LM324

AN-666 DTMF Generation with 3.58 MHz Crystal

Literature Number: SNOA148

DTMF Generation with a 3.58 MHz Crystal

National Semiconductor
Application Note 666
Verne H. Wilson
June 1990

DTMF (Dual Tone Multiple Frequency) is associated with digital telephony, and provides two selected output frequencies (one high band, one low band) for a duration of 100 ms. DTMF generation consists of selecting and combining two audio tone frequencies associated with the rows (low band frequency) and columns (high band frequency) of a pushbutton touch tone telephone keypad.

This application note outlines two different methods of DTMF generation using a COP820C/840C microcontroller clocked with a 3.58 MHz crystal in the divide by 10 mode. This yields an instruction cycle time of 2.79 μs. The application note also provides a low true row/column decoder for the DTMF keyboard.

The first method of DTMF generation provides two PWM (Pulse Width Modulation) outputs on pins G3 and G2 of the G port for 100 ms. These two PWM outputs represent the selected high band and low band frequencies respectively, and must be combined externally with an LM324 op amp or equivalent feed back circuit to produce the DTMF signal.

The second method of DTMF generation uses ROM lookup tables to simulate the two selected DTMF frequencies. These table lookup values for the selected high band and low band frequencies are then combined arithmetically. The high band frequencies contain a higher bias value to compensate for the DTMF requirement that the high band frequency component be 2 dB above the low band frequency component to compensate for losses in transmission. The resultant value from the arithmetic combination of sine wave values is output on L port pins L0 to L5, and must be combined externally with a six input resistor ladder network to produce the DTMF signal. This resultant value is updated every 118 μs. The COP820C/840C timer is used to time out the 100 ms duration of the DTMF. A timer interrupt at the end of the 100 ms is used to terminate the DTMF output. The external ladder network need not contain any active components, unlike the first method of DTMF generation with the two PWM outputs into the LM324 op amp.

The associated COP820C/840C program for the DTMF generation is organized as three subroutines. The first subroutine (KBRDEC) converts the low true column/row input from the DTMF keyboard into the associated DTMF hexadecimal digit. In turn, this hex digit provides the input for the other two subroutines (DTMFGP and DTMFLP), which represent the two different methods of DTMF generation. These three subroutines contain 35, 94, and 301 bytes of COP820C/840C code respectively, including all associated ROM tables. The Program Code/ROM table breakdowns are 19/16, 78/16, and 88/213 bytes respectively.

DTMF KEYBOARD MATRIX

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 Hz, 770 Hz, 852 Hz, and 941 Hz, while the high band frequencies are 1209 Hz, 1336 Hz, 1477 Hz, and 1633 Hz. The DTMF keyboard input decode subroutine assumes that the keyboard is encoded in a low true row/column format, where the keyboard is strobed sequentially with four low true column selects with each returning a low true row select. The low true column and row selects are encoded in the upper and lower nibbles respectively of the accumulator, which serves as the input to the DTMF keyboard input decode subroutine. The subroutine will then generate the DTMF hexadecimal digit associated with the DTMF keyboard input digit.

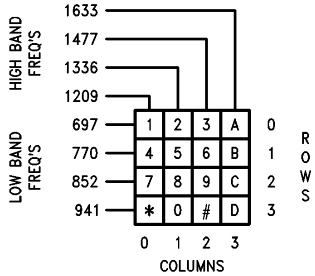
The DTMF keyboard decode subroutine (KBRDEC) utilizes a common ROM table lookup for each of the two nibbles representing the low true column and row encodings for the keyboard. The only legal low true nibbles for a single key input are E, D, B, and 7. All other low true nibble values represent multiple keys, no key, or no column strobe. Results from two legal nibble table lookups (from the same 16 byte ROM table) are combined to form a hex digit with the binary format of 0000RRCC, where RR represents the four row values and CC represents the four column values. The illegal nibbles are trapped, and the subroutine is exited with a RET (return) command to indicate multiple keys or no key. A pair of legal nibble table lookups result in the subroutine being exited with a RETSK (return and skip) command to indicate a single key input. This KBRDEC subroutine uses 35 bytes of code, consisting of 19 bytes of program code and 16 bytes of ROM table.

DTMF GENERATION USING PWM AND AN OP AMP

The first DTMF generation method (using the DTMFGP subroutine) generates the selected high band and low band frequencies as PWM (Pulse Width Modulation) outputs on pins G3 and G2 respectively of the G port. The COP820C/840C microcontrollers each contain only one timer, and three times must be generated to satisfy the DTMF application. These three times are the half periods of the two selected frequencies and the 100 ms duration period. Obviously the single timer can only generate one of the required times, while the program must generate the two remaining times. The solution lies in dividing the 100 ms duration time by the half periods for each of the eight DTMF frequencies, and then examining the respective high band and low band quotients and remainders. Naturally these divisions must be normalized to the instruction cycle time (t_C). 100 ms represents 35796 t_C's. The results of these divisions are detailed in Table I.

The four high band frequencies are produced by running the COP820C/840C timer in PWM (Pulse Width Modulation) mode, while the program produces the four low band frequencies and the 100 ms duration timeout. The programmed times are 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.

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



TL/DD/10740–22

FIGURE 1. DTMF Keyboard Matrix

TABLE I. Frequency Half Periods, Quotients and Remainders

	Freq. Hz	Half Period in μs	Half Period in t_C's	100 ms/0.5P in t_C's	
				Quotient	Remainder
Low Band Frequencies	697	717.36	257	139	73
	770	649.35	232	154	68
	852	586.85	210	170	96
	941	531.35	190	188	76
High Band Frequencies	1209	413.56	148	241	128
	1336	374.25	134	267	18
	1477	338.53	121	295	101
	1633	306.18	110	325	46

Note: 100 ms represents 35796 t_C's.

The DTMFGP subroutine starts by transforming the DTMF hex digit in the accumulator (with binary format 0000RRCC) into low and high frequency vectors with binary formats 0011RR11 and 0011CC00 respectively. The transformation of the hex digit 0000RRCC (where RR is the row select and CC is the column select) into the frequency vectors is shown in Table II. The conversion produces a timer vector 0011CC00 (T), and three programmed counter vectors for R1, R2, and R3. The formats for the three counter vectors are 0011RR11 (F), 0011RR10 (Q), and 0011RR01 (R). These four vectors created from the core vector are used as inputs for a 16 byte ROM table using the LAID (Load Accumulator InDirect) instruction. One of these four vectors (the T vector) is a function of the column bits (CC), while the other three vectors (F, Q, R) are a function of the row bits (RR). 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 III.

TABLE II. DTMF Hex Digit Translation

DTMF Hex Digit— 0000RRCC —

```
                **
              *    *
            *        *
          *            *
        *                *
```

Timer Vector	Timer	T	0011CC00
Half Period Vector	R1	F	0011RR11
100 ms Quotient Vector	R2	Q	0011RR10
100 ms Remainder Vector	R3	R	0011RR01

TABLE III. Frequency Parameter ROM Translation Table

T— Timer **F— Frequency** **Q— Quotient** **R— Remainder**

Address	**Data (Decimal)**	**Vector**
0x30	147	T
0x31	10	R
0x32	140	Q
0x33	38	F
0x34	133	T
0x35	9	R
0x36	155	Q
0x37	33	F
0x38	120	T
0x39	14	R
0x3A	171	Q
0x3B	31	F
0x3C	109	T
0x3D	10	R
0x3E	189	Q
0x3F	26	F

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*.

	Program		**Bytes/ Cycles**	**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			3	
	JP	LUP2	1/3			3	
	JP	FINI	1/3				
LUP2:	DRSZ	R0	1/3		3	3	
	JP	LUP2	1/3		3	1	
	LD	A,[X]	1/3			3	
	IFEQ	A,#31	2/2			2	
	JP	LUP1	1/3		1	3	30
	NOP		1/1		1		
	NOP		1/1		1		
	IFEQ	A,#38	2/2		2		
	JP	LUP1	1/3	1	3		35
	LAID		1/3	3			
	NOP		1/1	1			
	JP	LUP1	1/3	3			40

Table III Frequency	**Stall Loop**	**Total Cycles**	**Half Period**
[(38 − 1)	× 6]	+ 35	= 257
[(33 − 1)	× 6]	+ 40	= 232
[(31 − 1)	× 6]	+ 30	= 210
[(26 − 1)	× 6]	+ 40	= 190

FIGURE 2. Time Balancing for Half Period Loop

TABLE IV. Time Balancing for Remainder Loop

Table III Remainder	Stall Loop	R Loop Overhead	Total Cycles	Table I Remainder
[(10 − 1)	× 6]	+ 20	= 74	73
[(9 − 1)	× 6]	+ 20	= 68	68
[(14 − 1)	× 6]	+ 20	= 98	96
[(10 − 1)	× 6]	+ 20	= 74	76

Note that the Q value in Table III is one greater than the quotient in Table I to compensate for the fact that the quotient count down to zero test is performed early in the half period loop. The overhead in the remainder loop is 20 instruction cycles. The detailed time balancing for the remainder loop is shown in Table IV.

The selected high band frequency is achieved by loading the half period count in t_C's minus one (from Table III) into the timer autoreload register and running the timer in PWM output mode. The minus one is necessary since the timer toggles the G3 output bit when it underflows (counts down through zero), at which time the contents of the autoreload register are transferred into the timer.

In summary, the input digit from the keyboard (encoded in low true column/row format) is translated into a digit matrix vector XXXXRRCC which is checked for 1001RRCC to indicate a single key entry. No key or multiple key entries will set a flag and terminate the DTMF subroutine. The digit matrix vector for a single key is transformed into the core vector 0000RRCC. The core vector is then translated into four other vectors (T, F, Q, R) which in turn are used to select four parameters from a 16 byte ROM table. These four parameters are used to load the timer, and the respective half period, quotient, and remainder counters. The 16 byte ROM table must be located starting at ROM location 0030 (or 0X30) in order to minimize program size, and has reference setups with the "OR A,#033" instruction for the F vector and the "OR A,#030" instruction for the T vector.

The three parameters associated with the two R 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
LUP:    X       A,[B]
        LD      A,[B,]
        LAID
        X       A,[B+]
        DEC     A
        IFBNE   #4
        JP      LUP
```

This program loads the F frequency vector into R1, and then decrements the vector each time around the loop. The vector is successively moved with the exchange commands from R1 to R2 to R3 as one of the same exchange commands loads the data from the ROM table into R1, R2, and R3. This successive decrementation of the F vector changes the F vector into the Q vector, and then changes the Q vector into the R vector. These vectors are used to access the ROM table with the LAID instruction. 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 minus one are stored in the timer section of the ROM table. The selected value from this frequency ROM table is stored in the timer autoreload register. 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.

This first DTMF generation subroutine for the COP820C/840C uses 94 bytes of code, consisting of 78 bytes of program code and 16 bytes of ROM table. A program test routine to sequentially call the DTMFGP subroutine for each of the 16 keyboard input digits is supplied with the listing for the DTMF35 program. This test routine uses a 16 byte ROM table to supply the low true encoded column/row keyboard input to the accumulator. An input from the I0 input pin of the I port is used to select which DTMF generation subroutine is to be used. The DTMFGP subroutine is selected with I0 = 0.

A TYPICAL OP AMP CONFIGURATION FOR MIXING THE TWO DTMF PWM OUTPUTS IS SHOWN IN *FIGURE 3*.

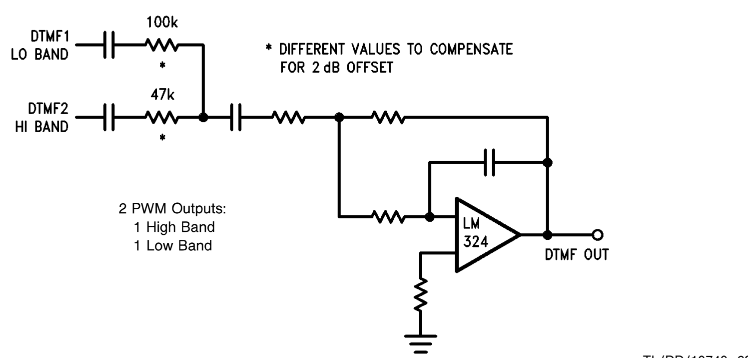


FIGURE 3. Typical Op Amp Configuration for Mixing DTMF PWM Outputs

DTMF GENERATION USING A RESISTOR LADDER NETWORK

The second DTMF generation method (using the DTMFLP subroutine) generates and combines values from two table lookups simulating the two selected sine waves. The high band frequency table values have a higher base line value (16 versus 13) than the low band frequency table values. This higher bias for the high frequency values is necessary to satisfy the DTMF requirement that the high band DTMF frequencies need a value 2 dB greater than the low band DTMF frequencies to compensate for losses in transmission.

The resultant value from arithmetically combining the table lookup low band and high band frequency values is output on pins L0 to L5 of the L port in order to feed into a six input external resistor ladder network. The resultant value is updated every 117⅓ µs (one cycle of the LUP42 program loop). The LUP42 program loop contains 42 instruction cycles (t_C's) of 2.7936511 µs each for a total loop time of 117⅓ µs. The COP820C/840C timer is used to count out the 100 ms DTMF duration time.

An interrupt from the timer terminates the 100 ms DTMF output. Note that the Stack Pointer (SP) must be adjusted following the timer interrupt before returning from the DTMFLP subroutine.

The DTMFLP subroutine starts by quadrupling the value of the DTMF hex digit value in the accumulator, and then adding an offset value to reach the first value in the telephone key table. The telephone key ROM table contains four values associated with each of the 16 DTMF hex keys. These four values represent the low and high frequency table sizes and table starting addresses associated with the pair of frequencies (one low band, one high band) associated with each DTMF key. The FRLUP section of the program loads the four associated telephone key table values from the ROM table into the registers LFTBSZ (Low Freq Table Size), LFTADR (Low Freq Table Address), HFTBSZ (High Freq Table Size), and HFTADR (High Freq Table Address). The program then initializes the timer and autoreload register, starts the timer, and then jumps to LUP42. Note that the timer value in t_C's is 100 ms plus one LUP42 time, since the initial DTMF output is not until the end of the LUP42 program.

Multiples of the magic number 118 µs (approximately) are close approximations to all eight of the DTMF frequencies. The LUP42 program uses 42 instruction cycles (of 2.7936511 µs each) to yield a LUP42 time of 117⅓ µs. The purpose of the LUP42 program is to update the six L port outputs by accessing and then combining the next set of values from the selected low band and high band sine wave frequency tables in the ROM. The ROM table offset frequency pointers (LFPTR and HFPTR) must increment each time and then wrap around from top to bottom of the two selected ROM tables. The ROM table size parameters (LFTBSZ and HFTBSZ) for the selected frequencies are tested during each LUP42 to determine if the wrap around from ROM table top to bottom is necessary. The wrap around is implemented by clearing the frequency pointer in question. Note that the ROM tables are mapped from a reference of 0 to table size minus one, so that the table size is used in a direct comparison with the frequency offset pointer to test for the need for a wrap around. Also note that the offset pointer incremented value is used during the following LUP42 cycle, while the pre-incremented value of the pointer is used during the current cycle. However, it is the incremented value that is tested versus the table size for the need to wrap around.

After the low band and high band ROM table sine wave frequency values are accessed in each cycle of the LUP42 program, they are added together and then output to pins L0–L5 of the L port. As stated previously, the low band frequency values have a lower bias than the high band frequency values to compensate for the required 2 dB offset. Specifically, the base line and maximum values for the low frequency values are 13 and 26 respectively, while the base line and maximum values for the high frequency values are 16 and 32 respectively. Thus the combined base line value is 29, while the combined maximum value is 58. This gives a range of values on the L port output (L0–L5) from 0 to 58.

The minimum time necessary for the LUP42 update program loop is 36 instruction cycles including the jump back to the start of the loop. Consequently, two LAID instructions are inserted just prior to the jump back instruction at the end of LUP42 to supply the six extra NOP instruction cycles needed to increase the LUP42 instruction cycles from 36 to 42. A three cycle LAID instruction can always be used to simulate three single cycle NOP instructions if the accumulator data is not needed.

Table V shows the multiple LUP42 approximation to the eight DTMF frequencies, including the number of sine wave cycles and data points in the approximation. As an example, three cycles of a sine wave with a total of 19 data points across the three cycles is used to approximate the 1336 Hz DTMF frequency. The 19 cycles of LUP42 times the LUP42 time of 117⅓ µs is divided into the three cycles to yield a value of 1345.69 Hz. This gives an error of +0.73% when compared with the DTMF value of 1336 Hz. This is well within the 1.5% North American DTMF error range.

TABLE V. DTMF Frequency Approximation Table

DTMF Freq.	# of Sine Wave Cycles	# of Data Points	Calculation	Approx. Freq.	% Error
697	4	49	4/(49 x 117⅓)	= 695.73	−0.18
770	1	11	1/(11 x 117⅓)	= 774.79	+0.62
852	1	10	1/(10 x 117⅓)	= 852.27	+0.03
941	1	9	1/(9 x 117⅓)	= 946.97	+0.63
1209	1	7	1/(7 x 117⅓)	= 1217.53	+0.71
1336	3	19	3/(19 x 117⅓)	= 1345.69	+0.73
1477	4	23	4/(23 x 117⅓)	= 1482.21	+0.35
1633	4	21	4/(21 x 117⅓)	= 1623.38	−0.59

The frequency approximation is equal to the number of cycles of sine wave divided by the time in the total number of LUP42 cycles before the ROM table repeats.

The values in the DTMF sine wave ROM tables are calculated by computing the sine value at the appropriate points, scaling the sine value up to the base line value, and then adding the result to the base line value. The following example will help to clarify this calculation.

Consider the three cycles of sine wave across 19 data points for the 1336 Hz high band frequency. The first value in the table is the base line value of 16. With 2π radians per sine wave cycle, the succeeding values in the table represent the sine values of $1 \times (6\pi/19)$, $2 \times (6\pi/19)$, $3 \times (6\pi/19)$, . . . , up to $18 \times (6\pi/19)$. Consider the seventh and eighth values in the table, representing the sine values of $6 \times (6\pi/19)$ and $7 \times (6\pi/19)$ respectively. The respective calculatons of $16 \times \sin[6 \times (6\pi/19)]$ and $16 \times \sin[7 \times (6\pi/19)]$ yield values of -5.20 and 9.83. Rounding to the nearest integer gives values of -5 and 10. When added to the base line value of 16, these values yield the results 11 and 26 for the seventh and eighth values in the 1336 Hz DTMF ROM table. Symmetry in the loop of 19 values in the DTMF table dictates that the fourteenth and thirteenth values in the table are 21 and 6, representing values of 5 and -10 from the calculations.

The area under a half cycle of sine wave relative to the area of the surrounding rectangle is $2/\pi$, where π radians represent the sine wave half cycle. This surrounding rectangle has a length of π and a height of 1, with the height representing the maximum sine value. Consequently, the area of the surrounding rectangle is π. The integral of the area under the half sine wave from 0 to π is equal to 2. The ratio of $2/\pi$ is equal to 63.66%, so that the total of the values for each half sine wave should approximate 63.66% of the sum of the max values. The maximum values (relative to the base line) are 13 and 16 respectively for the low and high band DTMF frequencies.

For the previous 1336 Hz example, the total of the absolute values for the 19 sine values from the 1336 Hz ROM table is equal to 196. The surrounding rectangle for the three cycles of sine wave is 19 by 16 for a total area of 304. The ratio of 196/304 is 64.47% compared with the $2/\pi$ ratio of 63.66%. Thus the sine wave approximation gives an area abundance of 0.81% (equal to $64.47 - 63.66$).

An application of the sine wave area criteria is shown in the generation of the DTMF 852 Hz frequency. The ten sine values calculated are 0, 7.64, 12.36, 12.36, 7.64, 0, -7.64, -12.36, -12.36, and -7.64. Rounding off to the nearest integer yields values of 0, 8, 12, 12, 8, 0, -8, -12, -12 and -8. The total of these values (absolute numbers) is 80, while the area of the surrounding rectangle is 130 (10 x 13). The ratio of 80/130 is 61.54% compared with the $2/\pi$ ratio of 63.66%. Thus the sine wave approximation gives an area deficiency of 2.12% (equal to $63.66 - 61.54$), which is overly deficient. Consequently, two of the ten sine values are augmented to yield sine values of 0, 8, 12, 13*, 8, 0, -8, -12, -13*, and -8. This gives an absolute total of 82 and a ratio of 82/130, which equals 63.08% and serves as a much better approximation to the $2/\pi$ ratio of 63.66%.

The sine wave area criteria is also used to modify two values in the DTMF 941 Hz frequency. The nine sine values calculated are 0, 8.36, 12.80, 11.26, 4.45, -4.45, -11.26, -12.80, and -8.36. Rounding off to the nearest integer yields values of 0, 8, 13, 11, 4, -4, -11, -13, and -8. The total of these values (absolute numbers) is 72, while the area of the surrounding rectangle is 117 (9 x 13). The ratio of 72/117 is 61.54% compared to the $2/\pi$ ratio of 63.66%. Thus the sine wave approximation gives an area deficiency of 2.12% (equal to $63.66 - 61.54$),which is overly deficient. Rounding up the two values of 4.45 and -4.45 to 5 and -5, rather than down to 4 and -4, yields values of 0, 8, 13, 11, 5, -5, -11, -13 and -8. This gives an absolute total of 74 and a ratio of 74/117, which equals 63.25% and serves as a much better approximation to the $2/\pi$ ratio of 63.66%.

With these modified values for the 852 and 941 DTMF frequencies, the area criteria ratio of $2/\pi = 63.66\%$ for the sine wave compared to the surrounding rectangle has the following values:

DTMF Freq.	Sum of Values	Rectangle Area	Percentage	Diff.
697 Hz	406	49 x 13 = 637	63.74%	+0.08%
770 Hz	92	11 x 13 = 143	64.34%	+0.68%
852 Hz	82	10 x 13 = 130	63.08%	−0.58%
941 Hz	74	9 x 13 = 117	63.25%	−0.41%
1209 Hz	72	7 x 16 = 112	64.29%	+0.63%
1336 Hz	196	19 x 16 = 304	64.47%	+0.81%
1477 Hz	232	23 x 16 = 368	63.04%	−0.62%
1633 Hz	216	21 x 16 = 336	64.29%	+0.63%

The LUP42 program loop is interrupted by the COP820C/840C timer after 100 ms of DTMF output. As stated previously, the Stack Pointer (SP) must be adjusted (incremented by 2) following the timer interrupt before returning from the DTMFLP subroutine.

This second DTMF generation subroutine for the COP820C/840C uses 301 bytes of code, consisting of 88 bytes of program code and 213 bytes of ROM table. The following is a summary of the DTMFLP subroutine code allocation.

DTMFLP Code Allocation	# of Bytes
1. Subroutine Header Code	42
2. Interrupt Code	16
3. LUP42 Code	30
4. Telephone Key Table	64
5. Sine Value Tables	149
Total	301

A program test routine to sequentially call the DTMFLP subroutine for each of the 16 DTMF keyboard input digits is supplied with the listing for the DTMF35 program. This test routine uses a 16 byte ROM table to supply the low true encoded column/row keyboard input to the accumulator. An input from the I0 pin of the I port is used to select which DTMF generation subroutine is to be used. The DTMFLP subroutine is selected with I0 = 1.

A TYPICAL RESISTOR LADDER NETWORK IS SHOWN IN *FIGURE 4*.

SUMMARY

In summary, the DTMF35 program assumes a COP820C/840C clocked with a 3.58 MHz crystal in divide by 10 mode. The DTMF35 program contains three subroutines, KBRDEC, DTMFGP, and DTMFLP. The KBRDEC subroutine is a low true DTMF keyboard decoder, while the DTMFGP and DTMFLP subroutines represent the alternative methods of DTMF generation.

The KBRDEC subroutine provides a low true decoding of the DTMF keyboard input and assumes that the keyboard input has been encoded in a low true column/row format, with the columns of the keyboard being sequentially strobed.

The DTMFGP subroutine produces two PWM (Pulse Width Modulation) outputs (representing the selected high and low band DTMF frequencies) for combination with an external op amp network (LM324 or equivalent).

The DTMFLP subroutine produces six bits of combined high band and low band DTMF frequency output for combination in an external resistor ladder network. This output represents a combined sine wave simulation of the two selected DTMF frequencies by combining values from two selected ROM tables, and updating these values every 118 μs.

The three DTMF35 subroutines contain the following number of bytes of program and ROM table memory:

Subroutine	# of Bytes of Program	# of Bytes of ROM Table	Total # of Bytes
KBRDEC	19	16	35
DTMFGP	78	16	94
DTMFLP	88	213	301

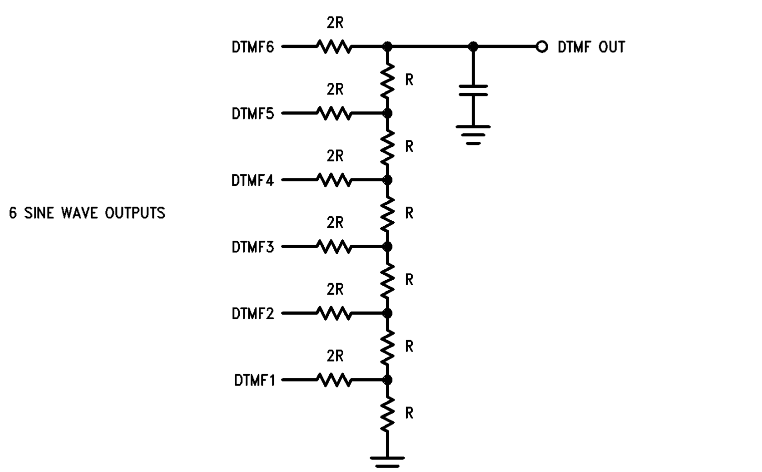


FIGURE 4. Typical Resistor Ladder Network

```
;
; DTMF GENERATION WITH A 3.58 MHZ      VERNE H. WILSON
;        CRYSTAL FOR COP820C/840C        10/28/89
;
; DTMF - DUAL TONE MULTIPLE FREQUENCY
;
; PROGRAM NAME: DTMF35.MAC
;
        .TITLE DTMF35
        .CHIP 840
;
;
; THIS DTMF PROGRAM IS BASED ON A COP820C/840C RUNNING
; WITH A CKI CLOCK OF 3.579545 MHZ (TV COLOR CRYSTAL
; FREQUENCY) IN DIVIDE BY 10 MODE, FOR AN INSTRUCTION
; CYCLE TIME OF 2.7936511 MICROSECONDS.
;
; THIS PROGRAM CONTAINS THREE SUBROUTINES, ONE FOR A
; LOW TRUE ROW/COLUMN DTMF KEYBOARD DECODING (KBRDEC),
; AND THE OTHER TWO (DTMFGP, DTMFLP) FOR ALTERNATE
; METHODS OF DTMF GENERATION.
;
; KEYBOARD INPUT DATA IS IN ACCUMULATOR WITH A
;    LOW TRUE FORMAT AS FOLLOWS:
;        BITS 7 TO 4 : LOW TRUE COLUMN VALUE (E,D,B,7)
;        BITS 3 TO 0 : LOW TRUE ROW VALUE (E,D,B,7)
;
; ASSUMPTION MADE THAT COLUMN STROBES (LOW TRUE) ARE
;    OUTPUT, WHILE ROW VALUES (LOW TRUE) ARE INPUT.
;
; THE FIRST METHOD OF DTMF GENERATION CONSISTS OF
; GENERATING TWO PWM OUTPUTS ON THE G PORT G2 AND G3
; OUTPUT PINS. THESE TWO OUTPUTS NEED TO BE MIXED
; EXTERNALLY WITH AN APPROPIATE LM324 OP AMP FEEDBACK
; CIRCUIT TO GENERATE THE DTMF.
;
; THE SECOND METHOD OF DTMF GENERATION USES ROM LOOKUP
; TABLES TO SIMULATE THE TWO DTMF SINE WAVES AND
; COMBINES THEM ARITHMETICALLY. THE RESULT IS OUTPUT ON
; THE LOWER SIX BITS OF THE L PORT (L0 - L5). THESE SIX
; OUTPUTS ARE COMBINED EXTERNALLY WITH A LADDER NETWORK
; TO GENERATE THE DTMF.
;
; THE SECOND DTMF GENERATION METHOD USES APPROXIMATELY
; THREE TIMES AS MUCH ROM CODE (INCLUDING PROGRAM CODE
; AND ROM TABLES) AS THE FIRST METHOD, BUT HAS THE
; ADVANTAGE OF ELIMINATING THE COST OF THE EXTERNAL
; ACTIVE COMPONENT (LM324 OR EQUIVALENT).
;
; BOTH OF THE DTMF SUBROUTINES GENERATE THEIR OUTPUTS
; FOR A PERIOD OF 100 MILLISECONDS.
```

TL/DD/10740–1

```
                    ;
                    ; DECLARATIONS:
                    ;
     0000                     KDATA  = 0            ; *** KEYBOARD DATA ***
     00D0                     PORTLD = 0D0          ; PORTL DATA REG
     00D1                     PORTLC = 0D1          ; PORTL CONFIG REG
     00D4                     PORTGD = 0D4          ; PORTG DATA REG
     00D5                     PORTGC = 0D5          ; PORTG CONFIG REG
     00D7                      PORTI = 0D7          ; PORTI INPUT PINS
     00DC                      PORTD = 0DC          ; PORTD REG
     00EA                      TMRLO = 0EA          ; TIMER LOW COUNTER
     00EB                      TMRHI = 0EB          ; TIMER HIGH COUNTER
     00EC                      TAULO = 0EC          ; TMR AUTORELOAD REG LO
     00ED                      TAUHI = 0ED          ; TMR AUTORELOAD REG HI
     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
                    ;
0000 DD2F           START:    LD        SP,#02F     ; INITIALIZE STACK PTR
                    ;
                    ;                            KEYBOARD HEX DIGIT MATRIX
                    ;                                     1  2  3  A
0002 DEDC                     LD        B,#PORTD  ;       4  5  6  B
0004 9E00                     LD        [B],#0    ;       7  8  9  C
0006 A0             LOOP:     RC                  ;       *  0  #  D
0007 AE                       LD        A,[B]     ; DTMF TEST LOOP
0008 9405                     ADD       A,#5      ; SEQUENCE IS 1,5,9,D,4,
000A A6                       X         A,[B]     ; 8,#,A,7,0,3,B,*,2,6,C
000B 6C                       RBIT      4,[B]     ; HEX MATRIX TO LOOKUP
000C 9420                     ADD       A,#020    ;    TABLE FOR LOW TRUE
000E A4                       LAID                ;    COLUMN/ROW INPUT TO
000F 3210                     JSR       KBRDEC    ;    KBRDEC SUBROUTINE
0011 A1                       SC                  ; SET C IF NOT SINGLE KEY
0012 DED7                     LD        B,#PORTI  ; TEST BIT 0 OF PORTI TO
0014 70                       IFBIT     0,[B]     ;    DETERMINE WHICH
0015 03                       JP        BYPA      ;    DTMF SUBROUTIINE
0016 3040                     JSR       DTMFGP    ; TWO PWM OUTPUTS ON
0018 02                       JP        BYPB      ;    G PORT PINS G2,G3
0019 308E           BYPA:     JSR       DTMFLP    ; SIX LADDER OUTPUTS ON
                                                  ;    L PORT PINS L0 - L5
001B DEDC           BYPB:     LD        B,#PORTD  ; D0 WILL TOGGLE FOR EACH
001D E8                       JP        LOOP      ;    CALL OF SUBROUTINE
                    ;
                    ;
                    ;
```

TL/DD/10740–2

```
                                     .FORM
                        ;
                        ; KEYBOARD DIGIT MATRIX TABLE
                        ;
        0020                         . = 020
                        ;
                        ;                  1   5   9   D   4   8   #   A
0020 EE                        .BYTE    0EE,0DD,0BB,077,0ED,0DB,0B7,07E
0021 DD
0022 BB
0023 77
0024 ED
0025 DB
0026 B7
0027 7E
                        ;                  7   0   3   B   *   2   6   C
0028 EB                        .BYTE    0EB,0D7,0BE,07D,0E7,0DE,0BD,07B
0029 D7
002A BE
002B 7D
002C E7
002D DE
002E BD
002F 7B
                        ;
                        ;
                        ;
                        ;
                        ;
                        ; FIRST DTMF SUBROUTINE (DTMFGP) PRODUCES TWO PWM
                        ;     (PULSE WIDTH MODULATION) OUTPUTS ON PINS G3, G2
                        ;
                        ;
                        ; 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
                        ;
                        ; NOTE THAT ALL COUNTS MUST BE NORMALIZED TO THE
                        ;     2.7936511 MICROSECOND INSTRUCTION CYCLE Tc
                        ;
                        ; 100 MSEC REPRESENTS 35796 Tc's
                        ;
                        ;
```

TL/DD/10740–3

```
;
;
; HALF PERIODS FOR THE 8 DTMF FREQUENCIES (697,770,852,
;     941,1209,1336,1477, AND 1633 KHZ) ARE 257,232,
;     210,190,148,134,121, AND 110 Tc's RESPECTIVELY
;
; THE 100 MSEC DIVIDED BY HALF PERIOD QUOTIENTS ARE
;     139,154,170,188,241,267,295, AND 325 RESPECTIVELY
;
; THE 100 MSEC DIVIDED BY HALF PERIOD REMAINDERS ARE
;     72,67,95,75,127,17,100, AND 45 RESPECTIVELY
;
;
;
;
; BINARY FORMAT FOR THE HEX DIGIT KEY VALUE FROM THE
;    KBRDEC SUBROUTINE IS 0000RRCC,
;       WHERE  -  RR IS ROW SELECT (LB FREQUENCIES)
;              -  CC IS COLUMN SELECT (HB FREQUENCIES)
;
; FREQUENCY VECTORS (HB & LB) FOR FREQ PARAMETER TABLE
;    MADE FROM KEY VALUE
;
; HB FREQ VECTORS (4) END WITH 00 FOR TIMER COUNTS,
;    WHERE VECTOR FORMAT IS 0011CC00
;
; LB FREQUENCY VECTORS (12) END WITH:
;    11 FOR HALF PERIOD LOOP COUNTS,
;         WHERE VECTOR FORMAT IS 0011RR11
;    10 FOR 100 MSEC DIVIDED BY HALF PERIOD QUOTIENTS,
;         WHERE VECTOR FORMAT IS 0011RR10
;    01 FOR 100 MSEC DIVIDED BY HALF PERIOD REMAINDERS,
;         WHERE VECTOR FORMAT IS 0011RR01
;
; FREQ PARAMETER TABLE AT HEX 003*  (REQUIRED LOCATION)
;
;
;
;              KEY VALUE
;               0000RRCC
;
;    TIMER     T    CC00
;    R1        F    RR11
;    R2        Q    RR10
;    R3        R    RR01
;
;
;
```

TL/DD/10740–4

```
                               .FORM
                    ;
                    ;
                    ;FREQUENCY AND 100 MSEC PARAMETER TABLE
                    ;
0030 93                      .BYTE     147             ; T
0031 0A                      .BYTE      10             ; R
0032 8C                      .BYTE     140             ; Q
0033 26                      .BYTE      38             ; F
0034 85                      .BYTE     133             ; T
0035 09                      .BYTE       9             ; R
0036 9B                      .BYTE     155             ; Q
0037 21                      .BYTE      33             ; F
0038 78                      .BYTE     120             ; T
0039 0E                      .BYTE      14             ; R
003A AB                      .BYTE     171             ; Q
003B 1F                      .BYTE      31             ; F
003C 6D                      .BYTE     109             ; T
003D 0A                      .BYTE      10             ; R
003E BD                      .BYTE     189             ; Q
003F 1A                      .BYTE      26             ; F
                    ;
                    ;
                    ;
0040 DED5           DTMFGP:  LD        B,#PORTGC    ; CONFIGURE G PORT
0042 9B3F                    LD        [B-],#03F    ;    FOR OUTPUTS
0044 6B                      RBIT      3,[B]        ; OPTIONAL HB RESET
0045 6A                      RBIT      2,[B]        ; OPTIONAL LB RESET
0046 5F                      LD        B,#KDATA
0047 A6                      X         A,[B]        ; STORE KEY VALUE
0048 AE                      LD        A,[B]        ; KEY VALUE TO ACC
0049 9733                    OR        A,#033       ; CREATE LB FREQ VECTOR
004B DEF1                    LD        B,#R1        ;    FROM KEY VALUE
004D A6             LUP:     X         A,[B]
004E AE                      LD        A,[B]        ; THREE PARAMETERS
004F A4                      LAID                   ;    FROM LOW BAND
0050 A2                      X         A,[B+]       ;    FREQ ROM TABLE
0051 8B                      DEC       A            ;    TO R1,R2,R3
0052 44                      IFBNE     #4
0053 F9                      JP        LUP
0054 5F                      LD        B,#KDATA
0055 AE                      LD        A,[B]        ; KEY VALUE TO ACC
0056 65                      SWAP      A            ; CREATE HB FREQ VECTOR
0057 A0                      RC                     ;    FROM KEY VALUE
0058 B0                      RRC       A
0059 B0                      RRC       A
005A 9730                    OR        A,#030
005C A4                      LAID                   ; HB FREQ TABLE
005D DEEA                    LD        B,#TMRLO     ;   (1 PARAMETER)
005F 9A0F                    LD        [B+],#15     ; INSTRUCTION CYCLE
0061 9A00                    LD        [B+],#0      ;   TIME UNTIL TOGGLE
```

TL/DD/10740–5

```
0063 A2                  X       A,[B+]      ; HB FREQ PARAMETER TO
0064 9A00                LD      [B+],#0     ;   AUTORELOAD REGISTER
0066 9EB0                LD      [B],#0B0    ; START TIMER PWM
0068 DED4                LD      B,#PORTGD
006A DCF1                LD      X,#R1
006C BB        LUP1:     LD      A,[X-]
006D 72                  IFBIT   2,[B]       ; TEST LB OUTPUT
006E 03                  JP      BYP1
006F B2                  X       A,[X+]
0070 7A                  SBIT    2,[B]       ; SET LB OUTPUT
0071 03                  JP      BYP2
0072 B8        BYP1:     NOP
0073 6A                  RBIT    2,[B]       ; RESET LB OUTPUT
0074 B2                  X       A,[X+]
0075 C2        BYP2:     DRSZ    R2          ; DECR. QUOT. COUNT
0076 01                  JP      LUP2
0077 0E                  JP      FINI        ; Q COUNT FINISHED
0078 C0        LUP2:     DRSZ    R0          ; DECR. F COUNT
0079 FE                  JP      LUP2        ; LB (HALF PERIOD)
               ;
007A BE                  LD      A,[X]       ; ******************
007B 921F                IFEQ    A,#31       ; BALANCE        ***
007D EE                  JP      LUP1        ; LOW BAND       ***
007E B8                  NOP                 ; FREQUENCY      ***
007F B8                  NOP                 ; RESIDUE        ***
0080 9226                IFEQ    A,#38       ; DELAY FOR      ***
0082 E9                  JP      LUP1        ; EACH OF THE    ***
0083 A4                  LAID                ; FOUR LOW BAND  ***
0084 B8                  NOP                 ; FREQUENCIES    ***
0085 E6                  JP      LUP1        ; ******************
0086 C3        FINI:     DRSZ    R3        ; DECR. REMAINDER COUNT
0087 FE                  JP      FINI      ; REM. COUNT NOT FINISHED
0088 BDEE6C              RBIT    4,CNTRL   ; STOP TIMER
008B 6B                  RBIT    3,[B]     ; OPTIONAL CLR HB OUTPUT
008C 6A                  RBIT    2,[B]     ; OPTIONAL CLR LB OUTPUT
008D 8E                  RET               ; RETURN FROM SUBROUTINE
               ;
               ;
               ;
```

TL/DD/10740–6

```
                        .FORM
            ;
            ; SECOND DTMF SUBROUTINE (DTMFLP) PRODUCES SIX
            ;     COMBINED LOW BAND AND HIGH BAND FREQUENCY
            ;     SINE WAVE OUTPUTS ON PINS L0 - L5
            ;
            ; SIX L PORT OUTPUTS (L0 - L5) FEED INTO AN EXTERNAL
            ; RESISTOR LADDER NETWORK TO CREATE THE DTMF OUTPUT.
            ;
            ; FOUR VALUES FROM A KEYBOARD ROM TABLE ARE LOADED
            ; INTO LFTBSZ (LOW FREQ TABLE SIZE), LFTADR (LOW
            ; FREQ TABLE ADDRESS), HFTBSZ (HIGH FREQ TABLE SIZE),
            ; AND HFTADR (HIGH FREQ TABLE ADDRESS).
            ;
            ; LUP42 USES THE LFPTR (LOW FREQ POINTER) AND HFPTR
            ; (HIGH FREQ POINTER) TO ACCESS THE SINE DATA TABLES
            ; FOR THE SELECTED FREQUENCIES ONCE PER LOOP. THESE
            ; POINTERS ARE BOTH INCREMENTED ONCE PER LUP42.
            ;
            ; LUP42 PROGRAM LOOP UPDATES THE OUTPUT VALUE EVERY
            ; 117 1/3 uSEC BY SELECTING AND THEN COMBINING NEW
            ; VALUES FROM THE SELECTED LOW BAND AND HIGH BAND
            ; FREQUENCY ROM TABLES WHICH SIMULATE THE SINE WAVES
            ; FOR THE TWO FREQUENCIES.
            ;
            ; MULTIPLES OF THE MAGIC NUMBER OF APPROXIMATELY
            ; 118 uSEC ARE CLOSE APPROXIMATIONS TO ALL EIGHT OF
            ; THE DTMF FREQUENCIES.
            ;
            ; COP820C/840C TIMER USED TO INTERRUPT THE DTMF LUP42
            ; PROGRAM LOOP AFTER 100 MSEC TO FINISH THE DTMF
            ; OUTPUT AND RETURN FROM THE DTMFLP SUBROUTINE. NOTE
            ; THAT THE STACK POINTER (SP) MUST BE ADJUSTED AFTER
            ; THE INTERRUPT BEFORE RETURNING FROM THE SUBROUTINE.
            ;
            ;
            ;
            ;
            ;
            ; DECLARATIONS:
            ;
0005                    LFPTR  =  05          ; LOW FREQ POINTER
0006                    TEMP   =  06          ; TEMPORARY
0007                    HFPTR  =  07          ; HIGH FREQ POINTER
0008                    LFTBSZ =  08          ; LO FREQ TABLE SIZE
0009                    LFTADR =  09          ; LO FREQ TABLE ADDR
000A                    HFTBSZ =  0A          ; HI FREQ TABLE SIZE
000B                    HFTADR =  0B          ; HI FREQ TABLE ADDR
            ;
0004                    TRUN   =  04
            ;
```

TL/DD/10740–7

```
                    ;
008E BCD1FF         DTMFLP:  LD       PORTLC,#0FF     ; INITIALIZE PORT L
0091 BCD01D                  LD       PORTLD,#29      ;    FOR NO TONE OUT
0094 BC0500                  LD       LFPTR,#0        ; INITIALIZE OFFSET
0097 58                      LD       B,#HFPTR        ;    POINTERS FOR
0098 9A00                    LD       [B+],#0         ;    DTMF SINE WAVE
009A A0                      RC                       ;    TABLE LOOKUP
009B 65                      SWAP     A               ; QUADRUPLE KEY
009C B0                      RRC      A               ;    VALUE AND ADD
009D B0                      RRC      A               ;    OFFSET FOR KEY
009E 94B8                    ADD      A,#0B8          ;    TABLE LOOKUP
00A0 A6             FRLUP:   X        A,[B]           ; LOAD FOUR VALUES
00A1 AE                      LD       A,[B]           ;    FROM ROM KEY
00A2 A4                      LAID                     ;    TABLE INTO LOW
00A3 A2                      X        A,[B+]          ;    FREQ LFTBSZ,
00A4 8A                      INC      A               ;    LFTADR, AND HI
00A5 4C                      IFBNE    #0C             ;    FREQ HFTBSZ,
00A6 F9                      JP       FRLUP           ;    HFTADR
00A7 DEEA                    LD       B,#TMRLO        ; INITIALIZE TIMER
00A9 9A00                    LD       [B+],#0         ;    WITH A tC COUNT
00AB 9A8C                    LD       [B+],#140       ;    EQUIVALENT TO
00AD 9A00                    LD       [B+],#0         ;    100 MSEC PLUS
00AF 9A8C                    LD       [B+],#140       ;    A LUP42 TIME
00B1 9A80                    LD       [B+],#080       ; TIMER PWM, NO OUT
00B3 9B11                    LD       [B-],#011       ; ENABLE TMR INTRPT
00B5 7C                      SBIT     TRUN,[B]        ; START TIMER
00B6 210F                    JMP      LUP42
                    ;
                    ;
                    ;
                    ;
                    ; TELEPHONE KEY TABLE:
                    ;
                    ;    TABLE FORMAT:
                    ;         PARAMETER 1:  # OF LOW FREQ TABLE VALUES
                    ;         PARAMETER 2:  BASE ADDR. OF LOW FREQ VALUES
                    ;         PARAMETER 3:  # OF HIGH FREQ TABLE VALUES
                    ;         PARAMETER 4:  BASE ADDR. OF HIGH FREQ VALUES
                    ;
                    ; KEY 1
00B8 31                      .BYTE    49,02D,7,07C
00B9 2D
00BA 07
00BB 7C
                    ;
                    ; KEY 2
00BC 31                      .BYTE    49,02D,19,083
00BD 2D
00BE 13
00BF 83
                    ;
```

TL/DD/10740-8

```
371                       ; KEY 3
372 00C0 31                          .BYTE      49,02D,23,096
    00C1 2D
    00C2 17
    00C3 96
373                       ;
374                       ; KEY A
375 00C4 31                          .BYTE      49,02D,21,0AD
    00C5 2D
    00C6 15
    00C7 AD
376                       ;
377                       ; KEY 4
378 00C8 0B                          .BYTE      11,05E,7,07C
    00C9 5E
    00CA 07
    00CB 7C
379                       ;
380                       ; KEY 5
381 00CC 0B                          .BYTE      11,05E,19,083
    00CD 5E
    00CE 13
    00CF 83
382                       ;
383                       ; KEY 6
384 00D0 0B                          .BYTE      11,05E,23,096
    00D1 5E
    00D2 17
    00D3 96
385                       ;
386                       ; KEY B
387 00D4 0B                          .BYTE      11,05E,21,0AD
    00D5 5E
    00D6 15
    00D7 AD
388                       ;
389                       ; KEY 7
390 00D8 0A                          .BYTE      10,069,7,07C
    00D9 69
    00DA 07
    00DB 7C
391                       ;
392                       ; KEY 8
393 00DC 0A                          .BYTE      10,069,19,083
    00DD 69
    00DE 13
    00DF 83
394                       ;
395                       ; KEY 9
396 00E0 0A                          .BYTE      10,069,23,096
    00E1 69
```

TL/DD/10740–9

```
     00E2 17
     00E3 96
                           ;
                           ; KEY C
     00E4 0A                          .BYTE     10,069,21,0AD
     00E5 69
     00E6 15
     00E7 AD
                           ;
                           ; KEY *
     00E8 09                          .BYTE     9,073,7,083
     00E9 73
     00EA 07
     00EB 83
                           ;
                           ; KEY 0
     00EC 09                          .BYTE     9,073,19,07C
     00ED 73
     00EE 13
     00EF 7C
                           ;
                           ; KEY #
     00F0 09                          .BYTE     9,073,23,096
     00F1 73
     00F2 17
     00F3 96
                           ;
                           ; KEY D
     00F4 09                          .BYTE     9,073,21,0AD
     00F5 73
     00F6 15
     00F7 AD
                           ;
                           ;
                           ;
                           ;
          00FF                        .=00FF
                           ;
     00FF BCD01D           INTRPT:    LD        PORTLD,#29      ; BASE LINE VALUE
     0102 DEEF                        LD        B,#PSW          ; 100 MSEC INTERRUPT
     0104 9B00                        LD        [B-],#0         ;    FROM TIMER
     0106 9E00                        LD        [B],#0          ; CLR PSW AND CNTRL
     0108 DEFD                        LD        B,#SP           ; RESTORE STACK
     010A AE                          LD        A,[B]           ;    POINTER (SP)
     010B 8A                          INC       A               ;    TO ITS VALUE
     010C 8A                          INC       A               ;    BEFORE THE
     010D A6                          X         A,[B]           ;    INTERRUPT
     010E 8E                          RET                       ; RETURN FROM
                                                                ;    SUBROUTINE
                           ;
                           ;
```

TL/DD/10740–10

```
                          .FORM
                 ;
                 ; LUP42 CONSISTS OF 42 COP840C INSTRUCTION CYCLE TIMES
                 ; LUP42 TIMING LOOP IS 42 / 0.3579545 = 117 1/3 uSEC
                 ;
                 ;
010F 5A          LUP42:   LD      B,#LFPTR
0110 AE                   LD      A,[B]          ; INCREMENT LOW FREQ
0111 8A                   INC     A              ;    OFFSET POINTER
0112 57                   LD      B,#LFTBSZ      ; TEST IF LFPTR
0113 82                   IFEQ    A,[B]          ;    BEYOND LIMIT
0114 64                   CLR     A              ; REINITIALIZE LFPTR
0115 5A                   LD      B,#LFPTR       ;    FOR NEXT TIME
0116 A6                   X       A,[B]
0117 56                   LD      B,#LFTADR      ; ADD PTR TO LO FREQ
0118 84                   ADD     A,[B]          ;    TABLE ADDRESS
0119 A4                   LAID                   ; LOW FREQ COMPONENT
011A 59                   LD      B,#TEMP        ; RESULT TO TEMP
011B A2                   X       A,[B+]
011C AE                   LD      A,[B]          ; INCREMENT HI FREQ
011D 8A                   INC     A              ;    OFFSET POINTER
011E 55                   LD      B,#HFTBSZ      ; TEST IF HFPTR
011F 82                   IFEQ    A,[B]          ;    BEYOND LIMIT
0120 64                   CLR     A              ; REINITIALIZE HFPTR
0121 58                   LD      B,#HFPTR       ;    FOR NEXT TIME
0122 A6                   X       A,[B]
0123 54                   LD      B,#HFTADR      ; ADD PTR TO HI FREQ
0124 84                   ADD     A,[B]          ;    TABLE ADDRESS
0125 A4                   LAID                   ; HI FREQ COMPONENT
0126 59                   LD      B,#TEMP        ; ADD LOW FREQ VALUE
0127 84                   ADD     A,[B]          ;   TO HI FREQ VALUE
0128 9CD0                 X       A,PORTLD       ; RESULT TO PORT L
012A A4                   LAID                   ; EQUIVALENT OF
012B A4                   LAID                   ;    SIX NOP'S
012C E2                   JP      LUP42          ; TIMING LOOP OF
                                                 ;    117 1/3 uSEC
                 ;
                 ;
                 ;
                 ;
```

TL/DD/10740–11

```
                .FORM
        ;
        ; THE FREQUENCY APPROXIMATION IS EQUAL TO THE NUMBER OF
        ; CYCLES OF SINE WAVE DIVIDED BY THE TIME IN THE TOTAL
        ; NUMBER OF LUP42 CYCLES BEFORE THE REPETITION OF THE
        ; ROM TABLE. AS AN EXAMPLE, CONSIDER THE THREE CYCLES
        ; OF SINE WAVE AND 19 VALUES IN THE ASSOCIATED 1336 HZ
        ; ROM TABLE. THE 19 CYCLES OF LUP42 TIMES THE LUP42
        ; TIME OF 117 1/3 uSEC IS DIVIDED INTO THE THREE CYCLES
        ; OF SINE WAVE TO YIELD A VALUE OF 1345.69 HZ AS THE
        ; 1336 HZ APPROXIMATION.
        ;
        ; THE VALUES IN THE ROM TABLES FOR THE DTMF SINE WAVES
        ; SHOULD WRAP AROUND END TO END IN EITHER DIRECTION TO
        ; FORM A SYMETRICAL LOOP. THE FIRST VALUE IN THE ROM
        ; TABLE REPRESENTS THE BASE LINE FOR THAT FREQUENCY.
        ;
        ; THE HIGH BAND DTMF FREQUENCIES HAVE A BASE LINE VALUE
        ; OF 16 AND A MAXIMUM VALUE OF 32. THE LOW BAND DTMF
        ; FREQUENCIES HAVE A BASE LINE VALUE OF 13 AND A
        ; MAXIMUM VALUE OF 26. THIS DIFFERENCE IN BASE LINE
        ; VALUES IS NECESSARY TO SATISFY THE REQUIREMENT OF THE
        ; HIGH BAND FREQUENCIES NEEDING A LEVEL 2 dB ABOVE THE
        ; LEVEL OF THE LOW BAND FREQUENCIES TO COMPENSATE FOR
        ; LOSSES IN TRANSMISSION. THE SUM OF THE TWO BASE LINE
        ; VALUES YIELDS A BASE LINE VALUE OF 29, WHILE THE SUM
        ; OF THE TWO MAXIMUM VALUES YIELDS A MAXIMUM VALUE OF
        ; 58. THUS THE SIX BIT DTMF OUTPUT FROM THE L PORT TO
        ; THE LADDER NETWORK RANGES FROM 0 TO 58, WITH A BASE
        ; LINE VALUE OF 29.
        ;
        ; THE VALUES IN THE DTMF SINE WAVE TABLES ARE
        ; CALCULATED BY COMPUTING THE SINE VALUE AT THE
        ; APPROPIATE POINTS, SCALING THE SINE VALUE UP TO THE
        ; BASE LINE VALUE, AND THEN ADDING THE RESULT TO THE
        ; BASE LINE VALUE. THE FOLLOWING EXAMPLE WILL HELP TO
        ; CLARIFY THIS CALCULATION.
        ;
        ; CONSIDER THE THREE CYCLES OF SINE WAVE ACROSS 19
        ; DATA POINTS FOR THE 1336 HZ DTMF HIGH BAND FREQUENCY.
        ; THE FIRST VALUE IN THE TABLE IS THE BASE LINE VALUE
        ; OF 16. WITH 2 PI RADIANS PER SINE WAVE CYCLE,
        ; THE SUCCEEDING VALUES IN THE TABLE REPRESENT THE
        ; SINE VALUES OF 1 X (6 PI / 19), 2 X (6 PI / 19),
        ; 3 X (6 PI / 19), . . . . , UP TO 18 X (6 PI / 19).
        ; LET US NOW CONSIDER THE SEVENTH AND EIGHTH VALUES
        ; IN THE TABLE, REPRESENTING THE SINE VALUES OF
        ; 6 X (6 PI / 19) AND 7 X (6 PI / 19) RESPECTIVELY.
        ; THE CALCULATIONS OF 16 X SIN [6 X (6 PI / 19)] AND
        ; 16 X SIN [7 X (6 PI / 19)] YIELD VALUES OF - 5.20 AND
        ; 9.83 RESPECTIVELY. ROUNDED TO THE NEAREST INTEGER
```

TL/DD/10740–12

```
                          ; GIVES VALUES OF - 5 AND 10. WHEN ADDED TO THE BASE
                          ; LINE VALUE OF 16, THESE VALUES YIELD THE RESULTS
                          ; 11 AND 26 FOR THE SEVENTH AND EIGHTH VALUES IN THE
                          ; 1336 HZ DTMF TABLE. SYMMETRY IN THE LOOP OF 19 VALUES
                          ; IN THE DTMF TABLE DICTATES THAT THE FOURTEENTH AND
                          ; THIRTEENTH VALUES IN THE TABLE ARE 21 AND 6,
                          ; REPRESENTING VALUES OF 5 AND - 10 FROM THE
                          ; CALCULATIONS.
                          ;
                          ; THE AREA UNDER A HALF CYCLE OF SINE WAVE RELATIVE TO
                          ; THE AREA OF THE SURROUNDING RECTANGLE IS 2/PI, WHERE
                          ; PI RADIANS REPRESENT THE SINE WAVE HALF CYCLE. THIS
                          ; SURROUNDING RECTANGLE HAS A LENGTH OF PI AND A HEIGHT
                          ; OF 1, WITH THE HEIGHT REPRESENTING THE MAXIMUM SINE
                          ; VALUE. CONSEQUENTLY, THE AREA OF THIS SURROUNDING
                          ; RECTANGLE IS PI. THE INTEGRAL OF THE AREA UNDER THE
                          ; HALF SINE WAVE FROM 0 TO PI IS EQUAL TO 2. THE RATIO
                          ; OF 2/PI IS EQUAL TO 63.66 % , SO THAT THE TOTAL OF
                          ; THE VALUES FOR EACH HALF SINE WAVE SHOULD APPROXIMATE
                          ; 63.66 % OF THE SUM OF THE MAX VALUES. THE MAXIMUM
                          ; VALUES (RELATIVE TO THE BASE LINE) ARE 13 AND 16
                          ; RESPECTIVELY, FOR THE LOW AND HIGH BAND FREQUENCIES.
                          ;
                          ;
                          ;
                          ;
                          ;
                          ; LF697:   4 CYCLES OF SINE WAVE SPREAD
                          ;                  ACROSS 49 TIMING LOOP (LUP42) CYCLES
                          ;
                          ;          FREQ. = 4 / (49 X 117 1/3) = 695.73 HZ
                          ;          ERROR = (697 - 695.73) / 697  = - 0.18 %
                          ;
012D 0D                            .BYTE     13,19,24,26,25,20,14,7,2,0
012E 13
012F 18
0130 1A
0131 19
0132 14
0133 0E
0134 07
0135 02
0136 00
0137 01                            .BYTE     1,5,11,18,23,26,25,21,15,9
0138 05
0139 0B
013A 12
013B 17
013C 1A
013D 19
013E 15
```

TL/DD/10740–13

```
0140 09
0141 03                .BYTE    3,0,1,4,10,16,22,25,26,23
0142 00
0143 01
0144 04
0145 0A
0146 10
0147 16
0148 19
0149 1A
014A 17
014B 11                .BYTE    17,11,5,1,0,3,8,15,21,25
014C 0B
014D 05
014E 01
014F 00
0150 03
0151 08
0152 0F
0153 15
0154 19
0155 1A                .BYTE    26,24,19,12,6,1,0,2,7
0156 18
0157 13
0158 0C
0159 06
015A 01
015B 00
015C 02
015D 07
            ;
            ;
            ; LF770:   1 CYCLE OF SINE WAVE SPREAD
            ;                  ACROSS 11 TIMING LOOP (LUP42) CYCLES
            ;
            ;          FREQ. = 1 / (11 X 117 1/3) = 774.79 HZ
            ;          ERROR = (774.79 - 770) / 770  = + 0.62 %
            ;
015E 0D                .BYTE    13,20,25,26,23,17,9,3,0,1
015F 14
0160 19
0161 1A
0162 17
0163 11
0164 09
0165 03
0166 00
0167 01
0168 06                .BYTE    6
            ;
```

TL/DD/10740–14

```
                          ;
                          ; LF852:   1 CYCLE OF SINE WAVE SPREAD
                          ;                  ACROSS 10 TIMING LOOP (LUP42) CYCLES
                          ;
                          ;          FREQ. = 1 / (10 X 117 1/3) = 852.27 HZ
                          ;          ERROR = (852.27 - 852) / 852  = + 0.03 %
                          ;
0169 0D                             .BYTE     13,21,25,26,21,13,5,1,0,5
016A 15
016B 19
016C 1A
016D 15
016E 0D
016F 05
0170 01
0171 00
0172 05
                          ;
                          ;
                          ; LF941:   1 CYCLE OF SINE WAVE SPREAD
                          ;                  ACROSS 9 TIMING LOOP (LUP42) CYCLES
                          ;
                          ;          FREQ. = 1 / (9 X 117 1/3) = 946.97 HZ
                          ;          ERROR = (946.97 - 941) / 941  = + 0.63 %
                          ;
0173 0D                             .BYTE     13,21,26,24,18,8,2,0,5
0174 15
0175 1A
0176 18
0177 12
0178 08
0179 02
017A 00
017B 05
                          ;
                          ;
                          ;
                          ; HF1209:  1 CYCLE OF SINE WAVE SPREAD
                          ;                  ACROSS 7 TIMING LOOP (LUP42) CYCLES
                          ;
                          ;          FREQ. = 1 / (7 X 117 1/3) = 1217.53 HZ
                          ;          ERROR = (1217.53 - 1209) / 1209  = + 0.71 %
                          ;
017C 10                             .BYTE     16,29,32,23,9,0,3
017D 1D
017E 20
017F 17
0180 09
0181 00
0182 03
                          ;
```

TL/DD/10740–15

```
                        ;
                        ; HF1336:  3 CYCLES OF SINE WAVE SPREAD
                        ;                  ACROSS 19 TIMING LOOP (LUP42) CYCLES
                        ;
                        ;          FREQ. = 3 / (19 X 117 1/3) = 1345.69 HZ
                        ;          ERROR = (1345.69 - 1336) / 1336  = + 0.73 %
                        ;
0183 10                           .BYTE     16,29,31,19,4,0,11,26,32,24
0184 1D
0185 1F
0186 13
0187 04
0188 00
0189 0B
018A 1A
018B 20
018C 18
018D 08                           .BYTE     8,0,6,21,32,28,13,1,3
018E 00
018F 06
0190 15
0191 20
0192 1C
0193 0D
0194 01
0195 03
                        ;
                        ;
                        ; HF1477:  4 CYCLES OF SINE WAVE SPREAD
                        ;                  ACROSS 23 TIMING LOOP (LUP42) CYCLES
                        ;
                        ;          FREQ. = 4 / (23 X 117 1/3) = 1482.21 HZ
                        ;          ERROR = (1482.21 - 1477) / 1477  = + 0.35 %
                        ;
0196 10                           .BYTE     16,30,29,14,1,4,20,32,26,10
0197 1E
0198 1D
0199 0E
019A 01
019B 04
019C 14
019D 20
019E 1A
019F 0A
01A0 00                           .BYTE     0,8,24,32,22,6,0,12,28,31
01A1 08
01A2 18
01A3 20
01A4 16
01A5 06
01A6 00
```

TL/DD/10740–16

```
 01A7 0C
 01A8 1C
 01A9 1F
 01AA 12                        .BYTE     18,3,2
 01AB 03
 01AC 02
                     ;
                     ;
                     ; HF1633:  4 CYCLES OF SINE WAVE SPREAD
                     ;                  ACROSS 21 TIMING LOOP (LUP42) CYCLES
                     ;
                     ;          FREQ. = 4 / (21 X 117 1/3) = 1623.38 HZ
                     ;          ERROR = (1633 - 1623.38) / 1633  = - 0.59 %
                     ;
 01AD 10                        .BYTE     16,31,27,9,0,11,29,30,14,0
 01AE 1F
 01AF 1B
 01B0 09
 01B1 00
 01B2 0B
 01B3 1D
 01B4 1E
 01B5 0E
 01B6 00
 01B7 07                        .BYTE     7,25,32,18,2,3,21,32,23,5
 01B8 19
 01B9 20
 01BA 12
 01BB 02
 01BC 03
 01BD 15
 01BE 20
 01BF 17
 01C0 05
 01C1 01                        .BYTE     1
                     ;
                     ;
                     ;
```

TL/DD/10740–17

```
                            .FORM
                  ;
                  ; DTMF KEYBOARD DECODE SUBROUTINE (KBRDEC)
                  ;
                  ; KEYBOARD INPUT DATA IS IN ACCUMULATOR WITH A
                  ;    LOW TRUE FORMAT AS FOLLOWS:
                  ;        BITS 7 TO 4 : LOW TRUE COLUMN VALUE (E,D,B,7)
                  ;        BITS 3 TO 0 : LOW TRUE ROW VALUE (E,D,B,7)
                  ;
                  ; ASSUMPTION MADE THAT COLUMN STROBES (LOW TRUE) ARE
                  ;    OUTPUT, WHILE ROW VALUES (LOW TRUE) ARE INPUT.
                  ;
                  ; LOW TRUE COLUMN/ROW INPUT DIGIT IN ACCUMULATOR IS
                  ;     TRANSFORMED INTO A DTMF HEX DIGIT KEY VALUE
                  ;
                  ; TABLE LOOKUP TRANSFORMATION CHECKS FOR MULTIPLE KEYS,
                  ;     NO KEY, OR NO COLUMN SELECT, AND THEN PRODUCES
                  ;     A DTMF HEX DIGIT KEY VALUE WITH A BINARY FORMAT
                  ;     OF 0000RRCC FOR A SINGLE KEY INPUT,
                  ;     WHERE   -  RR IS LOW BAND (LB) FREQUENCY SELECT
                  ;             -  CC IS HIGH BAND (HB) FREQUENCY SELECT
                  ;
                  ; KBRDEC SUBROUTINE IS EXITED WITH A RETURN (RET)
                  ;     COMMAND TO INDICATE MULTIPLE KEYS, NO KEY,
                  ;     OR NO COLUMN SELECT
                  ;
                  ; KBRDEC SUBROUTINE IS EXITED WITH A RETURN AND SKIP
                  ;     (RETSK) COMMAND TO INDICATE A SINGLE KEY ENTRY
                  ;
                  ;
      0200                  .=0200
                  ;
                  ; LOW TRUE TRANSLATION TABLE  - ONLY E,D,B,7 ACCEPTABLE
                  ;
0200 C0                     .BYTE  0C0,0C0,0C0,0C0,0C0,0C0,0C0,0C
0201 C0
0202 C0
0203 C0
0204 C0
0205 C0
0206 C0
0207 0C
0208 C0                     .BYTE  0C0,0C0,0C0,8,0C0,4,0,0C0
0209 C0
020A C0
020B 08
020C C0
020D 04
020E 00
020F C0
                  ;
```

TL/DD/10740–18

```
                    ;
0210 5F             KBRDEC:   LD        B,#KDATA
0211 A6                       X         A,[B]     ; STORE LOW TRUE
0212 AE                       LD        A,[B]     ;   COLUMN/ROW VALUE
0213 95F0                     AND       A,#0F0    ; EXTRACT LOW TRUE COLUMN
0215 65                       SWAP      A         ;   & PUT IN LOWER NIBBLE
0216 A4                       LAID                ; 0000CC00 FROM TABLE
0217 A0                       RC                  ; SHIFT TABLE VALUE DOWN
0218 B0                       RRC       A         ;   TWO BITS TO PRODUCE
0219 B0                       RRC       A         ;   000000CC
021A A6                       X         A,[B]     ; STORE RESULT
021B 950F                     AND       A,#0F     ; EXTRACT LOW TRUE ROW
021D A4                       LAID                ; 0000RR00 FROM TABLE
021E 84                       ADD       A,[B]     ; ADD TO PRODUCE 0000RRCC
021F 930F                     IFGT      A,#0F     ; RETURN IF MULTIPLE KEYS,
0221 8E                       RET                 ;    NO KEYS, OR NO COLUMN
0222 8D                       RETSK               ; RETURN AND SKIP
                                                  ;    IF SINGLE KEY
                    ;
                    ;
                    ;
                              .END
```

TL/DD/10740–19

```
B       00FE         BYP1    0072         BYP2    0075         BYPA    0019
BYPB    001B         CNTRL   00EE         DTMFGP  0040         DTMFLP  008E
FINI    0086         FRLUP   00A0         HFPTR   0007         HFTADR  000B
HFTBSZ  000A         INTRPT  00FF   *     KBRDEC  0210         KDATA   0000
LFPTR   0005         LFTADR  0009         LFTBSZ  0008         LOOP    0006
LUP     004D         LUP1    006C         LUP2    0078         LUP42   010F
PORTD   00DC         PORTGC  00D5         PORTGD  00D4         PORTI   00D7
PORTLC  00D1         PORTLD  00D0         PSW     00EF         R0      00F0
R1      00F1         R2      00F2         R3      00F3         SP      00FD
START   0000   *     TAUHI   00ED   *     TAULO   00EC   *     TEMP    0006
TMRHI   00EB   *     TMRLO   00EA         TRUN    0004         X       00FC
```

TL/DD/10740–20

LIFE SUPPORT POLICY

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