

ORG 0053h

RETI

;\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

ORG 0060h

MAIN:

MOV IEIP2, #0A4h ; enable time interval interrupt

SETB IT0 ; INT0 edge triggered

SETB EX0 ; enable INT0 (button on eval board)

SETB EA ; enable interrupts

MOV INTVAL, #14h ; initialise intval to 20

; => 20 unit delay

BLINK: MOV R7, #50

LOOP: MOV A, #010 ; Blink light 50 times at 10Hz =>5s

CALL DELAY

CPL LED

DJNZ R7, LOOP

MOV TIMECON, #13h ; initialise timecon to

; -count in secs

; -start all time counters

SETB LED ; turn off light when in power down

MOV PCON, #22h ; power down the ADuC842

; Execution stops here until the ADuC

; is powered up again by either an

; external interrupt or a Time Interval

; Interrupt (20s)

; Note: if using external data mem

; make sure ALE remains toggling after

; you power up again. i.e. PCON.4=0

MOV TIMECON, #12h ; disable TCEN to reset counter to 0

; and to temporarily stop counter

JMP BLINK

;\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

; DELAY

DELAY: ; Delays by 10ms \* A

; 10mSec based on 2.097152MHZ

; Core Clock

; i.e. default ADuC842 Clock

MOV R0,A ; Acc holds delay variable

DLY0: MOV R1,#01Bh ; Set up delay loop0

DLY1: MOV R2,#0FFh ; Set up delay loop1

DJNZ R2,$ ; Dec R2 until R2 is zero

DJNZ R1,DLY1 ; Dec R1 & Jump DLY1 until R1 is 0

DJNZ R0,DLY0 ; Dec R0 & Jump DLY0 until R0 is 0

RET ; Return from subroutine

;\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

END