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

;

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

;

; Date : October 2003

;

; File : DMAcont.asm

;

; Description : performs continuous mode DMA conversions on a

; single ADC channel at 138KSPS (assuming an

; 11.0592MHz Mclk).

; Debugger or emulator must be used to view results.

;

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

$MOD841 ; use 8052&ADuC841 predefined symbols

DMACOUNT EQU 64 ; number of AD readings to take

DMAINIT EQU 10h ; top nibble of DMAINIT = ADC channel

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

; DEFINE VARIABLES IN INTERNAL RAM

DSEG

ORG 0060h

DMASTOPH: DS 1 ; DMA stop address hi byte

DMASTOPL: DS 1 ; DMA stop address lo byte

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

; DEFINE SEGMENT OF EXTERNAL RAM

XSEG

ORG 000000h

DMASTART: DS DMACOUNT\*2 ; location for DMA results

DMASTOP: DS 4 ; location for DMA stop sequence

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

; BEGINNING OF CODE

CSEG

ORG 0000h

JMP MAIN ; jump to main program

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

; INTERRUPT VECTOR SPACE

ORG 0033h ; (ADC ISR)

CLR CCONV ; stop conversions

CLR C ; clear C to indicate DMA done

RETI

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

; MAIN PROGRAM

ORG 004Bh

MAIN:

; PRECONFIGURE external RAM for DMA capture on a single channel...

MOV ADCCON1,#00h ; set MD1, MD0 TO 0

MOV DPTR,#DMASTOP ; store DMASTOP 16bit value..

MOV DMASTOPL,DPL ; ..as a high byte & low byte

MOV DMASTOPH,DPH ; (for use in GETSTOPFLAG subr)

MOV DPTR,#DMASTART ; set DPTR to DMASTART address

SETUP: MOV A,#DMAINIT ; set up x-mem with init value

MOVX @DPTR,A

INC DPTR

CLR A ; clear second byte

MOVX @DPTR,A

INC DPTR

CALL GETSTOPFLAG ; C cleared if DPTR>=DMAEND

JC SETUP

MOV A,#DMAINIT ; "dummy" DMA location..

MOVX @DPTR,A ; ..to preceed stop command

INC DPTR

CLR A

MOVX @DPTR,A

INC DPTR

MOV A,#0F0h ; DMA stop command

MOVX @DPTR,A

; CONFIGURE ADC for DMA conversion...

MOV DMAL,#0 ; start address for DMA operation

MOV DMAH,#0 ; (must write DMA registers in this

MOV DMAP,#0 ; order: DMAL, DMAH, DMAP)

MOV ADCCON2,#040h ; enable DMA mode

MOV ADCCON1,#09Ch ; 7.2us conv+acq time

SETB EA ; enable interrupts

SETB EADC ; enable ADC interrupt

; LAUNCH DMA conversion... when complete, ADC interrupt will clear C

SETB CCONV ; start continuous ADC conversions

SETB C

JC $ ; wait for DMA to finish

NOP ;.................................... SET BREAKPOINT HERE

; REPEAT entire program...

JMP MAIN

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

; SUBROUTINE

GETSTOPFLAG: ; clears C if DPTR>=DMASTOP

SETB C

MOV A,DPH

CJNE A,DMASTOPH,RET1 ; C cleared if DPH>=DMASTOPH

MOV A,DPL

CJNE A,DMASTOPL,RET1 ; C cleared if DPL>=DMASTOPL

RET1: RET

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

END