COP820CJ,COP840CJ,COP880C,COP884BC, COP888CF,COP888CL,COP888EK,COP888FH, COP888GW,COP8ACC5,COP8AME9,COP8CBE9, COP8CBR9,COP8CCE9,COP8CCR9,COP8CDR9, COP8SAA7,COP8SAC7,COP8SBR9,COP8SCR9, COP8SDR9,COP8SGE5,COP8SGE7,COP8SGG5, COP8SGH5,COP8SGK5,COP8SGR5,COP8SGR7, COP912C,DS3658

AN-714 Using COP800 Devices to Control DC Stepper Motors

Literature Number: SNLA019

Using COP800 Devices to Control DC Stepper Motors

National Semiconductor
Application Note 714
Michelle Giles
September 1990

INTRODUCTION

COP800 devices can be used to control DC stepper motors with limited effort. This application note describes the use of a COP820 to control the speed, direction and rotation angle of a stepper motor. In addition to the COP820, this application requires a quad high current peripheral driver (DS3658) to meet the high current needs of the stepper motor.

DC STEPPER MOTOR

A DC stepper motor translates current pulses into rotor movement. A typical motor contains four winding coils labeled red, yellow/white, red/white, and yellow. Applying current to these windings forces the motor to step. For normal operation, two windings are activated (pulsed) concurrently. The motor moves clockwise one step per change in windings activated with the following activation sequence: red and yellow, yellow and red/white, red/white and yellow/white, yellow/white and red, repeat. Half-steps may be generated by altering the sequence to: red and yellow, yellow, yellow and red/white, red/white, red/white and yellow/white, yellow/white, yellow/white and red, red, repeat. The motor runs in a counterclockwise direction if either sequence is applied in reverse order. The speed of rotation (number of steps/second) is controlled by the frequency of the pulses.

COP820 CONTROL OF STEPPER MOTOR

The COP820 controls the stepper motor by sending pulse sequences to the motor windings in response to control commands. Commands executed by the code in this application include: single step the motor in a clockwise or counterclockwise direction (i.e. rotate the rotor through a certain number of degrees), run the motor continuously at one of four speeds in a clockwise or counterclockwise direction, and stop the motor.

Note: Half-stepping is not implemented in this example.

During continuous mode operation, the 16-bit timer of the COP820 is used to control the speed of the stepper motor. The timer is set up with a value that causes an underflow once every x seconds or at a frequency of 1/x. Each underflow of the timer interrupts the microcontroller. In response to the timer interrupt, the microcontroller generates a new pulse and causes a single step of the motor. Thus the motor steps at the frequency of the timer underflows. This application sets up the timer to generate interrupts at four different frequencies. These frequencies produce the following motor speeds: 25 steps/second, 100 steps/second, 200 steps/second, and 400 steps/second.

The determination of which windings to activate and deactivate to step the motor is performed by a single subroutine in this example. A block of memory is allocated to store a step pointer and the four possible stepper drive values are shown in Table I (9,C,6,3). Consecutive memory locations are used to store the stepper drive values so that applying the value from location X and then location X+1 (or X−1) causes the motor to step once. The motor drive subroutine increments or decrements the pointer to the current drive value based on the selection of a clockwise or counterclockwise direction. Writing the value from the newly selected location to the motor causes a single step of the motor in the appropriate direction.

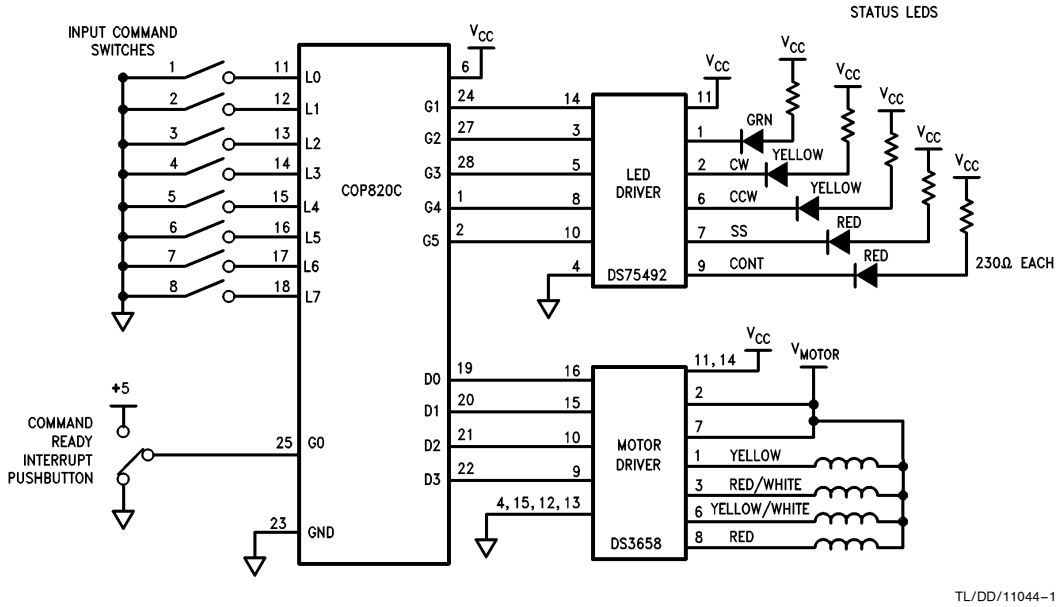
During single step operation, the microcontroller steps the motor the exact number of times requested in the control command. Each step corresponds to 1.8 degrees of rotor movement. Therefore, a request to perform 200 steps will rotate the rotor through one complete revolution (360 degrees) at a fixed speed.

A block diagram of the application is shown in *Figure 1*. A flowchart of the code used to control the motor is given in *Figure 2*. The complete code is given at the end.

TABLE I. Stepper Motor Drive Sequence

Step	Yellow	Red/White	Yellow/White	Red	Hex Value
0	ON	OFF	OFF	ON	9
1	ON	ON	OFF	OFF	C
2	OFF	ON	ON	OFF	6
3	OFF	OFF	ON	ON	3
4	ON	OFF	OFF	ON	9

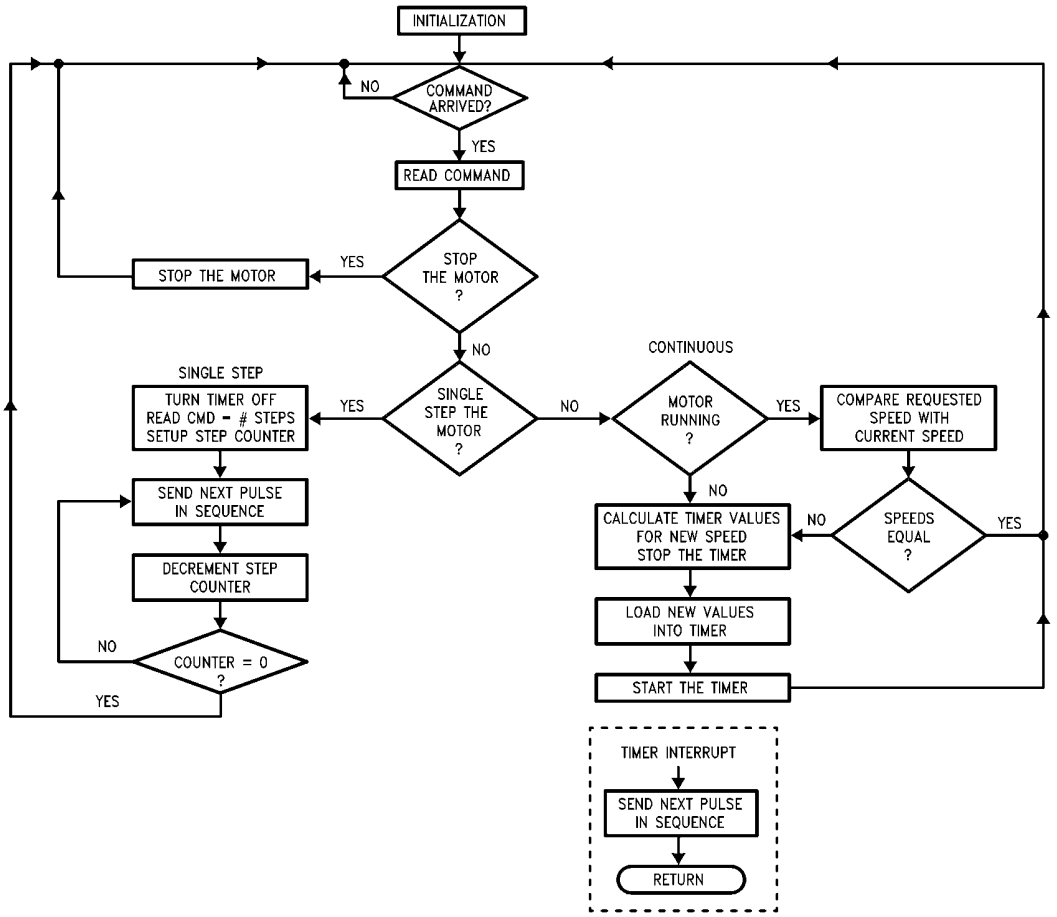
 TL/DD11044 RRD-B30M75/Printed in U. S. A.



TL/DD/11044–1

FIGURE 1. Schematic Diagram

Program Code Flow Chart



TL/DD/11044–2

FIGURE 2. Program Flowchart

NATIONAL SEMICONDUCTOR CORPORATION
COP800 CROSS ASSEMBLER, REV: D1, 12 OCT 88

```
                        ;STEPPER MOTOR CONTROL PROGRAM
                        ;MAY 1990
                        ;
                        ;This program controls the speed, direction, and degree of rotation of
                        ;a DC stepper motor.
                        ;
                        ;                                Memory Map
                        ; RAM   CONTENTS
                        ;  00   (MS0) step motor drive value 09H    (two windings active per pulse)
                        ;  01   (MS1) step motor drive value 0CH
                        ;  02   (MS2) step motor drive value 06H
                        ;  03   (MS3) step motor drive value 03H
                        ;  04   (CMD) control command
                        ;       bit7 - bit4 = motor speed or upper nibble of # single steps
                        ;       bit 3       = unused
                        ;       bit 2       = (MODE) single step or continuous mode select (1 = ss)
                        ;       bit 1       = (DIR) cw or ccw direction select (1 = cw)
                        ;       bit 0       = (GO)  motor go or motor stop select (1 = stop)
                        ;  05   (STEPS) lower byte of number of single steps
                        ;  07   (FLGREG) flag register
                        ;       bit 0       = (INT) ready to read in cmd (ext int occured)
                        ;       bit1 - bit7 = unused
                        ;  14   (TVAL0) value to load in lower byte of timer for speed X
                        ;  15   (TVAL1) value to load in upper byte of timer for speed X
                        ;  D2   (PORTLP) port L input pins used for incomming commands
                        ;  D4   (PORTGD) port G data pins used to drive status LEDs
                        ;  DC   (PORTD)  port D data pins used to ouput pulses to the stepper motor
                        ;  F0   (CREG0) step counter register zero
                        ;  F1   (CREG1) step counter register one
                        ;  F2   (STPPTR) pointer to current step motor drive value (RAM 00 - 03)
                        ;
                        ;REGISTER AND CONSTANT DEFINITIONS

                        ;COMMAND BITS
0000                            GO      = 0                     ;GO COMMAND BIT
                                                                ;  1 = STOP         0 = GO
0001                            DIR     = 1                     ;DIRECTION COMMAND BIT
                                                                ;  1 = CW           0 = CCW
0002                            MODE    = 2                     ;MODE COMMAND BIT
                                                                ;  1 = SINGLE STEP  0 = CONTINUOUS

                        ;PORTG BITS
0000                            INT     = 0                     ;FLAG BIT (SET IF EXTINT OCCURS)
0001                            READY   = 1                     ;READY LED
0002                            CW      = 2                     ;CLOCKWISE LED
0003                            CCW     = 3                     ;COUNTER CLOCKWISE LED
0004                            SS      = 4                     ;SINGLE STEP LED
0005                            NS      = 5                     ;CONTINUOUS (NON-STOP) LED

                        ;REGISTERS
0004                            CMD     = 04                    ;INPUT COMMAND STORAGE REGISTER
```

TL/DD/11044–3

NATIONAL SEMICONDUCTOR CORPORATION
COP800 CROSS ASSEMBLER, REV: D1, 12 OCT 88

```
  52       0005                    STEPS   = 05                ;INPUT #STEPS/SPEED REGISTER
  53       00F0                    CREG0   = 0F0               ;COUNTER REGISTER
  54       00F1                    CREG1   = 0F1               ;COUNTER REGISTER
  55       0007                    FLGREG  = 07                ;FLAG REGISTER (FLAG BITS)
  56       00F2                    STPPTR  = 0F2               ;CURRENT MOTOR STEP POINTER
  57       0014                    TVAL0   = 014               ;MOTOR SPEED LOAD VALUES
  58       0015                    TVAL1   = 015
  59       0000                    MS0     = 00                ;STEPPER MOTOR DRIVE VALUES
  60       0001                    MS1     = 01
  61       0002                    MS2     = 02
  62       0003                    MS3     = 03
  63
  64                           ;ASSIGNMENTS FOR COP820
  65
  66       00D5                    PORTGC  = 0D5
  67       00D4                    PORTGD  = 0D4
  68       00D6                    PORTGP  = 0D6
  69       00D1                    PORTLC  = 0D1
  70       00D0                    PORTLD  = 0D0
  71       00D2                    PORTLP  = 0D2
  72       00DC                    PORTD   = 0DC
  73       00D7                    PORTI   = 0D7
  74       00E9                    SIOR    = 0E9
  75       00EA                    TMRLO   = 0EA
  76       00EB                    TMRHI   = 0EB
  77       00EC                    TAULO   = 0EC
  78       00ED                    TAUHI   = 0ED
  79       00EE                    CNTRL   = 0EE
  80       00EF                    PSW     = 0EF
  81
  82
  83       0000                    GIE     = 0
  84       0001                    ENI     = 1
  85       0002                    BUSY    = 2
  86       0003                    IPND    = 3
  87       0004                    ENTI    = 4
  88       0005                    TPND    = 5
  89
  90       0002                    IEDG    = 2
  91       0003                    MSEL    = 3
  92       0004                    TRUN    = 4
  93       0005                    TC3     = 5
  94       0006                    TC2     = 6
  95       0007                    TC1     = 7
  96
  97
  98                               .CHIP   820
  99
 100                           ;INITIALIZATION OF REGISTERS     ;*******************************
 101 0000 DD2F                    LD      SP,#02F
 102 0002 BCEE80                  LD      CNTRL,#080
```

TL/DD/11044–4

```
 103 0005 BCEF03                LD      PSW,#003                ;GLOBAL INT ENABLE/EXTINT ENABLE
 104 0008 BCD401                LD      PORTGD,#01
 105 000B BCD53E                LD      PORTGC,#03E             ;CONFIG PORTG FOR OUTPUTS
 106 000E BCDC09                LD      PORTD,#09               ;START MOTOR DRIVE VALUE
 107 0011 BCD100                LD      PORTLC,#00              ;CONFIG PORTL FOR INPUTS
 108 0014 BCD0FF                LD      PORTLD,#0FF             ;CONFIG PORTL FOR WEAK PULL-UPS
 109 0017 5F                    LD      B,#MS0                  ;SETUP MOTOR DRIVE VALUES
 110 0018 9A09                  LD      [B+],#09
 111 001A 9A0C                  LD      [B+],#0C
 112 001C 9A06                  LD      [B+],#06
 113 001E 9E03                  LD      [B],#03
 114 0020 D200                  LD      STPPTR,#00              ;INIT STEP POINTER
 115 0022 BC0700                LD      FLGREG,#00              ;INIT FLAG REGISTER
 116
 117
 118                    ;READ, DECODE, AND EXECUTE COMMAND
 119                                                            ;*******************************
 120 0025 BDD479        TOP:    SBIT    READY,PORTGD            ;TURN ON READY FOR NEXT CMD LED
 121 0028 3081                  JSR     WAIT                    ;WAIT FOR CMD AND READ CMD
 122 002A BDD469                RBIT    READY,PORTGD            ;TURN OFF READY FOR NEXT CMD LED
 123 002D 9C04                  X       A,CMD                   ;STORE IN CMD REGISTER
 124 002F BD0470                IFBIT   GO,CMD                  ;IF STOP BIT SET
 125 0032 08                    JP      STOP                    ;THEN STOP MOTOR
 126 0033 BD0472                IFBIT   MODE,CMD                ;ELSE CHEK MODE
 127 0036 3041                  JSR     SSTEP                   ;IF MODE SET THEN GO SINGLE STEP
 128 0038 305F                  JSR     CONT                    ;ELSE GO CONTINUOUS
 129 003A EA                    JP      TOP                     ;GO WAIT FOR NEXT COMMAND
 130                    STOP:                                   ;STOP THE MOTOR
 131 003B 308E                  JSR     TMRSET                  ;STOP THE TIMER
 132 003D BCD401                LD      PORTGD,#01              ;TURN OFF ALL LEDS
 133 0040 E4                    JP      TOP                     ;GO WAIT FOR NEXT CMD
 134
 135
 136                    ;SINGLE STEP THE MOTOR  (SS)
 137
 138                    SSTEP:                                  ;*******************************
 139 0041 308E                  JSR     TMRSET                  ;STOP TIMER
 140 0043 BCD410                LD      PORTGD,#010             ;TURN ON SS LED (RST ALL OTHER LEDS)
 141 0046 3081                  JSR     WAIT                    ;WAIT FOR CMD BYTE 2 (# STEPS)
 142 0048 8A                    INC     A                       ;ADD 1 TO CORRECT FOR LOOP
 143 0049 9CF0                  X       A,CREG0                 ;STORE #STEPS IN LOBYTE COUNT REG
 144 004B 9D04                  LD      A,CMD                   ;LOAD HIBYTE # STEPS
 145 004D 65                    SWAP    A                       ;MOVE TO LOWER NIBBLE
 146 004E 950F                  AND     A,#0F                   ;GET RID OF UPPER BITS
 147 0050 8A                    INC     A                       ;ADD 1 TO CORRECT FOR LOOP
 148 0051 9CF1                  X       A,CREG1                 ;MOVE TO HIBYTE OF COUNT REG
 149 0053 C0            TP2:    DRSZ    CREG0                   ;DECR LOBYTE AND IF NOT ZERO
 150 0054 05                    JP      DO                      ;THEN GO DO A STEP
 151 0055 C1            MID:    DRSZ    CREG1                   ;ELSE DECR HIBYTE AND IF NOT ZERO
 152 0056 01                    JP      DO2                     ;THEN GO DO A STEP AND RST LO COUNT
 153 0057 8D                    RETSK                           ;ELSE END OF LOOP    RETURN
```

TL/DD/11044–5

```
154 0058 D0FF           DO2:    LD      CREG0,#0FF              ;RESET LOBYTE OF COUNTER
155 005A 3098           DO:     JSR     NXTVAL                  ;STEP THE MOTOR
156 005C 3158                   JSR     DELAY                   ;SLOW THE STEPPING
157 005E F4                     JP      TP2                     ;GO TO TOP OF LOOP
158
159
160                     ;RUN THE MOTOR CONTINUOUSLY  (NS = NON-STOP = CONTINUOUSLY)
161
162                     CONT:                                   ;*****************************
163 005F BDEE74                 IFBIT   TRUN,CNTRL              ;IF MOTOR ALREADY RUNNING NS
164 0062 01                     JP      CHKSPD                  ;THEN CHECK THE CURRENT SPEED
165 0063 03                     JP      SETGO                   ;ELSE GO START THE MOTOR
166 0064 3148           CHKSPD: JSR     SPEED                   ;COMPARE INPUT WITH ACTUAL SPD
167 0066 8E                     RET                             ;IF EQUAL RET ELSE RESTART MOTOR
168 0067 308E           SETGO:  JSR     TMRSET                  ;STOP THE TIMER
169 0069 BCD420                 LD      PORTGD,#020             ;TURN ON CONTINUOUS LED
170 006C 3126                   JSR     TIMVAL                  ;CALCULATE TIMER (SPEED) VALUE
171 006E AE                     LD      A,[B]                   ;LOAD A WITH TVAL1
172 006F 9CEB                   X       A,TMRH1                 ;MOVE SPEED VAL INTO TIMER
173 0071 AB                     LD      A,[B-]                  ;LOAD A WITH TVAL1 POINT TO TVAL0
174 0072 9CED                   X       A,TAUH1                 ;MOVE SPEED VAL INTO AUTORELOAD REG
175 0074 AE                     LD      A,[B]                   ;LOAD A WITH TVAL0
176 0075 9CEA                   X       A,TMRLO                 ;MOVE SPEED VAL INTO TIMER
177 0077 AE                     LD      A,[B]                   ;LOAD A WITH TVAL0
178 0078 9CEC                   X       A,TAULO
179 007A BDEF7C                 SBIT    ENTI,PSW                ;ENABLE TIMER INTERRUPT
180 007D BDEE7C                 SBIT    TRUN,CNTRL              ;START THE TIMER
181 0080 8E                     RET                             ;RET TO MAIN AND WAIT FOR TMRINT
182
183                     ;SUPPORT ROUTINES ********************************************************
184
185                     WAIT:                                   ;*****************************
186                             ;WAIT FOR AN EXTERNAL INTERRUPT TO SIGNAL AN INCOMMING COMMAND
187                             ;READ THE INCOMMING COMMAND FROM PORT L
188 0081 BD0770                 IFBIT   INT,FLGREG              ;IF EXTERNAL INTERRUPT OCCURED
189 0084 01                     JP      OUT                     ;THEN JUMP OUT OF LOOP
190 0085 FB                     JP      WAIT                    ;ELSE CONTINUE TO WAIT
191 0086 BD0768         OUT:    RBIT    INT,FLGREG              ;RESET EXTERNAL INTERRUPT FLAG
192 0089 9DD2                   LD      A,PORTLP                ;READ INCOMMING COMMAND
193 008B 96FF                   XOR     A,#0FF                  ;COMPLEMENT INCOMMING COMMAND
194 008D 8E                     RET                             ;RETURN COMMAND IN ACC
195
196
197                     TMRSET:                                 ;*****************************
198                             ;RESET THE TIMER
199 008E BDEE6C                 RBIT    TRUN,CNTRL              ;STOP THE TIMER
200 0091 BDEF6D                 RBIT    TPND,PSW                ;RESET THE TIMER PENDING BIT
201 0094 BDEF6C                 RBIT    ENTI,PSW                ;DISABLE TIMER INTERRUPT
202 0097 8E                     RET
203
204
```

TL/DD/11044–6

```
205                     NXTVAL:                                 ;*******************************
206                             ;SEND THE NEXT DRIVE VALUE TO STEP THE MOTOR ONE STEP IN THE
207                             ;APPROPRIATE DIRECTION (CW OR CCW)
208 0098 9DF2                   LD      A,STPPTR                ;LOAD STEP VALUE POINTER
209 009A DED4                   LD      B,#PORTGD               ;POINT TO PORT G
210 009C BD0471                 IFBIT   DIR,CMD                 ;IF CLOCKWISE
211 009F 11                     JP      IPTR                    ;THEN GO INCREMENT POINTER
212 00A0 6A             DPTR:   RBIT    CW,[B]                  ;ELSE RST CW LED
213 00A1 7B                     SBIT    CCW,[B]                 ;TURN ON CCW LED
214 00A2 8B                     DEC     A                       ;AND DECREMENT POINTER
215 00A3 92FF                   IFEQ    A,#0FF                  ;IF OFF BOTTOM OF STEPS
216 00A5 9803                   LD      A,#03                   ;THEN LOOP TO TOP OF STEPS
217 00A7 9CF2           WRVAL:  X       A,STPPTR                ;A -> STPPTR (SAVE NEW STPPTR)
218 00A9 9DF2                   LD      A,STPPTR                ;[STPPTR] -> PORTD (LOOKUP VAL)
219 00AB 9CFE                   X       A,B
220 00AD AE                     LD      A,[B]
221 00AE 9CDC                   X       A,PORTD                 ;WRITE STEP VALUE TO MOTOR
222 00B0 8E                     RET
223 00B1 6B             IPTR:   RBIT    CCW,[B]                 ;TURN OFF CCW LED
224 00B2 7A                     SBIT    CW,[B]                  ;TURN ON CW LED
225 00B3 8A                     INC     A                       ;INCREMENT THE STEP POINTER
226 00B4 9204                   IFEQ    A,#04                   ;IF OFF TOP OF STEPS
227 00B6 64                     CLR     A                       ;THEN LOOP TO BOTTOM OF STEPS
228 00B7 EF                     JP      WRVAL                   ;GO WRITE VALUE TO MOTOR
229
230
231                     ;INTERRUPT HANDLERS
232       00FF                  . = 0FF                         ;*******************************
233                             ;BRANCH TO THE APPROPRIATE INTERRUPT HANDLER
234 00FF BDEF75                 IFBIT   TPND,PSW                ;TIMER UNDERFLOW
235 0102 08                     JP      TMRINT
236 0103 BDEF73                 IFBIT   IPND,PSW                ;EXTERNAL INTERRUPT
237 0106 16                     JP      EXTINT
238 0107 BDEF78                 SBIT    GIE,PSW                 ;SOFTWARE TRAP
239 010A 8D                     RETSK
240
241                     TMRINT:                                 ;*******************************
242                             ;RESET THE TIMER INTERRUPT PENDING BIT AND STEP THE MOTOR
243 010B 9CF9                   X       A,0F9                   ;CONTEXT SAVE ROUTINE
244 010D 9DFE                   LD      A,B
245 010F 9CFA                   X       A,0FA
246 0111 BDEF6D                 RBIT    TPND,PSW                ;RESET PENDING BIT
247 0114 3098                   JSR     NXTVAL                  ;STEP THE MOTOR
248 0116 9DFA                   LD      A,0FA                   ;CONTEXT RESTORE ROUTINE
249 0118 9CFE                   X       A,B
250 011A 9DF9                   LD      A,0F9
251 011C 8F                     RETI
252
253                     EXTINT:                                 ;*******************************
254 011D BD0778                 SBIT    INT,FLGREG              ;SET INTERRUPT OCCURED FLAG
255 0120 3158                   JSR     DELAY                   ;WAIT
```

TL/DD/11044–7

```
256 0122 BDEF6B                 RBIT    IPND,PSW                ;RESET PENDING BIT
257 0125 8F                     RETI
258
259                     ;SUPPORT ROUTINES CONTINUED
260
261                     TIMVAL:                                 ;*******************************
262                             ;During continuous operation, the motor is stepped once every
263                             ;timer underflow. Therefore, a timer value is calculated that will
264                             ;produce timer underflows every X microseconds causing the motor
265                             ;to step Xsteps/second.
266                             ;For example:  To step 100 times per second.
267                             ;       microseconds/step = 1000000uS/sec x 1sec/100steps = 10000
268                             ;       10000uS/step = 02718Hex uS/step
269                             ;       1uS = one count down of the timer
270                             ; Therefore, load the timer with 02718H for 100 steps/sec.
271
272 0126 DE14                      LD      B,#TVAL0                ;POINT TO STORAGE FOR TIMVAL
273 0128 BD0474                 IFBIT   4,CMD                   ;IF LOWEST SPEED BIT SET
274 012B 17                     JP      SLOWER                  ;THEN USE SLOWEST SPEED
275 012C BD0475                 IFBIT   5,CMD                   ;IF SECOND LOWEST SPD BIT SET
276 012F 0E                     JP      SLOW                    ;THEN USE SLOW SPEED
277 0130 BD0476                 IFBIT   6,CMD                   ;IF SECOND HIGHEST SPD BIT SET
278 0133 05                     JP      FAST                    ;THEN USE FAST SPEED
279 0134 9A02           FASTER: LD      [B+],#02                ;ELSE USE FASTEST SPEED
280 0136 9E08                   LD      [B],#08                 ;400steps/sec = 2rev/sec
281 0138 8E                     RET
282 0139 9A88           FAST:   LD      [B+],#088               ;200steps/sec = 1rev/sec
283 013B 9E13                   LD      [B],#013
284 013D 8E                     RET
285 013E 9A18           SLOW:   LD      [B+],#018               ;100steps/sec = .5rev/sec
286 0140 9E27                   LD      [B],#027
287 0142 8E                     RET
288 0143 9A54           SLOWER: LD      [B+],#054               ;25steps/sec = .125rev/sec
289 0145 9E9C                   LD      [B],#09C
290 0147 8E                     RET
291
292
293                     SPEED:                                  ;*******************************
294                             ;COMPARE CURRENT MOTOR SPEED WITH DESIRED MOTOR SPEED
295 0148 3126                   JSR     TIMVAL                  ;CALCULATED DESIRED SPEED VAL
296 014A 9D14                   LD      A,TVAL0
297 014C BDEC82                 IFEQ    A,TAULO                 ;IF DESIRED LBYTE EQUALS CURRENT LBYTE
298 014F 01                     JP      TSTHI                   ;THEN GO TEST HI-BYTE
299 0150 8D                     RETSK                           ;ELSE NOT EQUAL RETURN AND SKIP
300 0151 9D15           TSTHI:  LD      A,TVAL1
301 0153 BDED82                 IFEQ    A,TAUHI                 ;IF HI-BYTE EQUALS CURRENT HI-BYTE
302 0156 8E                     RET                             ;THEN DESIRED = CURRENT RETURN
303 0157 8D                     RETSK                           ;ELSE DESIRED != CURRENT RET & SKIP
304
305                     DELAY:                                  ;*******************************
306                             ;INSERT A DELAY
```

TL/DD/11044–8

NATIONAL SEMICONDUCTOR CORPORATION
COP800 CROSS ASSEMBLER, REV: D1, 12 OCT 88

```
307 0158 D301                LD      0F3,#01              ;FOR SINGLE STEP & EXTINT DEBOUNCE
308 015A D4FF          DLY1: LD      0F4,#0FF             ;APPROX .256mS X 6
309 015C C4            DLY2: DRSZ    0F4
310 015D FE                  JP      DLY2
311 015E C3                  DRSZ    0F3
312 015F FA                  JP      DLY1
313 0160 8E                  RET
314
315                          .END
```

TL/DD/11044–9

```
B       00FE          BUSY    0002  *       CCW     0003          CHKSPD  0064
CMD     0004          CNTRL   00EE          CONT    005F          CREG0   00F0
CREG1   00F1          CW      0002          DELAY   0158          DIR     0001
DLY1    015A          DLY2    015C          DO      005A          DO2     0058
DPTR    00A0  *       ENI     0001  *       ENTI    0004          EXTINT  011D
FAST    0139          FASTER  0134  *       FLGREG  0007          GIE     0000
GO      0000          IEDG    0002  *       INT     0000          IPND    0003
IPTR    00B1          MID     0055  *       MODE    0002          MS0     0000
MS1     0001  *       MS2     0002  *       MS3     0003  *       MSEL    0003  *
NS      0005  *       NXTVAL  0098          OUT     0086          PORTD   00DC
PORTGC  00D5          PORTGD  00D4          PORTGP  00D6  *       PORTI   00D7  *
PORTLC  00D1          PORTLD  00D0          PORTLP  00D2          PSW     00EF
READY   0001          SETGO   0067          SIOR    00E9  *       SLOW    013E
SLOWER  0143          SP      00FD          SPEED   0148          SS      0004  *
SSTEP   0041          STEPS   0005  *       STOP    003B          STPPTR  00F2
TAUHI   00ED          TAULO   00EC          TC1     0007  *       TC2     0006  *
TC3     0005  *       TIMVAL  0126          TMRHI   00EB          TMRINT  010B
TMRLO   00EA          TMRSET  008E          TOP     0025          TP2     0053
TPND    0005          TRUN    0004          TSTHI   0151          TVAL0   0014
TVAL1   0015          WAIT    0081          WRVAL   00A7          X       00FC
```

TL/DD/11044–10

MACRO TABLE

```
   NO WARNING LINES

   NO ERROR LINES

  282  ROM BYTES USED

SOURCE CHECKSUM = 80C0
OBJECT CHECKSUM = 0520

INPUT   FILE C:MOTOR.MAC
LISTING FILE C:MOTOR.PRN
OBJECT  FILE C:MOTOR.LM
```

TL/DD/11044–11

LIFE SUPPORT POLICY

NATIONAL'S PRODUCTS ARE NOT AUTHORIZED FOR USE AS CRITICAL COMPONENTS IN LIFE SUPPORT DEVICES OR SYSTEMS WITHOUT THE EXPRESS WRITTEN APPROVAL OF THE PRESIDENT OF NATIONAL SEMICONDUCTOR CORPORATION. As used herein:

1. Life support devices or systems are devices or systems which, (a) are intended for surgical implant into the body, or (b) support or sustain life, and whose failure to perform, when properly used in accordance with instructions for use provided in the labeling, can be reasonably expected to result in a significant injury to the user.
2. A critical component is any component of a life support device or system whose failure to perform can be reasonably expected to cause the failure of the life support device or system, or to affect its safety or effectiveness.

National Semiconductor Corporation
2900 Semiconductor Drive
P.O. Box 58090
Santa Clara, CA 95052-8090
Tel: 1(800) 272-9959
TWX: (910) 339-9240

National Semiconductor GmbH
Livry-Gargan-Str. 10
D-82256 Fürstenfeldbruck
Germany
Tel: (81-41) 35-0
Telex: 527649
Fax: (81-41) 35-1

National Semiconductor Japan Ltd.
Sumitomo Chemical
Engineering Center
Bldg. 7F
1-7-1, Nakase, Mihama-Ku
Chiba-City,
Ciba Prefecture 261
Tel: (043) 299-2300
Fax: (043) 299-2500

National Semiconductor Hong Kong Ltd.
13th Floor, Straight Block,
Ocean Centre, 5 Canton Rd.
Tsimshatsui, Kowloon
Hong Kong
Tel: (852) 2737-1600
Fax: (852) 2736-9960

National Semiconductores Do Brazil Ltda.
Rue Deputado Lacorda Franco
120-3A
Sao Paulo-SP
Brazil 05418-000
Tel: (55-11) 212-5066
Telex: 391-1131931 NSBR BR
Fax: (55-11) 212-1181

National Semiconductor (Australia) Pty, Ltd.
Building 16
Business Park Drive
Monash Business Park
Nottinghill, Melbourne
Victoria 3168 Australia
Tel: (3) 558-9999
Fax: (3) 558-9998

IMPORTANT NOTICE

Texas Instruments Incorporated and its subsidiaries (TI) reserve the right to make corrections, modifications, enhancements, improvements, and other changes to its products and services at any time and to discontinue any product or service without notice. Customers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All products are sold subject to TI's terms and conditions of sale supplied at the time of order acknowledgment.

TI warrants performance of its hardware products to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are used to the extent TI deems necessary to support this warranty. Except where mandated by government requirements, testing of all parameters of each product is not necessarily performed.

TI assumes no liability for applications assistance or customer product design. Customers are responsible for their products and applications using TI components. To minimize the risks associated with customer products and applications, customers should provide adequate design and operating safeguards.

TI does not warrant or represent that any license, either express or implied, is granted under any TI patent right, copyright, mask work right, or other TI intellectual property right relating to any combination, machine, or process in which TI products or services are used. Information published by TI regarding third-party products or services does not constitute a license from TI to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property of the third party, or a license from TI under the patents or other intellectual property of TI.

Reproduction of TI information in TI data books or data sheets is permissible only if reproduction is without alteration and is accompanied by all associated warranties, conditions, limitations, and notices. Reproduction of this information with alteration is an unfair and deceptive business practice. TI is not responsible or liable for such altered documentation. Information of third parties may be subject to additional restrictions.

Resale of TI products or services with statements different from or beyond the parameters stated by TI for that product or service voids all express and any implied warranties for the associated TI product or service and is an unfair and deceptive business practice. TI is not responsible or liable for any such statements.

TI products are not authorized for use in safety-critical applications (such as life support) where a failure of the TI product would reasonably be expected to cause severe personal injury or death, unless officers of the parties have executed an agreement specifically governing such use. Buyers represent that they have all necessary expertise in the safety and regulatory ramifications of their applications, and acknowledge and agree that they are solely responsible for all legal, regulatory and safety-related requirements concerning their products and any use of TI products in such safety-critical applications, notwithstanding any applications-related information or support that may be provided by TI. Further, Buyers must fully indemnify TI and its representatives against any damages arising out of the use of TI products in such safety-critical applications.

TI products are neither designed nor intended for use in military/aerospace applications or environments unless the TI products are specifically designated by TI as military-grade or "enhanced plastic." Only products designated by TI as military-grade meet military specifications. Buyers acknowledge and agree that any such use of TI products which TI has not designated as military-grade is solely at the Buyer's risk, and that they are solely responsible for compliance with all legal and regulatory requirements in connection with such use.

TI products are neither designed nor intended for use in automotive applications or environments unless the specific TI products are designated by TI as compliant with ISO/TS 16949 requirements. Buyers acknowledge and agree that, if they use any non-designated products in automotive applications, TI will not be responsible for any failure to meet such requirements.

Following are URLs where you can obtain information on other Texas Instruments products and application solutions:

Products		**Applications**	
Audio	www.ti.com/audio	Communications and Telecom	www.ti.com/communications
Amplifiers	amplifier.ti.com	Computers and Peripherals	www.ti.com/computers
Data Converters	dataconverter.ti.com	Consumer Electronics	www.ti.com/consumer-apps
DLP® Products	www.dlp.com	Energy and Lighting	www.ti.com/energy
DSP	dsp.ti.com	Industrial	www.ti.com/industrial
Clocks and Timers	www.ti.com/clocks	Medical	www.ti.com/medical
Interface	interface.ti.com	Security	www.ti.com/security
Logic	logic.ti.com	Space, Avionics and Defense	www.ti.com/space-avionics-defense
Power Mgmt	power.ti.com	Transportation and Automotive	www.ti.com/automotive
Microcontrollers	microcontroller.ti.com	Video and Imaging	www.ti.com/video
RFID	www.ti-rfid.com		
OMAP Mobile Processors	www.ti.com/omap		
Wireless Connectivity	www.ti.com/wirelessconnectivity		
	TI E2E Community Home Page		e2e.ti.com

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265