DMATIMER PAGE 1

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

2 ;

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

4 ;

5 ; Date : October 2003

6 ;

7 ; File : DMAtimer.asm

8 ;

9 ; Description : performs Timer2 triggered DMA conversions on a

10 ; single ADC channel at 120KSPS (assuming 11.0592MHz

11 ; Mclk). Debugger or emulator must be used to view

12 ; results.

13 ;

14 ;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

15

16 $MOD841 ; use 8052&ADuC841 predefined symbols

17

0040 18 DMACOUNT EQU 64 ; number of AD readings to take

0010 19 DMAINIT EQU 10h ; top nibble of DMAINIT = ADC channel

20

21 ;\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

22 ; DEFINE VARIABLES IN INTERNAL RAM

---- 23 DSEG

0060 24 ORG 0060h

0060 25 DMASTOPH: DS 1 ; DMA stop address hi byte

0061 26 DMASTOPL: DS 1 ; DMA stop address lo byte

27

28 ;\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

29 ; DEFINE SEGMENT OF EXTERNAL RAM

---- 30 XSEG

0000 31 ORG 000000h

0000 32 DMASTART: DS DMACOUNT\*2 ; location for DMA results

0080 33 DMASTOP: DS 4 ; location for DMA stop sequence

34

35 ;\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

36 ; BEGINNING OF CODE

---- 37 CSEG

0000 38 ORG 0000h

0000 02004B 39 JMP MAIN ; jump to main program

40

41 ;\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

42 ; INTERRUPT VECTOR SPACE

0033 43 ORG 0033h ; (ADC ISR)

0033 C2CA 44 CLR TR2 ; stop Timer2

0035 C3 45 CLR C ; clear C to indicate DMA done

0036 32 46 RETI

47

48 ;====================================================================

49 ; MAIN PROGRAM

004B 50 ORG 004Bh

004B 51 MAIN:

52

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

004B 75EF00 54 MOV ADCCON1,#00h

004E 900080 55 MOV DPTR,#DMASTOP ; store DMASTOP 16bit value..

0051 858261 56 MOV DMASTOPL,DPL ; ..as a high byte & low byte

0054 858360 57 MOV DMASTOPH,DPH ; (for use in GETSTOPFLAG subr)

0057 900000 58 MOV DPTR,#DMASTART ; set DPTR to DMASTART address

DMATIMER PAGE 2

005A 7410 59 SETUP: MOV A,#DMAINIT ; set up x-mem with init value

005C F0 60 MOVX @DPTR,A

005D A3 61 INC DPTR

005E E4 62 CLR A ; clear second byte

005F F0 63 MOVX @DPTR,A

0060 A3 64 INC DPTR

0061 120097 65 CALL GETSTOPFLAG ; C cleared if DPTR>=DMAEND

0064 40F4 66 JC SETUP

67

0066 7410 68 MOV A,#DMAINIT ; "dummy" DMA location..

0068 F0 69 MOVX @DPTR,A ; ..to preceed stop command

0069 A3 70 INC DPTR

006A E4 71 CLR A

006B F0 72 MOVX @DPTR,A

006C A3 73 INC DPTR

74

006D 74F0 75 MOV A,#0F0h ; DMA stop command

006F F0 76 MOVX @DPTR,A

77

78 ; CONFIGURE ADC and Timer2 for DMA conversion...

79

0070 75D200 80 MOV DMAL,#0 ; Timer2 DMA must start from 0

0073 75D300 81 MOV DMAH,#0 ; (must write DMA registers in this

0076 75D400 82 MOV DMAP,#0 ; order: DMAL, DMAH, DMAP)

83

0079 75CAD2 84 MOV RCAP2L,#0D2h ; sample period = 2 \* T2 reload prd

007C 75CBFF 85 MOV RCAP2H,#0FFh ; = 2\*(10000h-FFD2h)\*90ns

007F 75CCD2 86 MOV TL2,#0D2h ; = 2\*46\*90ns

0082 75CDFF 87 MOV TH2,#0FFh ; = 8.28us

88

0085 75D840 89 MOV ADCCON2,#040h ; DMA mode

0088 75EF9E 90 MOV ADCCON1,#09Eh ; Timer2 mode

91

008B D2AF 92 SETB EA ; enable interrupts

008D D2AE 93 SETB EADC ; enable ADC interrupt

94

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

96

008F D2CA 97 SETB TR2 ; run Timer2 = start DMA

0091 D3 98 SETB C

0092 40FE 99 JC $ ; wait for DMA to finish

100

0094 00 101 NOP ;.................................... SET BREAKPOINT HERE

102

103 ; REPEAT entire program...

104

0095 80B4 105 JMP MAIN

106

107 ;\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

108 ; SUBROUTINE

109

0097 110 GETSTOPFLAG: ; clears C if DPTR>=DMASTOP

0097 D3 111 SETB C

0098 E583 112 MOV A,DPH

009A B56005 113 CJNE A,DMASTOPH,RET1 ; C cleared if DPH>=DMASTOPH

009D E582 114 MOV A,DPL

009F B56100 115 CJNE A,DMASTOPL,RET1 ; C cleared if DPL>=DMASTOPL

00A2 22 116 RET1: RET

DMATIMER PAGE 3

117

118 ;\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

119

120 END

VERSION 1.2h ASSEMBLY COMPLETE, 0 ERRORS FOUND

DMATIMER PAGE 4

ADCCON1. . . . . . . . . . . . . D ADDR 00EFH PREDEFINED

ADCCON2. . . . . . . . . . . . . D ADDR 00D8H PREDEFINED

DMACOUNT . . . . . . . . . . . . NUMB 0040H

DMAH . . . . . . . . . . . . . . D ADDR 00D3H PREDEFINED

DMAINIT. . . . . . . . . . . . . NUMB 0010H

DMAL . . . . . . . . . . . . . . D ADDR 00D2H PREDEFINED

DMAP . . . . . . . . . . . . . . D ADDR 00D4H PREDEFINED

DMASTART . . . . . . . . . . . . X ADDR 0000H

DMASTOP. . . . . . . . . . . . . X ADDR 0080H

DMASTOPH . . . . . . . . . . . . D ADDR 0060H

DMASTOPL . . . . . . . . . . . . D ADDR 0061H

DPH. . . . . . . . . . . . . . . D ADDR 0083H PREDEFINED

DPL. . . . . . . . . . . . . . . D ADDR 0082H PREDEFINED

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

EADC . . . . . . . . . . . . . . B ADDR 00AEH PREDEFINED

GETSTOPFLAG. . . . . . . . . . . C ADDR 0097H

MAIN . . . . . . . . . . . . . . C ADDR 004BH

RCAP2H . . . . . . . . . . . . . D ADDR 00CBH PREDEFINED

RCAP2L . . . . . . . . . . . . . D ADDR 00CAH PREDEFINED

RET1 . . . . . . . . . . . . . . C ADDR 00A2H

SETUP. . . . . . . . . . . . . . C ADDR 005AH

TH2. . . . . . . . . . . . . . . D ADDR 00CDH PREDEFINED

TL2. . . . . . . . . . . . . . . D ADDR 00CCH PREDEFINED

TR2. . . . . . . . . . . . . . . B ADDR 00CAH PREDEFINED