DACQUADR PAGE 1

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

2 ;

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

4 ;

5 ; Date : 31 May 2002

6 ;

7 ; File : DACquadr.asm

8 ;

9 ; Hardware : ADuC831

10 ;

11 ; Description : Outputs sine waves on DAC0 and DAC1 at 400Hz.

12 ; Output signals are in quadrature with eachother,

13 ; DAC1 leading DAC0 by 90 degrees. since each DAC is

14 ; updated when its DACxL register is written to, they

15 ; are not updated at the exact same moment, and a

16 ; phase error of (in this case) 0.625degrees results.

17 ; to address this problem, see code: "DACsync.asm".

18 ; Rate calculations assume an 11.0592MHz Mclk.

19 ;

20 ;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

21

22 $MOD831 ; Use 8052&ADuC831 predefined symbols

23

00B4 24 LED EQU P3.4 ; P3.4 drives red LED on eval board

25

26 ;\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

27 ; BEGINNING OF CODE

---- 28 CSEG

29

0000 30 ORG 0000h

0000 75EF80 31 MOV ADCCON1,#80H

0003 75FD1F 32 MOV DACCON,#01Fh ; both DACs on,12bit,asynchronous

0006 75FA08 33 MOV DAC0H,#008h

0009 75F900 34 MOV DAC0L,#000h ; DAC0 to mid-scale

000C 75FC0F 35 MOV DAC1H,#00Fh

000F 75FBFF 36 MOV DAC1L,#0FFh ; DAC1 to full-scale

37

0012 901000 38 MOV DPTR,#TABLE

39

0015 E4 40 STEP: CLR A ; 1

0016 93 41 MOVC A,@A+DPTR ; get high byte for mainDAC.. 2

0017 F5FA 42 MOV DAC0H,A ; ..and move it into DAC0 register 1

0019 7420 43 MOV A,#020h ; offset by 90deg for quadratureDAC 1

001B 93 44 MOVC A,@A+DPTR ; get high byte for quadratureDAC.. 2

001C F5FC 45 MOV DAC1H,A ; ..and move it into DAC1 register 1

001E A3 46 INC DPTR ; move on to get low bytes 2

47

001F E4 48 CLR A ; 1

0020 93 49 MOVC A,@A+DPTR ; get low byte for mainDAC.. 2

0021 F5F9 50 MOV DAC0L,A ; ..and update DAC0 1

0023 7420 51 MOV A,#020h ; offset by 90deg for quadratureDAC 1

0025 93 52 MOVC A,@A+DPTR ; get low byte for quadratureDAC.. 2

0026 F5FB 53 MOV DAC1L,A ; ..and update DAC1 1

0028 A3 54 INC DPTR ; move on for next data point 2

55

0029 53827F 56 ANL DPL,#07Fh ; wrap around at end of table 2

57

002C E5FA 58 MOV A,DAC0H ; 1

DACQUADR PAGE 2

002E A2E3 59 MOV C,ACC.3 ; MSB of DAC0 value 1

0030 92B4 60 MOV LED,C ; LED = MSB of DAC0 2

61

0032 00 62 NOP ; 1

0033 00 63 NOP ; 1

0034 00 64 NOP ; 1

0035 00 65 NOP ; 1

0036 00 66 NOP ; 1

0037 00 67 NOP ; 1

0038 00 68 NOP ; 1

0039 00 69 NOP ; 1

70

003A 80D9 71 JMP STEP ; 2

72

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

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

75 ; machine cycles. with an 11.0592MHz master clock, a machine cycle

76 ; is 1.085us, so the above loop takes 39.06us to update each data

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

78 ; table, this results in a 2.50ms period, i.e. a 400Hz frequency.

79

80 ;\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

81 ; SINE LOOKUP TABLE

1000 82 ORG 01000h

83

1000 84 TABLE:

85

1000 07FF 86 DB 007h, 0FFh

1002 08C8 87 DB 008h, 0C8h

1004 098E 88 DB 009h, 08Eh

1006 0A51 89 DB 00Ah, 051h

1008 0B0F 90 DB 00Bh, 00Fh

100A 0BC4 91 DB 00Bh, 0C4h

100C 0C71 92 DB 00Ch, 071h

100E 0D12 93 DB 00Dh, 012h

1010 0DA7 94 DB 00Dh, 0A7h

1012 0E2E 95 DB 00Eh, 02Eh

1014 0EA5 96 DB 00Eh, 0A5h

1016 0F0D 97 DB 00Fh, 00Dh

1018 0F63 98 DB 00Fh, 063h

101A 0FA6 99 DB 00Fh, 0A6h

101C 0FD7 100 DB 00Fh, 0D7h

101E 0FF5 101 DB 00Fh, 0F5h

1020 0FFF 102 DB 00Fh, 0FFh

1022 0FF5 103 DB 00Fh, 0F5h

1024 0FD7 104 DB 00Fh, 0D7h

1026 0FA6 105 DB 00Fh, 0A6h

1028 0F63 106 DB 00Fh, 063h

102A 0F0D 107 DB 00Fh, 00Dh

102C 0EA5 108 DB 00Eh, 0A5h

102E 0E2E 109 DB 00Eh, 02Eh

1030 0DA7 110 DB 00Dh, 0A7h

1032 0D12 111 DB 00Dh, 012h

1034 0C71 112 DB 00Ch, 071h

1036 0BC4 113 DB 00Bh, 0C4h

1038 0B0F 114 DB 00Bh, 00Fh

103A 0A51 115 DB 00Ah, 051h

103C 098E 116 DB 009h, 08Eh

DACQUADR PAGE 3

103E 08C8 117 DB 008h, 0C8h

1040 07FF 118 DB 007h, 0FFh

1042 0736 119 DB 007h, 036h

1044 0670 120 DB 006h, 070h

1046 05AD 121 DB 005h, 0ADh

1048 04EF 122 DB 004h, 0EFh

104A 043A 123 DB 004h, 03Ah

104C 038D 124 DB 003h, 08Dh

104E 02EC 125 DB 002h, 0ECh

1050 0257 126 DB 002h, 057h

1052 01D0 127 DB 001h, 0D0h

1054 0159 128 DB 001h, 059h

1056 00F1 129 DB 000h, 0F1h

1058 009B 130 DB 000h, 09Bh

105A 0058 131 DB 000h, 058h

105C 0027 132 DB 000h, 027h

105E 0009 133 DB 000h, 009h

1060 0000 134 DB 000h, 000h

1062 0009 135 DB 000h, 009h

1064 0027 136 DB 000h, 027h

1066 0058 137 DB 000h, 058h

1068 009B 138 DB 000h, 09Bh

106A 00F1 139 DB 000h, 0F1h

106C 0159 140 DB 001h, 059h

106E 01D0 141 DB 001h, 0D0h

1070 0257 142 DB 002h, 057h

1072 02EC 143 DB 002h, 0ECh

1074 038D 144 DB 003h, 08Dh

1076 043A 145 DB 004h, 03Ah

1078 04EF 146 DB 004h, 0EFh

107A 05AD 147 DB 005h, 0ADh

107C 0670 148 DB 006h, 070h

107E 0736 149 DB 007h, 036h ; end of table

150

1080 07FF 151 DB 007h, 0FFh ; repeat first 90degrees for quadratureDAC

1082 08C8 152 DB 008h, 0C8h

1084 098E 153 DB 009h, 08Eh

1086 0A51 154 DB 00Ah, 051h

1088 0B0F 155 DB 00Bh, 00Fh

108A 0BC4 156 DB 00Bh, 0C4h

108C 0C71 157 DB 00Ch, 071h

108E 0D12 158 DB 00Dh, 012h

1090 0DA7 159 DB 00Dh, 0A7h

1092 0E2E 160 DB 00Eh, 02Eh

1094 0EA5 161 DB 00Eh, 0A5h

1096 0F0D 162 DB 00Fh, 00Dh

1098 0F63 163 DB 00Fh, 063h

109A 0FA6 164 DB 00Fh, 0A6h

109C 0FD7 165 DB 00Fh, 0D7h

109E 0FF5 166 DB 00Fh, 0F5h

10A0 0FFF 167 DB 00Fh, 0FFh

168

169 ;\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

170

171 END

VERSION 1.2h ASSEMBLY COMPLETE, 0 ERRORS FOUND

DACQUADR 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

STEP . . . . . . . . . . . . . . C ADDR 0015H

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