COP820,COP840

AN-521 Dual Tone Multiple Frequency (DTMF)

Literature Number: SNOA782

Dual Tone Multiple Frequency (DTMF)

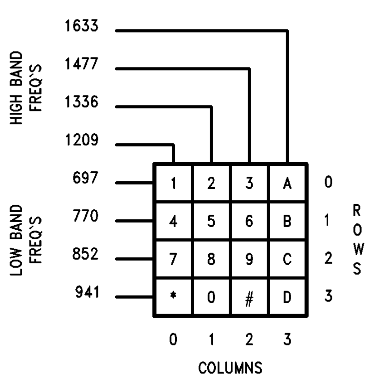
National Semiconductor
Application Note 521
Verne H. Wilson
June 1989

The DTMF (Dual Tone Multiple Frequency) application is associated with digital telephony, and provides two selected output frequencies (one high band, one low band) for a duration of 100 ms. A benchmark subroutine has been written for the COP820C/840C microcontrollers, and is outlined in detail in this application note. This DTMF subroutine takes 110 bytes of COP820C/840C code, consisting of 78 bytes of program code and 32 bytes of ROM table. The timings in this DTMF subroutine are based on a 20 MHz COP820C/840C clock, giving an instruction cycle time of 1 μs.

The matrix for selecting the high and low band frequencies associated with each key is shown in *Figure 1*. Each key is uniquely referenced by selecting one of the four low band frequencies associated with the matrix rows, coupled with selecting one of the four high band frequencies associated with the matrix columns. The low band frequencies are 697, 770, 852, and 941 Hz, while the high band frequencies are 1209, 1336, 1477, and 1633 Hz. The DTMF subroutine assumes that the key decoding is supplied as a low order hex digit in the accumulator. The COP820C/840C DTMF subroutine will then generate the selected high band and low band frequencies on port G output pins G3 and G2 respectively for a duration of 100 ms.

The COP820C/840C each contain only one timer. The problem is that three different times must be generated to satisfy the DTMF application. These three times are the periods of the two selected frequencies and the 100 ms duration period. Obviously the single timer can be used to generate any one (or possibly two) of the required times, with the program having to generate the other two (or one) times.

The solution to the DTMF problem lies in dividing the 100 ms time duration by the half periods (rounded to the nearest micro second) for each of the eight frequencies, and then examining the respective high band and low band quotients and remainders. The results of these divisions are detailed in Table I. The low band frequency quotients range from 139 to 188, while the high band quotients range from 241 to 326. The observation that only the low band quotients will each fit in a single byte dictates that the high band frequency be produced by the 16 bit (2 byte) COP820C/840C timer running in PWM (Pulse Width Modulation) Mode.



TL/DD/9662–1

FIGURE 1. DTMF Keyboard Matrix

The solution then is to use the program to produce the selected low band frequency as well as keep track of the 100 ms duration. This is achieved by using three programmed register counters R0, R2, and R3, with a backup register R1 to reload the counter R0. These three counters represent the half period, the 100 ms quotient, and the 100 ms remainder associated with each of the four low band frequencies.

The theory of operation in producing the selected low band frequency starts with loading the three counters with values obtained from a ROM table. The half period for the selected frequency is counted out, after which the G2 output bit is toggled. During this half period countout, the quotient counter is decremented. This procedure is repeated until the quotient counter counts out, after which the program branches to the remainder loop. During the remainder loop, the remainder counter counts out to terminate the 100 ms. Following the remainder countout, the G2 and G3 bits are both reset, after which the DTMF subroutine is exited. Great care must be taken in time balancing the half period loop for the selected low band frequency. Furthermore, the toggling of the G2 output bit (achieved with either a set or reset bit instruction) must also be exactly time balanced to maintain the half period time integrity. Local stall loops (consisting of a DRSZ instruction followed by a JP jump back to the DRSZ for a two byte, six instruction cycle loop) are embedded in both the half period and remainder loops. Consequently, the ROM table parameters for the half period and remainder counters are approximately only one sixth of what otherwise might be expected. The program for the half period loop, along with the detailed time balancing of the loop for each of the low band frequencies, is shown in *Figure 2*.

The DTMF subroutine makes use of two 16 byte ROM tables. The first ROM table contains the translation table for the input hex digit into the core vector. The encoding of the hex digit along with the hex digit ROM translation table is shown in Table II. The row and column bits (RR, CC) representing the low band and high band frequencies respectively of the keyboard matrix shown in *Figure 1*, are encoded in

TABLE I. Frequency Half Periods, Quotients, and Remainders

	Freq. Hz	Half Period 0.5P	Half Period in μs	100 ms/0.5P	
				Quotient	Remainder
Low Band Freq.'s	697	717.36	717	139	337
	770	649.35	649	154	54
	852	586.85	587	170	210
	941	531.35	531	188	172
High Band Freq.'s	1209	413.56	414 (256 + 158)	241	226
	1336	374.25	374 (256 + 118)	267	142
	1477	338.52	339 (256 + 83)	294	334
	1633	306.18	306 (256 + 50)	326	244

 RRD-B30M105/Printed in U. S. A.

the two upper and two lower bits of the hex digit respectively. Consequently, the format for the hex digit bits is RRCC, so that the input byte in the accumulator will consist of 0000RRCC. The program changes this value into 1101RRCC before using it in setting up the address for the hex digit ROM translation table.

The core vectors from the hex digit ROM translation table consist of a format of XX00TT00, where the two T (Timer) bits select one of four high band frequencies, while the two X bits select one of four low band frequencies. The core vector is transformed into four different inputs for the second ROM table. This transformation of the core vector is shown in Table III. The core vector transformation produces a timer vector 1100TT00 (T), and three programmed counter vectors for R1, R2, and R3. The formats for the three counter vectors are 1100XX11 (F), 1100XX10 (Q), and 1100XX01 (R) for R1, R2, and R3 respectively. These four vectors produced from the core vector are then used as inputs to the second ROM table. One of these four vectors (the T vector) is a function of the T bits from the core vector, while the other three vectors (F, Q, R) are a function of the X bits. This correlates to only one parameter being needed for the timer (representing the selected high band frequency), while three parameters are needed for the three counters (half period, 100 ms quotient, 100 ms remainder) associated with the low band frequency and 100 ms duration. The frequency parameter ROM translation table, accessed by the T, F, Q, and R vectors, is shown in Table IV.

Program			Bytes/Cycle	Conditional Cycles		Cycles	Total Cycles
	LD	B,#PORTGD	2/3				
	LD	X,#R1	2/3				
LUP1:	LD	A,[X−]	1/3			3	
	IFBIT	2,[B]	1/1			1	
	JP	BYP1	1/3	3	1		
	X	A,[X+]	1/3		3		
	SBIT	2,[B]	1/1		1		
	JP	BYP2	1/3		3		
BYP1:	NOP		1/1	1			
	RBIT	2,[B]	1/1	1			
	X	A,[X+]	1/3	3			
BYP2:	DRSZ	R2	1/3 DECREMENT			3	
	JP	LUP2	1/3 Q COUNT			3	
	JP	FINI	1/3				
LUP2:	DRSZ	R0	1/3 DECREMENT		3	3	
	JP	LUP2	1/3 F COUNT		3	1	
	NOP		1/1			1	
	LD	A,[X]	1/3			3	
	IFEQ	A,#104	2/2			2	
	JP	LUP1	1/3		1	3	31
	NOP		1/1		1		
	IFEQ	A,#93	2/2		2		
BACK:	JP	LUP1	1/3	1	3		35
	JP	BACK	1/3	3			
				3			39

Table IV Frequency	×	Stall Loop	+	Total Cycles	=	Half Period
((114 − 1)		x 6)		+ 39		= 717
((104 − 1)		x 6)		+ 31		= 649
((93 − 1)		x 6)		+ 35		= 587
((83 − 1)		x 6)		+ 39		= 531

FIGURE 2. Time Balancing for Half Period Loop

TABLE II. Hex Digit ROM Translation Table

```
            0         1         2         3
    ROW   697 Hz    770 Hz    852 Hz    941 Hz
 COLUMN  1209 Hz   1336 Hz   1477 Hz   1633 Hz

ADDRESS   DATA (HEX)     KEYBOARD
     *                               * HEX DIGIT IS RRCC,
   0xD0      000            1            WHERE R = ROW #
   0xD1      004            2              AND C = COLUMN #
   0xD2      008            3 -  -  - EXAMPLE: KEY 3 IS ROW #0,
   0xD3      00C            A            COLUMN #2, SO HEX DIGIT
   0xD4      040            4            IS 0010 = 2
   0xD5      044            5               RRCC
   0xD6      048            6
   0xD7      04C            B
   0xD8      080            7
   0xD9      084            8
   0xDA      088            9
   0xDB      08C            C
   0xDC      0C0            *
   0xDD      0C4            0
   0xDE      0C8            #
   0xDF      0CC            D
```

TABLE III. Core Vector Translation

```
CORE VECTOR   -   XX00TT00  - - - - - - - - - -
                                              *
                                             * *
                                            * * *
     TIMER VECTOR           TIMER   T     1100TT00
     HALF PERIOD VECTOR     R1      F     1100XX11
     QUOTIENT VECTOR        R2      Q     1100XX10
     REMAINDER VECTOR       R3      R     1100XX01
```

TABLE IV. Frequency Parameter ROM Translation Table

T - TIMER   F - FREQUENCY   Q - QUOTIENT   R - REMAINDER

ADDRESS	DATA (DEC)	VECTOR
0xC0	158	T
0xC1	53	R
0xC2	140	Q
0xC3	114	F
0xC4	118	T
0xC5	6	R
0xC6	155	Q
0xC7	104	F
0xC8	83	T
0xC9	32	R
0xCA	171	Q
0xCB	93	F
0xCC	50	T
0xCD	25	R
0xCE	189	Q
0xCF	83	F

In summary, the input hex digit selects one of 16 core vectors from the first ROM table. This core vector is then transformed into four other vectors (T, F, Q, R), which in turn are used to select four parameters from the second ROM table. These four parameters are used to load the timer, and the respective half period, quotient, and remainder counters. The first ROM table (representing the hex digit matrix table) is arbitrarily placed starting at ROM location 01D0, and has a reference setup with the ADD A,#0D0 instruction. The second ROM table (representing the frequency parameter table) must be placed starting at ROM location 01C0 (or 0xC0) in order to minimize program size, and has reference setups with the OR A,#0C3 instruction for the F vector and with the OR A,#0C0 instruction for the T vector.

The three parameters associated with the two X bits of the core vector require a multi-level table lookup capability with the LAID instruction. This is achieved with the following section of code in the DTMF subroutine:

```
        LD      B,#R1
        LD      X,#R4
        X       A,[X]
  LUP:  LD      A,[X]
        LAID
        X       A,[B+]
        DRSZ    R4
        IFBNE   #4
        JP      LUP
```

This program code loads the F frequency vector into R4, and then decrements the vector each time around the loop. This successive loop decrementation of the R4 vector changes the F vector into the Q vector, and then changes the Q vector into the R vector. This R4 vector is used to access the ROM table with the LAID instruction. The X pointer references the R4 vector, while the B pointer is incremented each time around the loop after it has been used to store away the three selected ROM table parameters (one per loop). These three parameters are stored in sequential RAM locations R1, R2, and R3. The IFBNE test instruction is used to skip out of the loop once the three selected ROM table parameters have been accessed and stored away.

The timer is initialized to a count of 15 so that the first timer underflow and toggling of the G3 output bit (with timer PWM mode and G3 toggle output selected) will occur at the same time as the first toggling of the G2 output bit. The half period counts for the high band frequencies range from 306 to 414, so these values minus 256 are stored in the timer section of the second ROM table. The selected value from this frequency ROM table is then stored in the lower half of the timer autoreload register, while a 1 is stored in the upper half. The timer is selected for PWM output mode and started with the instruction LD [B],#0B0 where the B pointer is selecting the CNTRL register at memory location 0EE.

The DTMF subroutine for the COP820C/840C uses 110 bytes of code, consisting of 78 bytes of program code and 32 bytes of ROM table. A program routine to sequentially call the DTMF subroutine for each of the 16 hex digit inputs is supplied with the listing for the DTMF subroutine.

```
NATIONAL SEMICONDUCTOR CORPORATION          PAGE:     1
 COP800 CROSS ASSEMBLER,REV:B,20 JAN 87
 DTMF

                ;DTMF PROGRAM FOR COP820C/840C          VERNE H. WILSON
                ;                                            5/1/89
                ;DTMF - DUAL TONE MULTIPLE FREQUENCY
                ;
                ;PROGRAM NAME: DTMF.MAC
                ;
                          .TITLE DTMF
                          .CHIP 840
                ;******* THE DTMF SUBROUTINE CONTAINS 110 BYTES *******
                ; ***** THE DTMF SUBROUTINE TIMES OUT IN 100MSEC *****
                ;    ** FROM THE FIRST TOGGLE OF THE G2/G3 OUTPUTS **
                ;      *** BASED ON A 20 MHZ COP820C/840C CLOCK ***
                ;
                ;G PORT IS USED FOR THE TWO OUTPUTS
                ;    -    HIGH BAND (HB) FREQUENCY OUTPUT ON G3
                ;    -    LOW BAND (LB) FREQUENCY OUTPUT ON G2
                ;
                ;TIMER COUNTS OUT
                ;    -    HB FREQUENCIES
                ;
                ;PROGRAM COUNTS OUT
                ;    -    LB FREQUENCIES
                ;    -    100 MSEC DIVIDED BY LB HALF PERIOD QUOTIENT
                ;    -    100 MSEC DIVIDED BY LB HALF PERIOD REMAINDER
                ;
                ;FORMAT FOR THE 16 HEX DIGIT MATRIX VECTOR IS 1101RRCC,
                ;    WHERE  -  RR IS ROW SELECT (LB FREQUENCIES)
                ;              -  CC IS COLUMN SELECT (HB FREQUENCIES)
                ;
                ;FORMAT FOR THE 16 CORE VECTORS FROM THE MATRIX SELECT
                ;    TABLE IS XX00TT00, WHERE  -  TT IS HB SELECT
                ;                                 XX IS LB SELECT
                ;
                ;FREQUENCY VECTORS (HB & LB) FOR FREQ PARAMETER TABLE
                ;    MADE FROM CORE VECTORS
                ;
                ;HB FREQUENCY VECTORS(4) END WITH 00 FOR TIMER COUNTS,
                ;    WHERE VECTOR FORMAT IS 1100TT00
                ;
                ;LB FREQUENCY VECTORS(12) END WITH:
                ;    11 FOR HALF PERIOD LOOP COUNTS,
                ;         WHERE VECTOR FORMAT IS 1100XX11
                ;    10 FOR 100 MSEC DIVIDED BY HALF PERIOD QUOTIENTS,
                ;         WHERE VECTOR FORMAT IS 1100XX10
                ;    01 FOR 100 MSEC DIVIDED BY HALF PERIOD REMAINDERS,
                ;         WHERE VECTOR FORMAT IS 1100XX01
                ;
                ;HEX DIGIT MATRIX TABLE AT HEX 01D* (OPTIONAL LOCATION,
                ;    DEPENDING ON  'ADD  A,#0D0' INST. IMMEDIATE VALUE)
                ;
                ;FREQ PARAMETER TABLE AT HEX 01C*  (REQUIRED LOCATION)
```

TL/DD/9662–2

```
NATIONAL SEMICONDUCTOR CORPORATION          PAGE:     2
 COP800 CROSS ASSEMBLER,REV:B,20 JAN 87
 DTMF

                        .FORM
                ;
                ;MAGIC:         CORE VECTOR
                ;                 XX00TT00
                ;
                ;     TIMER       T     TT00
                ;     R1          F     XX11
                ;     R2          Q     XX10
                ;     R3          R     XX01
                ;
                ;DECLARATIONS:
        00D0       PORTLD = 0D0           ; PORTL DATA REG
        00D1       PORTLC = 0D1           ; PORTL CONFIG REG
        00D4       PORTGD = 0D4           ; PORTG DATA REG
        00D5       PORTGC = 0D5           ; PORTG CONFIG REG
        00DC        PORTD = 0DC           ; PORTD REG
        00EA      TIMERLO = 0EA           ; TIMER LOW COUNTER
        00EE        CNTRL = 0EE           ; CONTROL REG
        00EF          PSW = 0EF           ; PROC STATUS WORD
        00F0           R0 = 0F0           ; LB FREQ LOOP COUNTER
        00F1           R1 = 0F1           ; LB FREQ LOOP COUNT
        00F2           R2 = 0F2           ; LB FREQ Q COUNT
        00F3           R3 = 0F3           ; LB FREQ R COUNT
        00F4           R4 = 0F4           ; LB FREQ TABLE VECTOR
                ;
   0000 DD2F    START:     LD        SP,#02F        ; HEX DIGIT MATRIX
   0002 BCD1FF             LD        PORTLC,#0FF    ; 1  2  3  A
   0005 BCD080             LD        PORTLD,#080    ; 4  5  6  B
   0008 DEDC               LD        B,#PORTD       ; 7  8  9  C
   000A 9E00               LD        [B],#0         ; *  0  #  D
   000C AE      LOOP:      LD        A,[B]          ; DTMF TEST LOOP
   000D 3160               JSR       DTMF           ; HEX MATRIX DIGIT
   000F DEDC               LD        B,#PORTD       ; TO SUBROUTINE IS
   0011 AE                 LD        A,[B]          ; OUTPUT TO PORTD
   0012 9405               ADD       A,#5           ; D0 WILL TOGGLE
   0014 A6                 X         A,[B]          ; FOR EACH CALL OF
   0015 6C                 RBIT      4,[B]          ; DTMF SUBROUTINE
   0016 9DD0               LD        A,PORTLD       ; PORTL OUTPUTS
   0018 A1                 SC                       ; PROVIDE SYNC
   0019 B0                 RRC       A              ; OUTPUT ORDER IS
   001A 9CD0               X         A,PORTLD       ; 1,5,9,D,4,8,#,A,
   001C EF                 JP        LOOP           ; 7,0,3,B,*,2,6,C
                ;
                ;
```

TL/DD/9662–3

```
     0160                  .=0160
                   ;
0160 DED5          DTMF:   LD       B,#PORTGC
0162 9B3F                  LD       [B-],#03F
0164 6B                    RBIT     3,[B]          ; OPTIONAL
0165 6A                    RBIT     2,[B]          ; OPTIONAL
                   ;
0166 94D0                  ADD      A,#0D0
0168 A4                    LAID                    ; DIGIT MATRIX TABLE
                   ;
0169 5F                    LD       B,#0
016A A6                    X        A,[B]
016B AE                    LD       A,[B]
017B 65                    SWAP     A
016C 97C3                  OR       A,#0C3
016E DEF1                  LD       B,#R1
0170 DCF4                  LD       X,#R4
0172 B6                    X        A,[X]
0173 BE            LUP:    LD       A,[X]
0174 A4                    LAID                    ; LB FREQ TABLES
0175 A2                    X        A,[B+]         ;   (3 PARAMETERS)
0176 C4                    DRSZ     R4
0177 44                    IFBNE    #4
0178 FA                    JP       LUP
                   ;
0179 5F                    LD       B,#0
017A AE                    LD       A,[B]
017C 97C0                  OR       A,#0C0
017E A4                    LAID                    ; HB FREQ TABLE
017F DEEA                  LD       B,#TIMERLO     ;   (1 PARAMETER)
0181 9A0F                  LD       [B+],#15
0183 9A00                  LD       [B+],#0
0185 A2                    X        A,[B+]
0186 9A01                  LD       [B+],#1
0188 9EB0                  LD       [B],#0B0       ; START TIMER PWM
                   ;
018A DED4                  LD       B,#PORTGD
018C DCF1                  LD       X,#R1
                   ;
018E BB            LUP1:   LD       A,[X-]
018F 72                    IFBIT    2,[B]          ; TEST LB OUTPUT
0190 03                    JP       BYP1
0191 B2                    X        A,[X+]
0192 7A                    SBIT     2,[B]          ; SET LB OUTPUT
0193 03                    JP       BYP2
0194 B8            BYP1:   NOP
0195 6A                    RBIT     2,[B]          ; RESET LB OUTPUT
0196 B2                    X        A,[X+]
0197 C2            BYP2:   DRSZ     R2             ; DECR. QUOT. COUNT
0198 01                    JP       LUP2
0199 0C                    JP       FINI           ; Q COUNT FINISHED
                   ;
019A C0            LUP2:   DRSZ     R0             ; DECR. F COUNT
019B FE                    JP       LUP2           ; LB (HALF PERIOD)
                   ;
019C B8                    NOP                     ; *************
019D BE                    LD       A,[X]          ; BALANCE
019E 9268                  IFEQ     A,#104         ; LB FREQUENCY
01A0 ED                    JP       LUP1           ; HALF PERIOD
                   ;                               ; RESIDUE
01A1 B8                    NOP                     ; DELAY FOR
01A2 925D                  IFEQ     A,#93          ; EACH OF 4
01A4 E9            BACK:   JP       LUP1           ; LB FREQ'S
01A5 FE                    JP       BACK           ; *************
                   ;
01A6 C3            FINI:   DRSZ     R3             ; DECR. REM. COUNT
01A7 FE                    JP       FINI           ; R CNT NOT FINISHED
                   ;
01A8 BDEE6C                RBIT     4,CNTRL        ; STOP TIMER
01AB 6B                    RBIT     3,[B]          ; CLR HB OUTPUT
01AC 6A                    RBIT     2,[B]          ; CLR LB OUTPUT
                   ;
01AD 8E                    RET
                   ;
```

TL/DD/9662–4

```
NATIONAL SEMICONDUCTOR CORPORATION          PAGE:      4
 COP800 CROSS ASSEMBLER,REV:B,20 JAN 87
 DTMF

                                       .FORM
                    ;
                    ;FREQUENCY AND 100MSEC PARAMETER TABLE
         01C0                .=01C0
                    ;
     01C0 9E                 .BYTE      158              ;  T
     01C1 35                 .BYTE       53              ;  R
     01C2 8C                 .BYTE      140              ;  Q
     01C3 72                 .BYTE      114              ;  F
     01C4 76                 .BYTE      118              ;  T
     01C5 06                 .BYTE        6              ;  R
     01C6 9B                 .BYTE      155              ;  Q
     01C7 68                 .BYTE      104              ;  F
     01C8 53                 .BYTE       83              ;  T
     01C9 20                 .BYTE       32              ;  R
     01CA AB                 .BYTE      171              ;  Q
     01CB 5D                 .BYTE       93              ;  F
     01CC 32                 .BYTE       50              ;  T
     01CD 19                 .BYTE       25              ;  R
     01CE BD                 .BYTE      189              ;  Q
     01CF 53                 .BYTE       83              ;  F
                    ;
                    ;DIGIT MATRIX TABLE
         01D0                .=01D0
                    ;                                          ROW   COL
     01D0 00                 .BYTE      000              ;  1    0     0
     01D1 04                 .BYTE      004              ;  2    0     1
     01D2 08                 .BYTE      008              ;  3    0     2
     01D3 0C                 .BYTE      00C              ;  A    0     3
     01D4 40                 .BYTE      040              ;  4    1     0
     01D5 44                 .BYTE      044              ;  5    1     1
     01D6 48                 .BYTE      048              ;  6    1     2
     01D7 4C                 .BYTE      04C              ;  B    1     3
     01D8 80                 .BYTE      080              ;  7    2     0
     01D9 84                 .BYTE      084              ;  8    2     1
     01DA 88                 .BYTE      088              ;  9    2     2
     01DB 8C                 .BYTE      08C              ;  C    2     3
     01DC C0                 .BYTE      0C0              ;  *    3     0
     01DD C4                 .BYTE      0C4              ;  0    3     1
     01DE C8                 .BYTE      0C8              ;  #    3     2
     01DF CC                 .BYTE      0CC              ;  D    3     3
                    ;
                             .END
```

TL/DD/9662–5

```
NATIONAL SEMICONDUCTOR CORPORATION          PAGE:     5
 COP800 CROSS ASSEMBLER,REV:B,20 JAN 87
 DTMF

SYMBOL TABLE

 B       00FE          BACK    01A4          BYP1    0194          BYP2    0197
 CNTRL   00EE          DTMF    0160          FINI    01A6          LOOP    000C
 LUP     0174          LUP1    018E          LUP2    019A          PORTD   00DC
 PORTGC  00D5          PORTGD  00D4          PORTLC  00D1          PORTLD  00D0
 PSW     00EF   *      R0      00F0          R1      00F1          R2      00F2
 R3      00F3          R4      00F4          SP      00FD          START   0000   *
 TIMERL  00EA          X       00FC


MACRO TABLE


   NO WARNING LINES

   NO ERROR LINES

  139  ROM BYTES USED

SOURCE CHECKSUM = 99A7
OBJECT CHECKSUM = 03E1

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

TL/DD/9662–6

The code listed in this App Note is available on Dial-A-Helper.

Dial-A-Helper is a service provided by the Microcontroller Applications Group. The Dial-A-Helper system provides access to an automated information storage and retrieval system that may be accessed over standard dial-up telephone lines 24 hours a day. The system capabilities include a MESSAGE SECTION (electronic mail) for communicating to and from the Microcontroller Applications Group and a FILE SECTION mode that can be used to search out and retrieve application data about NSC Microcontrollers. The minimum system requirement is a dumb terminal, 300 or 1200 baud modem, and a telephone.

With a communications package and a PC, the code detailed in this App Note can be down loaded from the FILE SECTION to disk for later use. The Dial-A-Helper telephone lines are:

Modem (408) 739-1162
Voice (408) 721-5582

For Additional Information, Please Contact Factory

Lit. # 100521

National Semiconductor Corporation
1111 West Bardin Road
Arlington, TX 76017
Tel: 1(800) 272-9959
Fax: 1(800) 737-7018

National Semiconductor Europe
Fax: (+49) 0-180-530 85 86
Email: cnjwge@tevm2.nsc.com
Deutsch Tel: (+49) 0-180-530 85 85
English Tel: (+49) 0-180-532 78 32
Français Tel: (+49) 0-180-532 93 58
Italiano Tel: (+49) 0-180-534 16 80

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 Semiconductor Japan Ltd.
Tel: 81-043-299-2309
Fax: 81-043-299-2408