DACSYNC PAGE 1

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

2 ;

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

4 ;

5 ; Date : 3 April 2002

6 ;

7 ; File : DACsync.asm

8 ;

9 ; Hardware : ADuC832

10 ;

11 ; Description : Outputs sine waves on DAC0 and DAC1 at 606Hz.

12 ; Output signals are in quadrature with eachother,

13 ; DAC1 leading DAC0 by 90 degrees. the SYNC bit is

14 ; used to ensure that both DAC outputs update

15 ; simultaneously thus avoiding a phase error of 0.625

16 ; degrees.

17 ; Rate calculations assume a core clock of 16.777216MHz, pllcon=0.

18 ;

19 ;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

20

21 $MOD832 ; Use 8052&ADuC832 predefined symbols

22

00B4 23 LED EQU P3.4 ; P3.4 drives red LED on eval board

24

25 ;\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

26 ; BEGINNING OF CODE

---- 27 CSEG

28

0000 29 ORG 0000h

0000 75EF80 30 MOV ADCCON1,#80H

0003 75D700 31 MOV PLLCON,#00H ; Max core frequency

0006 75FD1F 32 MOV DACCON,#01Fh ; both DACs on,12bit,asynchronous

0009 75FA08 33 MOV DAC0H,#008h

000C 75F900 34 MOV DAC0L,#000h ; DAC0 to mid-scale

000F 75FC0F 35 MOV DAC1H,#00Fh

0012 75FBFF 36 MOV DAC1L,#0FFh ; DAC1 to full-scale

37

0015 901000 38 MOV DPTR,#TABLE

39

0018 53FDFB 40 STEP: ANL DACCON,#0FBh ; clear SYNC bit 3

41

001B E4 42 CLR A ; 1

001C 93 43 MOVC A,@A+DPTR ; get high byte for mainDAC.. 2

001D F5FA 44 MOV DAC0H,A ; ..and move it into DAC0 register 1

001F 7420 45 MOV A,#020h ; offset by 90deg for quadratureDAC 1

0021 93 46 MOVC A,@A+DPTR ; get high byte for quadratureDAC.. 2

0022 F5FC 47 MOV DAC1H,A ; ..and move it into DAC1 register 1

0024 A3 48 INC DPTR ; move on to get low bytes 2

49

0025 E4 50 CLR A ; 1

0026 93 51 MOVC A,@A+DPTR ; get low byte for mainDAC.. 2

0027 F5F9 52 MOV DAC0L,A ; ..and update DAC0 1

0029 7420 53 MOV A,#020h ; offset by 90deg for quadratureDAC 1

002B 93 54 MOVC A,@A+DPTR ; get low byte for quadratureDAC.. 2

002C F5FB 55 MOV DAC1L,A ; ..and update DAC1 1

002E A3 56 INC DPTR ; move on for next data point 2

57

002F 43FD04 58 ORL DACCON,#004h ; set SYNC bit 3

DACSYNC PAGE 2

59

0032 53827F 60 ANL DPL,#07Fh ; wrap around at end of table 2

61

0035 E5FA 62 MOV A,DAC0H ; 1

0037 A2E3 63 MOV C,ACC.3 ; MSB of DAC0 value 1

0039 92B4 64 MOV LED,C ; LED = MSB of DAC0 2

65

003B 00 66 NOP ; 1

003C 00 67 NOP ; 1

68

003D 80D9 69 JMP STEP ; 2

70

71 ; numbers at right in the above loop represent the number of machine

72 ; cycles for each instruction. the complete loop takes exactly 36

73 ; machine cycles. with an 16.777216MHz master clock, a machine cycle

74 ; is 715ns, so the above loop takes 25.74us to update each data

75 ; point. since there are 64 data points in the below sine lookup

76 ; table, this results in a 1.64ms period, i.e. a 606Hz frequency.

77

78 ;\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

79 ; SINE LOOKUP TABLE

1000 80 ORG 01000h

81

1000 82 TABLE:

83

1000 07FF 84 DB 007h, 0FFh

1002 08C8 85 DB 008h, 0C8h

1004 098E 86 DB 009h, 08Eh

1006 0A51 87 DB 00Ah, 051h

1008 0B0F 88 DB 00Bh, 00Fh

100A 0BC4 89 DB 00Bh, 0C4h

100C 0C71 90 DB 00Ch, 071h

100E 0D12 91 DB 00Dh, 012h

1010 0DA7 92 DB 00Dh, 0A7h

1012 0E2E 93 DB 00Eh, 02Eh

1014 0EA5 94 DB 00Eh, 0A5h

1016 0F0D 95 DB 00Fh, 00Dh

1018 0F63 96 DB 00Fh, 063h

101A 0FA6 97 DB 00Fh, 0A6h

101C 0FD7 98 DB 00Fh, 0D7h

101E 0FF5 99 DB 00Fh, 0F5h

1020 0FFF 100 DB 00Fh, 0FFh

1022 0FF5 101 DB 00Fh, 0F5h

1024 0FD7 102 DB 00Fh, 0D7h

1026 0FA6 103 DB 00Fh, 0A6h

1028 0F63 104 DB 00Fh, 063h

102A 0F0D 105 DB 00Fh, 00Dh

102C 0EA5 106 DB 00Eh, 0A5h

102E 0E2E 107 DB 00Eh, 02Eh

1030 0DA7 108 DB 00Dh, 0A7h

1032 0D12 109 DB 00Dh, 012h

1034 0C71 110 DB 00Ch, 071h

1036 0BC4 111 DB 00Bh, 0C4h

1038 0B0F 112 DB 00Bh, 00Fh

103A 0A51 113 DB 00Ah, 051h

103C 098E 114 DB 009h, 08Eh

103E 08C8 115 DB 008h, 0C8h

1040 07FF 116 DB 007h, 0FFh

DACSYNC PAGE 3

1042 0736 117 DB 007h, 036h

1044 0670 118 DB 006h, 070h

1046 05AD 119 DB 005h, 0ADh

1048 04EF 120 DB 004h, 0EFh

104A 043A 121 DB 004h, 03Ah

104C 038D 122 DB 003h, 08Dh

104E 02EC 123 DB 002h, 0ECh

1050 0257 124 DB 002h, 057h

1052 01D0 125 DB 001h, 0D0h

1054 0159 126 DB 001h, 059h

1056 00F1 127 DB 000h, 0F1h

1058 009B 128 DB 000h, 09Bh

105A 0058 129 DB 000h, 058h

105C 0027 130 DB 000h, 027h

105E 0009 131 DB 000h, 009h

1060 0000 132 DB 000h, 000h

1062 0009 133 DB 000h, 009h

1064 0027 134 DB 000h, 027h

1066 0058 135 DB 000h, 058h

1068 009B 136 DB 000h, 09Bh

106A 00F1 137 DB 000h, 0F1h

106C 0159 138 DB 001h, 059h

106E 01D0 139 DB 001h, 0D0h

1070 0257 140 DB 002h, 057h

1072 02EC 141 DB 002h, 0ECh

1074 038D 142 DB 003h, 08Dh

1076 043A 143 DB 004h, 03Ah

1078 04EF 144 DB 004h, 0EFh

107A 05AD 145 DB 005h, 0ADh

107C 0670 146 DB 006h, 070h

107E 0736 147 DB 007h, 036h ; end of table

148

1080 07FF 149 DB 007h, 0FFh ; repeat first 90degrees for quadratureDAC

1082 08C8 150 DB 008h, 0C8h

1084 098E 151 DB 009h, 08Eh

1086 0A51 152 DB 00Ah, 051h

1088 0B0F 153 DB 00Bh, 00Fh

108A 0BC4 154 DB 00Bh, 0C4h

108C 0C71 155 DB 00Ch, 071h

108E 0D12 156 DB 00Dh, 012h

1090 0DA7 157 DB 00Dh, 0A7h

1092 0E2E 158 DB 00Eh, 02Eh

1094 0EA5 159 DB 00Eh, 0A5h

1096 0F0D 160 DB 00Fh, 00Dh

1098 0F63 161 DB 00Fh, 063h

109A 0FA6 162 DB 00Fh, 0A6h

109C 0FD7 163 DB 00Fh, 0D7h

109E 0FF5 164 DB 00Fh, 0F5h

10A0 0FFF 165 DB 00Fh, 0FFh

166

167 ;\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

168

169 END

VERSION 1.2h ASSEMBLY COMPLETE, 0 ERRORS FOUND

DACSYNC PAGE 4

ACC. . . . . . . . . . . . . . . D ADDR 00E0H PREDEFINED

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

DAC0H. . . . . . . . . . . . . . D ADDR 00FAH PREDEFINED

DAC0L. . . . . . . . . . . . . . D ADDR 00F9H PREDEFINED

DAC1H. . . . . . . . . . . . . . D ADDR 00FCH PREDEFINED

DAC1L. . . . . . . . . . . . . . D ADDR 00FBH PREDEFINED

DACCON . . . . . . . . . . . . . D ADDR 00FDH PREDEFINED

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

LED. . . . . . . . . . . . . . . NUMB 00B4H

P3 . . . . . . . . . . . . . . . D ADDR 00B0H PREDEFINED

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

STEP . . . . . . . . . . . . . . C ADDR 0018H

TABLE. . . . . . . . . . . . . . C ADDR 1000H