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 116KSPS.

11 ; Debugger or emulator must be used to view

12 ; results.

13 ;

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

15

16 $MOD842 ; use 8052&ADuC842 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 75D703 51 MAIN: MOV PLLCON,#03H

52

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

004E 75EF00 54 MOV ADCCON1,#00h

0051 900080 55 MOV DPTR,#DMASTOP ; store DMASTOP 16bit value..

0054 858261 56 MOV DMASTOPL,DPL ; ..as a high byte & low byte

0057 858360 57 MOV DMASTOPH,DPH ; (for use in GETSTOPFLAG subr)

005A 900000 58 MOV DPTR,#DMASTART ; set DPTR to DMASTART address

DMATIMER PAGE 2

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

005F F0 60 MOVX @DPTR,A

0060 A3 61 INC DPTR

0061 E4 62 CLR A ; clear second byte

0062 F0 63 MOVX @DPTR,A

0063 A3 64 INC DPTR

0064 12009C 65 CALL GETSTOPFLAG ; C cleared if DPTR>=DMAEND

0067 40F4 66 JC SETUP

67

0069 7410 68 MOV A,#DMAINIT ; "dummy" DMA location..

006B F0 69 MOVX @DPTR,A ; ..to preceed stop command

006C A3 70 INC DPTR

006D E4 71 CLR A

006E F0 72 MOVX @DPTR,A

006F A3 73 INC DPTR

74

0070 74F0 75 MOV A,#0F0h ; DMA stop command

0072 F0 76 MOVX @DPTR,A

77

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

79

0073 75D200 80 MOV DMAL,#0 ;

0076 75D300 81 MOV DMAH,#0 ;

0079 75D400 82 MOV DMAP,#0 ;

83

007C 75CAF6 84 MOV RCAP2L,#0F6h ; sample period = 2 \* T2 reload prd

007F 75CBFF 85 MOV RCAP2H,#0FFh ; = 2\*(10000h-FFF6h)\*0.476us

0082 75CCF6 86 MOV TL2,#0F6h ; = 2\*9\*0.476us

0085 75CDFF 87 MOV TH2,#0FFh ; = 8.5us

88

0088 00 89 NOP

0089 00 90 NOP

91

92

008A 75D840 93 MOV ADCCON2,#040h ; DMA mode

008D 75EF9E 94 MOV ADCCON1,#09Eh ; Timer2 mode

95

0090 D2AF 96 SETB EA ; enable interrupts

0092 D2AE 97 SETB EADC ; enable ADC interrupt

98

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

100

0094 D2CA 101 SETB TR2 ; run Timer2 = start DMA

0096 D3 102 SETB C

0097 40FE 103 JC $ ; wait for DMA to finish

104

0099 00 105 NOP ;.................................... SET BREAKPOINT HERE

106

107 ; REPEAT entire program...

108

009A 80AF 109 JMP MAIN

110

111 ;\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

112 ; SUBROUTINE

113

009C 114 GETSTOPFLAG: ; clears C if DPTR>=DMASTOP

009C D3 115 SETB C

009D E583 116 MOV A,DPH

DMATIMER PAGE 3

009F B56005 117 CJNE A,DMASTOPH,RET1 ; C cleared if DPH>=DMASTOPH

00A2 E582 118 MOV A,DPL

00A4 B56100 119 CJNE A,DMASTOPL,RET1 ; C cleared if DPL>=DMASTOPL

00A7 22 120 RET1: RET

121

122 ;\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

123

124 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 009CH

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

PLLCON . . . . . . . . . . . . . D ADDR 00D7H PREDEFINED

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

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

RET1 . . . . . . . . . . . . . . C ADDR 00A7H

SETUP. . . . . . . . . . . . . . C ADDR 005DH

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

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

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