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

;

; Author : ADI - Apps

;

; Date : April 2002

;

; File : adctrig.asm

;

; Description : Flash led an initial rate of 100ms

; Pressing INTO triggers single conversion

; The ADC result is written to internal memory

; The delay rate is increased

; The program waits for the next INTO to repeat the

; above sequence

;

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

;

$MOD814 ; Use ADuC814 predefined Symbols

CHAN EQU 04H

CSEG ; Defines the following as a segment of code

ORG 0000H ; Load Code at '0'

JMP MAIN ; Jump to MAIN

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

;INT0 ISR

ORG 0003h ; (INT0 ISR)

SETB SCONV ; INITIATE A MAIN ADC SINGLE CONVERSION

JNB ADCI,$ ; Wait for conversion results

CLR ADCI ; Clear ADC interrupt flag

MOV @R0,ADCDATAH ; Write ADC Result to memory

INC R0

MOV @R0,ADCDATAL

INC R0

INC A ; Increment delay

RETI ; Return from Interrupt

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

; DELAY

MAIN: ; Configure ADC

MOV ADCCON1,#080h ; power up ADC

MOV ADCCON2,#CHAN ; select channel to convert

MOV DACCON,#0DH ; Turn DAC0 on

MOV DAC0H,#08H ;

MOV DAC0L,#00H ; mid-scale

MOV R0,#80h ; save data at 80h

SETB IT0 ; INT0 edge triggered

SETB EA ; enable inturrupts

SETB EX0 ; enable INT0

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

BLINK:

CPL P3.3 ; blink LED

CALL DELAY

AJMP BLINK

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

; DELAY

DELAY: ; Delays by 100ms \* A

MOV R3,A ; Acc holds delay variable

DLY0: MOV R1,#022h ; Set up delay loop0

DLY1: MOV R2,#0FFh ; 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 R3,DLY0 ; Dec R0 & Jump DLY0 until R0 is 0

RET ; Return from subroutine

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

END