;======================================================================

;

; Author : ADI - Apps

;

; Date : January 2001

;

; File : ADCtrig.asm

;

; Hardware : ADuC816

;

; Description : Flash led an initial rate of 100ms

; Pressing INTO triggers single conversion

; The ADC result is written to external memory

; The delay rate is increased

; The program waits for the next INTO to repeat the

; above sequence

;

;======================================================================

;

$MOD816 ; Use ADuC816 predefined Symbols

FLAG EQU 00H ; Define Bit

CSEG ; Defines the following as a segment of code

ORG 0000H ; Load Code at '0'

JMP MAIN ; Jump to MAIN

;======================================================================

; (INT0 ISR)

ORG 0003h

MOV B,A ; Copy A (sets delay)

INC A ; Increment delay

MOV ADCMODE,#22H ; Initiate a PRIM ADC single conv

JNB RDY0,$ ; Wait for conversion results

; Write ADC Result H/M to ext. memory

MOV A,ADC0M ; read ADC Middle byte

MOVX @DPTR,A ; write Middle byte to ext memory

INC DPTR

MOV A,ADC0H ; read ADC High byte

MOVX @DPTR,A ; write low High byte to ext memory

INC DPTR

MOV A,B ; Restore A (sets delay)

INC A ; Increment delay

RETI ; Return from Interrupt

;======================================================================

ORG 004Bh ; Subroutines

;------------------------------------------------------------------

DELAY: ; Delays by 100ms \* A

MOV R0,A ; Acc holds delay variable

DLY0: MOV R1,#019h ; Set up delay loop0

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

DJNZ R2,$ ; Dec R2 & Jump here until R2 is 0

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

;======================================================================

MAIN: ; (main program)

; Configure ADC

MOV ADCMODE,#20H ; ENABLE MAIN ADC; Mode- Power down

MOV ADC0CON,#47H ; 24 BITS

; USE EXTERNAL REFERENCE

; AIN1-AIN2

; BIPOLAR MODE

; RANGE = +/-2.56V

SETB IT0 ; INT0 edge triggered

SETB EA ; enable inturrupts

SETB EX0 ; enable INT0

CLR FLAG ; Clear Bit defined as FLAG

MOV A,#01H ; Initialize A -> 1

BLINK: CPL P3.4 ; blink LED using compliment instruction

CALL DELAY ; Jump to subroutine DELAY

JNB FLAG,BLINK ; If FLAG is still cleared the jump to Blink

END