AN-1204 DTMF Tone Generator

Dual-tone multi-frequency signaling (DTMF) was first developed by Bell Labs in the 1950's as a method to support the then revolutionary push button phone. This signaling system uses a pair of tones to represent digital data that can be transmitted over the voice band of a telecommunication link. There are two groups of four frequencies defined in the system, and information is coded with one of the frequencies from each group transmitted simultaneously. This results in a total of sixteen frequency-pair combinations (one from each group) to represent sixteen different numbers, symbols and letters. DTMF use expanded into a broad range of communications and control applications, further supported by standardization by the International Telecommunications Union in its Recommendation Q.23.

In this application note, we implemented the digital and analog stages of a DTMF tone generator, so the eight tones of the standard and the resulting signal is generated. For this purpose, we used the GreenPAK™ SLG46620V and OpAmp SLG88104V. The resulting signal is the sum of two tones, selected by the corresponding row and column of the telephone keypad.

The system has four input bits to define which signal combination will be generated. It also has an enable input, which starts the generation and specifies the time duration of the produced signal. The output of this system is the resulting signal (the sum of the two signals with the selected frequencies) with frequency precision compatible with the ITU-T DTMF Standard.

DTMF signals

DTMF standard defines that the digits 0-9, the characters A, B, C and D and the symbols * and # are represented as a combination of two frequencies. These frequencies are classified into two groups, called High Group Frequencies and Low Group Frequencies. Table 1 shows the frequencies, groups and the corresponding digit representations.

	High Group				
Low Group		1209 Hz	1336 Hz	1477 Hz	1633 Hz
	697 Hz	1	2	3	A
	770 Hz	4	5	6	B
	852 Hz	7	8	9	C
	941 Hz	*	0	#	D

Table 1. DTMF Frequencies

The tone frequencies were chosen such that harmonics are avoided. With the frequencies selected, there is no frequency which is a multiple of another. Also, the sum or difference of two frequencies does not result in another DTMF frequency. In conclusion, harmonic distortion or intermodulation distortion is avoided.

The standard Q.23 specifies that each transmitted frequency must be within ± 1.8% of the nominal frequency and the total distortion products (resulting from harmonics or intermodulation) must be at least 20 dB below the fundamental frequencies.

The resulting signal, based on the previous description, can be modeled as:

$$s(t) = A\cos\left(2\pi f_{high} t\right) + A\cos\left(2\pi f_{low} t\right)$$

Where f_{high} and f_{low} are the corresponding frequencies of the High Group and the Low Group. In Figure 1, the resulting signal for the digit "1" is shown. In Figure 2, the corresponding frequency spectrum can be seen.

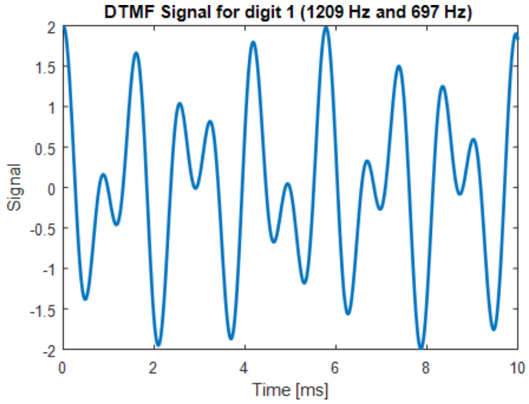


Figure 1. DTMF Signal

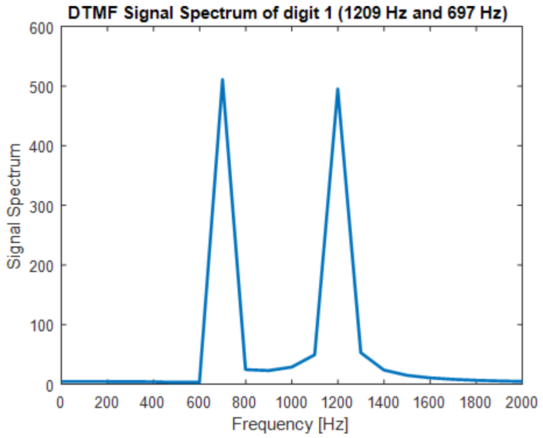


Figure 2. DTMF Signal Spectrum

The time duration of the signal is variable, depending on the application where the DTMF signaling is used. The most important categories of applications are defined by the nature of the dialing, so it varies between manual dialing and automatic dialing. In Table 2, a summary of typical time duration classified by the dialing type is shown.

	High Group		High Group	
	Min.	**Max.**	**Min.**	**Max.**
Manual Dialing	65 ms	-	80 ms	-
Automatic Dialing	65 ms	100 ms	80 ms	6500 ms

Table 2. DTMF Frequencies

To obtain more flexibility, the system implemented in this app note has one enable input, which starts the generation and defines the time duration of the signal, which will be equal to the enable pulse duration.

Analog stage

The ITU-T Recommendation Q.23 specifies the DTMF signals as analog signals, modeled as two sine waves. In this app note, the GreenPAK SLG46620V generates square waveforms with its frequencies corresponding to the desired frequencies. In order to obtain the sine waves with the correct frequencies and to obtain the resulting signal (the sum of two sine waves) analog filters and adders must be used. This is the main reason we used the SLG88104V Operational Amplifier for this project.

In Figure 3, a block diagram of the analog stage is shown.

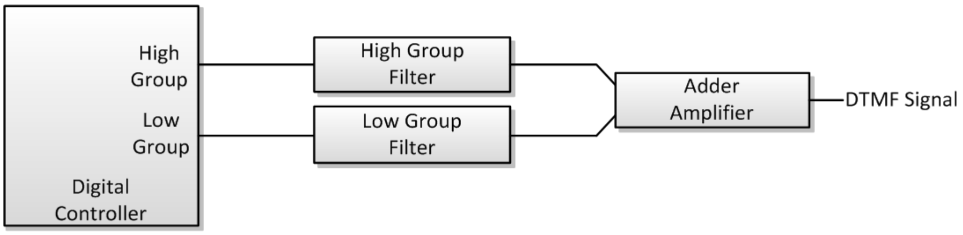


Figure 3. Analog Stage Block Diagram

We used two filters to obtain the sine waveforms from our square waves. Finally, the two waveforms are summed together to obtain the desired output.

In Figure 4, the obtained spectrum of a square waveform is shown after a Fourier analysis.

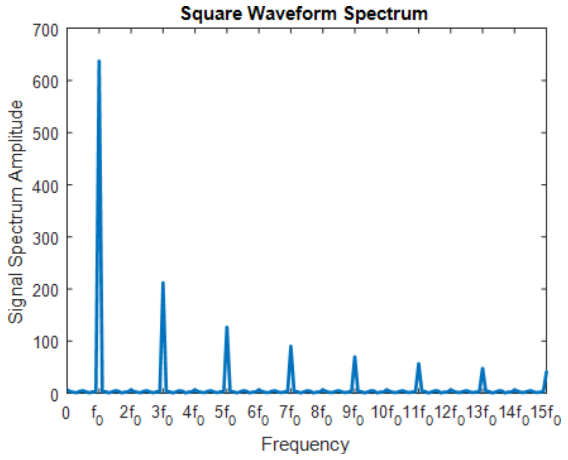


Figure 4. Square Waveform Spectrum

As you can see, the square waveform has only odd harmonics. If the square waveform with amplitude A is represented by its Fourier series, we find that:

$$s(t) = \frac{A}{2} + \frac{2A}{\pi}[\ \sin(2\pi f_0 t) + \frac{1}{3}\ \sin(2\pi 3 f_0 t) + \frac{1}{5}\sin(2\pi 5 f_0 t) + \cdots]$$

From this analysis, we can conclude that if the analog filters have enough attenuation for the harmonics, we can obtain sine signals with the same frequency of the square waveforms.

Considering the Q.23 standard specification about interference, all the harmonics must be attenuated 20 dB or more. Also, considering that any frequency of the Low Group can be combined with any frequency of the High Group, we designed two filters, one for each group.

We used a low-pass Butterworth topology for both filters. The attenuation of a Butterworth filter of order *n* can be calculated as:

$$A(f)[dB] = 10\log(A(f)^2) = 10\log\left(1 + \left(\frac{f}{f_c}\right)^{2n}\right)$$

Where f_C is the cut-off frequency of the filter and *n* is the order.

To determine the parameters of the filters, the attenuation between the lowest frequency and the highest frequency of each group can be 3dB maximum, so:

$$A(f_{Higher})[dB] - A(f_{Lower})[dB] > 3dB$$

Considering the absolute values,

$$\frac{A(f_{Higher})^2}{A(f_{Lower})^2} > 2$$

Also, as we previously stated, the attenuation of the harmonics must be 20 dB or more. The worst case is with the lowest frequency of the group, because its 3rd harmonic is the lowest harmonic and closer to the cut-off frequency. Considering that the 3rd harmonic is 3 times lower than the fundamental, the filter must encompass (in absolute values):

$$\frac{A(3f_{Lower})^2}{A(f_{Lower})^2} > \frac{10}{3}$$

If these equations are applied to both groups, the resulting filters must be 2nd order filters. This means that the filters will have two resistors and two capacitors if they are implemented with OpAmps. If we used 3rd order filters, the sensitivity to component tolerances would be reduced.

The selected cut-off frequencies of the filters are 977 Hz for the Low Group and 1695 Hz for the High Group. With these values, the level differences within a frequency group agree with the required values, and the sensitivity to changes in the cut-off frequency due to component tolerances is lower.

The schematic circuit of the filters implemented with SLG88104V is shown in Figure 5. The first R-C pair values are selected for not driving much current from the SLG46620V. The second stage of the filters determines the amplification, which is set to 0.2. The DC levels of the square waves set the operating point of the OpAmps to 2.5V. This undesired level is blocked by the output capacitors of the filters.

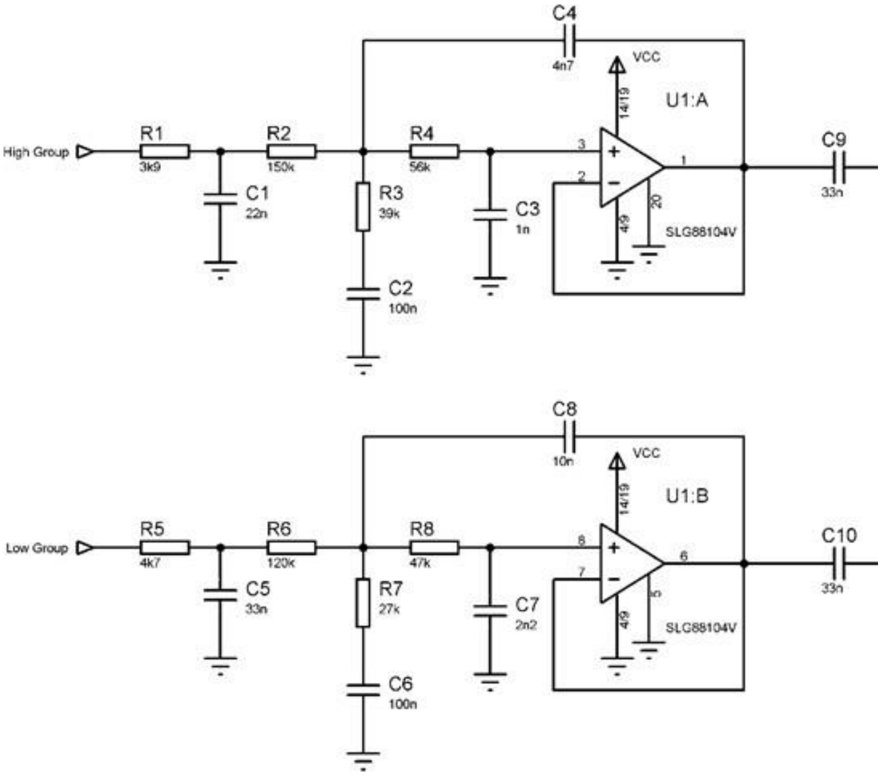


Figure 5. Filters Schematic Circuit

Finally, in the adder stage, the outputs of the filters are summed, so the resulting signal is the sum of the signal selected from the High Group and the signal selected from the Low Group. The level of the output signal can be adjusted with the two resistors R9 and R10 to compensate the attenuation of the filter stage. In Figure 6, the adder stage circuit is shown, and in Figure 7, the entire analog stage can be seen.

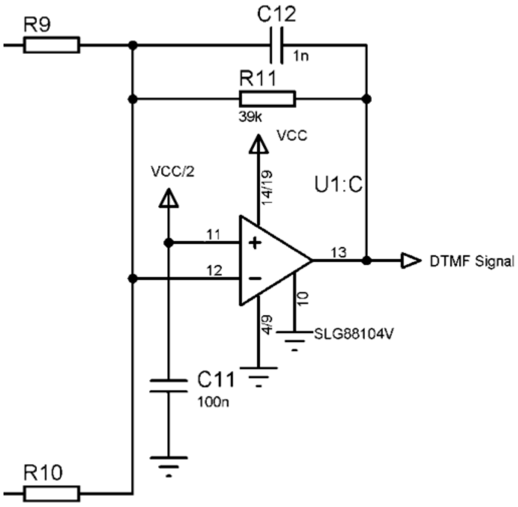


Figure 6. Adder Schematic Circuit

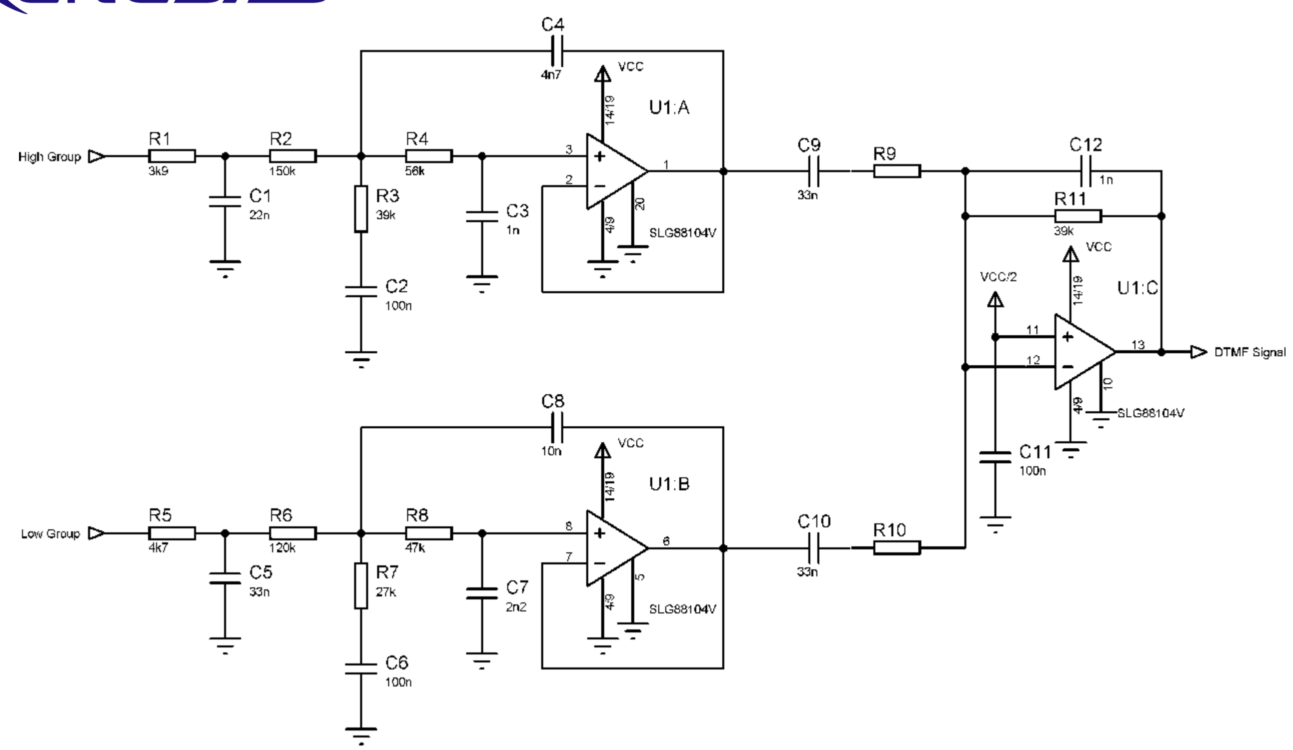


Figure 7. Analog Stage Schematic Circuit

Digital Stage

The digital stage of the DTMF Tone Generator is based on square waveforms generators for each frequency of the DTMF standard. We selected the GreenPAK SLG46620V for this implementation because it requires eight counters for the waveform generators. The outputs of this Stage are two square waveforms, one for each group.

The square waveforms are implemented with a counter and a D-Flip Flop configured to generate a 50% duty cycle square waveform. To do this, the counter is configured to have an output frequency equal to the double of the desired frequency, and the DFF divides the counter's output by two.

The clock source for the counters is the 2 MHz RC internal oscillator, divided by 4 or 12. This clock variation is based on the number of bits of each counter and the counter data required for each frequency.

Generating higher frequencies requires fewer counts, so most of the higher frequencies are implemented with 8 bit counters and RC internal oscillator divided by 4. For the same reason, lower frequencies are implemented with 14 bit counters.

The SLG46620V has three standard 14-bit counters available, so one of the lower frequencies was implemented with an 8-bit counter. To implement that frequency, we used the RC oscillator divided by 12 as the clock source to CNT8 to reduce the counter data to a number in the range of 0 to 255. To choose which frequency to implement with these conditions, the one with the higher counter data is selected to have less error. This is why the lowest frequency is implemented with this counter type.

In Table 3, the details of each square waveform are shown.

		Clock Condition	**Counter Data**	**Frequency Error [%]**
Low Group	697 Hz	RC/12	120	0.37
	770 Hz	RC/4	325	0.1
	852 Hz	RC/4	293	0.15
	941 Hz	RC/4	266	0.12
High Group	1209 Hz	RC/4	207	0.11
	1336 Hz	RC/4	187	0.07
	1477 Hz	RC/4	169	0.16
	1633 Hz	RC/4	153	0.06

Table 3. Square Waveform Generators Details

As can be seen from the table, all of the frequencies have an error of less than 1.8%, so they are DTMF Standard compliant. These theoretical values (based on an ideal frequency value of the RC oscillator) can be adjusted by measuring the output frequency to obtain the desired frequencies with the real value of the RC oscillator.

In this implementation, all the square waveform generators are working in parallel, but only one from each frequency group will be output at any time, so the user must select which frequency is output. To do this, we used 4 GPIOs (two bits for each group) with the truth table shown in Table 4 for the Low Group and Table 5 for the High Group.

	R1	**R0**	**Output Frequency**
Low Group	0	0	697 Hz
	0	1	770 Hz
	1	0	852 Hz
	1	1	941 Hz

Table 4. Low Group Selection Truth Table

	C1	**C0**	**Output Frequency**
High Group	0	0	1209 Hz Hz
	0	1	1336 Hz
	1	0	1477 Hz
	1	1	1633 Hz

Table 5. High Group Selection Truth Table

Figure 8 shows the logic diagram of the 852 Hz square waveform generator. This logic is repeated for each frequency with the corresponding counter data and LUT configuration.

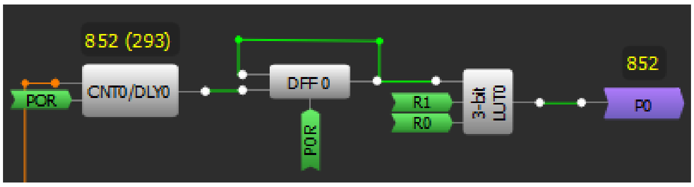
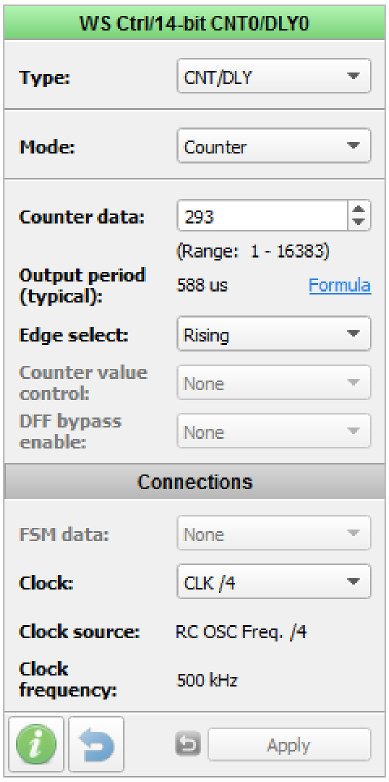


Figure 8. Square Waveform Generator

The counter runs freely with an output frequency (defined by the Counter Data) equal to twice the corresponding DTMF tone frequency. This configuration setting is shown in Figure 9.

The output of the counter is connected to the clock input of the D-Flip Flop. The output of the DFF is configured to be the inverted Q, so by connecting the output of the DFF to its input, the D-type is converted to a T-type. The configuration of the DFF can be seen in Figure 10.

The output of the DFF is connected to the LUT to select this frequency only when R1-R0 has the corresponding value. That is, if R1 is high and R0 is low, the output port copies the waveform. Else, the output port is always low. This configuration can be seen in Figure 11.



WS Ctrl/14-bit CNT0/DLY0

Type: CNT/DLY

Mode: Counter

Counter data: 293

(Range: 1 - 16383)

Output period (typical): 588 us Formula

Edge select: Rising

Counter value control: None

DFF bypass enable: None

Connections

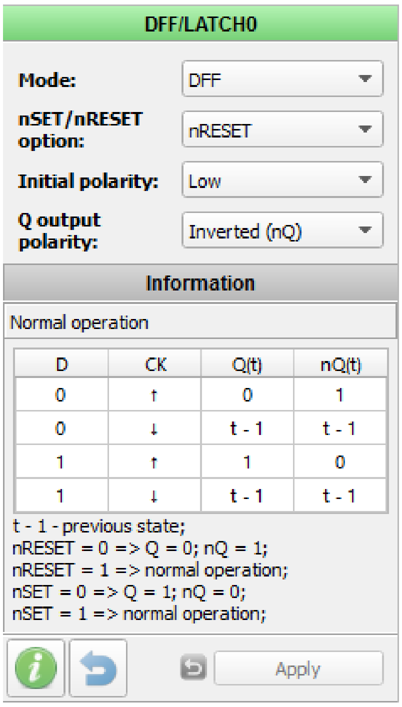
FSM data: None

Clock: CLK /4

Clock source: RC OSC Freq. /4

Clock frequency: 500 kHz

Figure 9. Counter of Square Waveform Generator



DFF/LATCH0

Mode: DFF

nSET/nRESET option: nRESET

Initial polarity: Low

Q output polarity: Inverted (nQ)

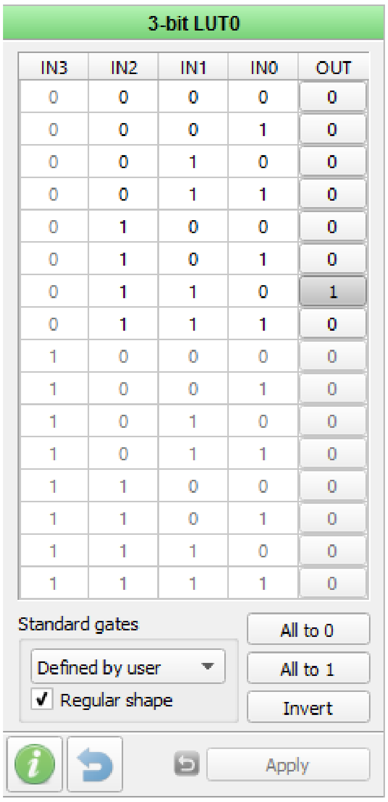
Information

Normal operation

D	CK	Q(t)	nQ(t)
0	↑	0	1
0	↓	t - 1	t - 1
1	↑	1	0
1	↓	t - 1	t - 1

t - 1 - previous state;
nRESET = 0 => Q = 0; nQ = 1;
nRESET = 1 => normal operation;
nSET = 0 => Q = 1; nQ = 0;
nSET = 1 => normal operation;

Figure 10. Flip Flop of Square Waveform Generator



3-bit LUT0

IN3	IN2	IN1	IN0	OUT
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	0
0	1	0	1	0
0	1	1	0	1
0	1	1	1	0
1	0	0	0	0
1	0	0	1	0
1	0	1	0	0
1	0	1	1	0
1	1	0	0	0
1	1	0	1	0
1	1	1	0	0
1	1	1	1	0

Standard gates: Defined by user

☑ Regular shape

Figure 11. Look Up Table of Square Waveform Generator

As mentioned earlier, this implementation also has an enable pin. With Enable is HIGH, it enables the two square waveform outputs and determines the duration of these tones, which is equal to the pulse duration on the pin. To implement this, we used a few more LUTs.

For the High Group Frequencies, one 4-bit LUT and one 2-bit LUT are used as shown in Figure 12.

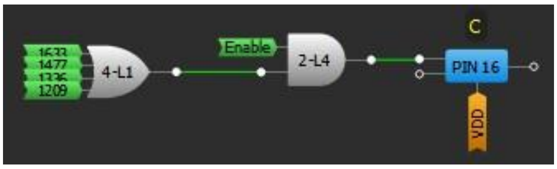
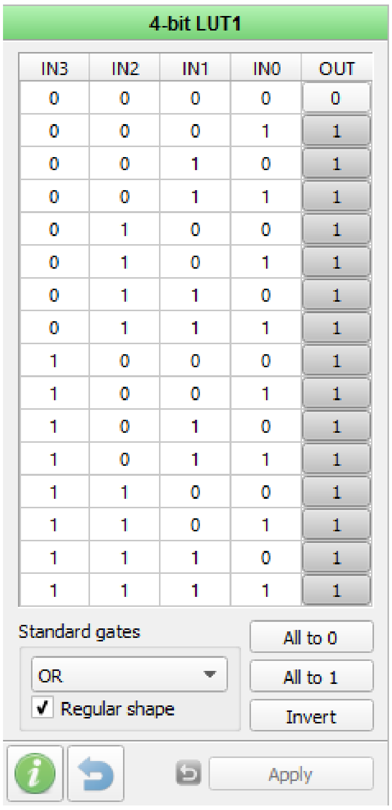


Figure 12. High Group Output Diagram

4-bit LUT1 is configured as an OR gate, so it outputs HIGH if anyone of its inputs is HIGH. The C1/C0 truth tables enable only one of the frequency generators at any time, so 4-bit LUT1 passes the selected waveform. The output of that LUT is connected to 2-bit LUT4, which passes the waveform only if the enable signal is HIGH. In Figure 13 and Figure 14, 4-bit LUT1 and 2-bit LUT 4 configurations are shown.



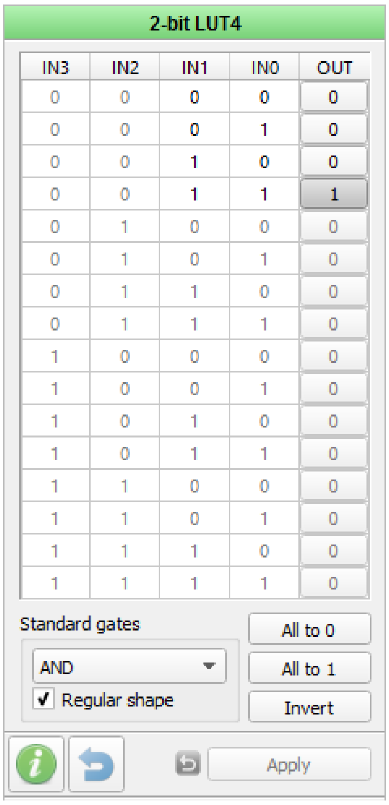
4-bit LUT1

IN3	IN2	IN1	IN0	OUT
0	0	0	0	0
0	0	0	1	1
0	0	1	0	1
0	0	1	1	1
0	1	0	0	1
0	1	0	1	1
0	1	1	0	1
0	1	1	1	1
1	0	0	0	1
1	0	0	1	1
1	0	1	0	1
1	0	1	1	1
1	1	0	0	1
1	1	0	1	1
1	1	1	0	1
1	1	1	1	1

Standard gates: OR

☑ Regular shape

Figure 13. 4-bit LUT1 Configuration



2-bit LUT4

IN3	IN2	IN1	IN0	OUT
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	1
0	1	0	0	0
0	1	0	1	0
0	1	1	0	0
0	1	1	1	0
1	0	0	0	0
1	0	0	1	0
1	0	1	0	0
1	0	1	1	0
1	1	0	0	0
1	1	0	1	0
1	1	1	0	0
1	1	1	1	0

Standard gates: AND

☑ Regular shape

Figure 14. 2-bit LUT4 Configuration

For the Low Group Frequencies, we used two 3-bit LUTs to create the same logic since there were no more 4-bit LUTs available.

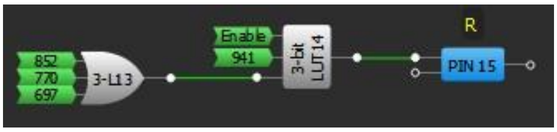


Figure 15. Low Group Output Diagram

The entire implementation is shown in Figure 16, while Figure 17 shows the entire circuit schematic.

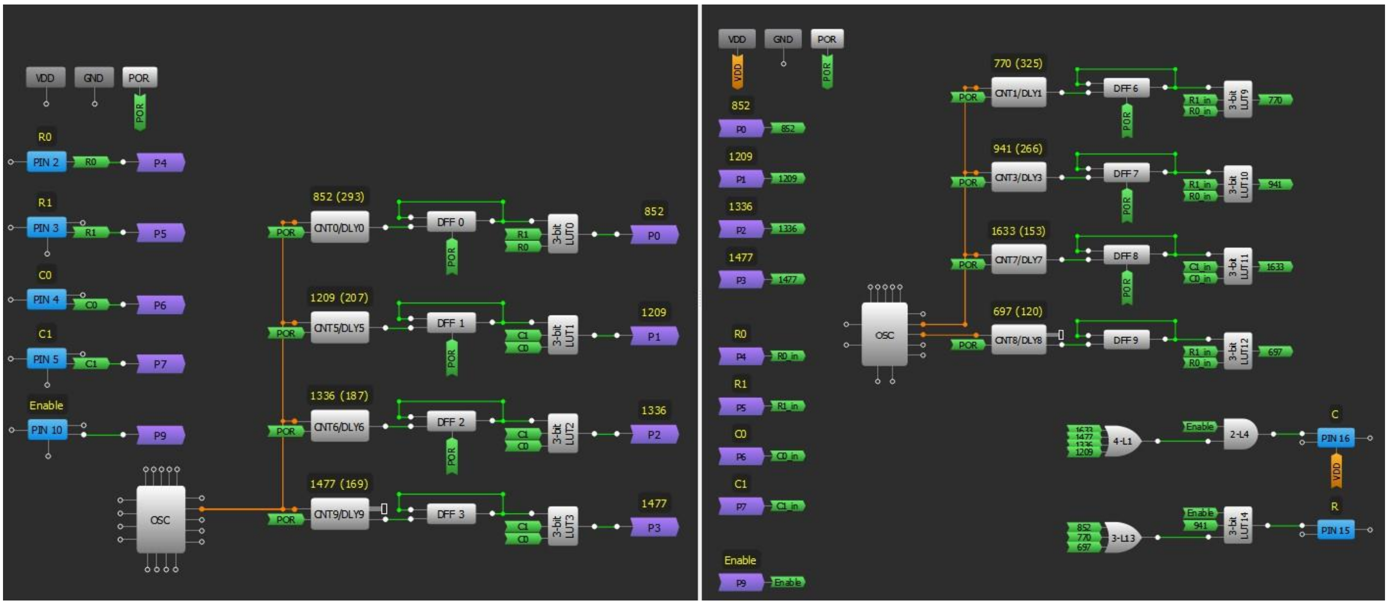


Figure 16. DTMF Tone Generator block diagram

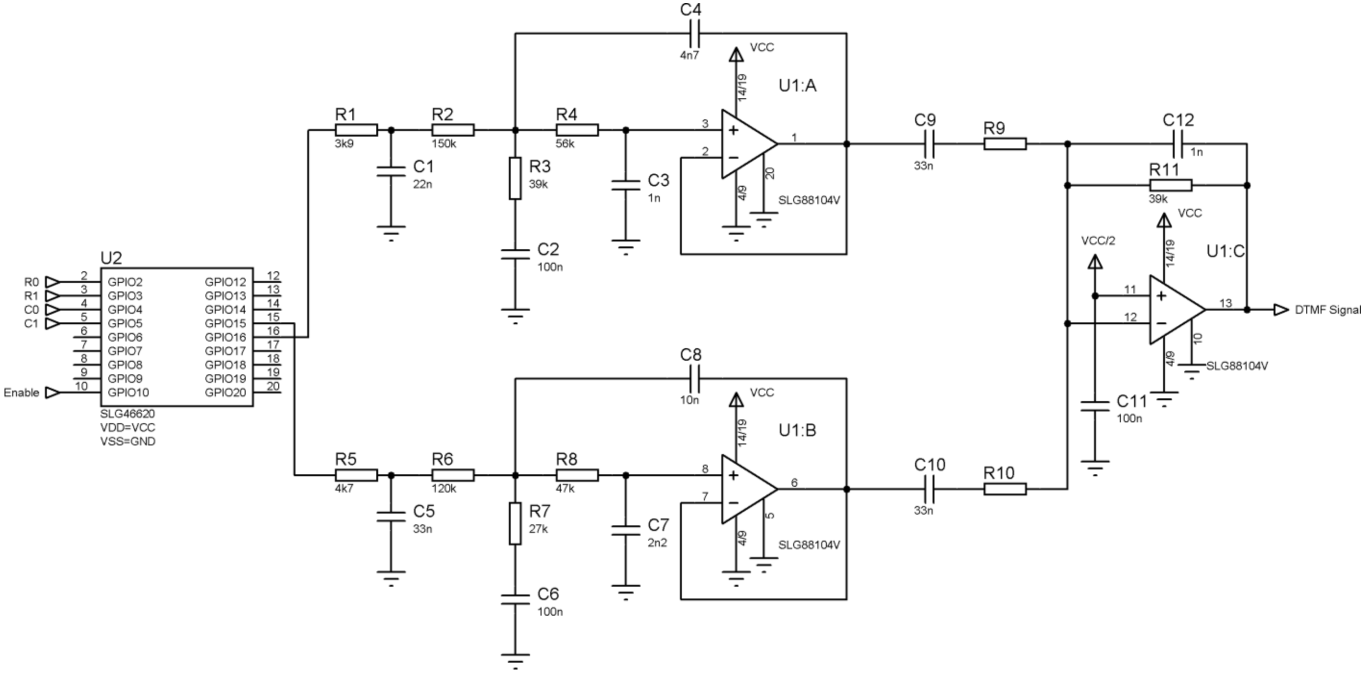
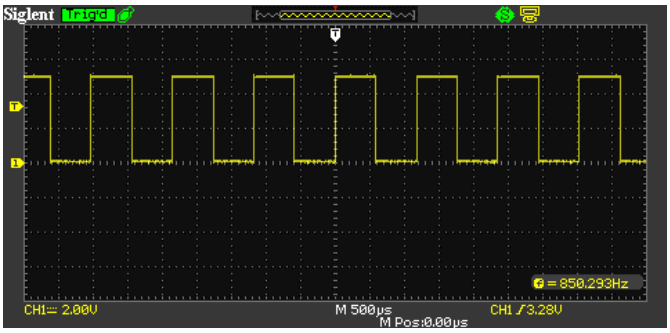


Figure 17. DTMF Tone Generator Schematic Circuit

Tests and Conclusion

To test the implementation, we acquired square wave outputs corresponding to the different DTMF signals with an oscilloscope to verify the frequencies. Figure 18 and Figure 19 display the square wave outputs for 852 Hz and 1477 Hz.



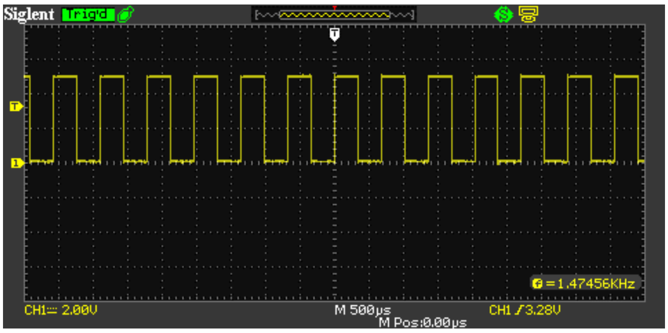
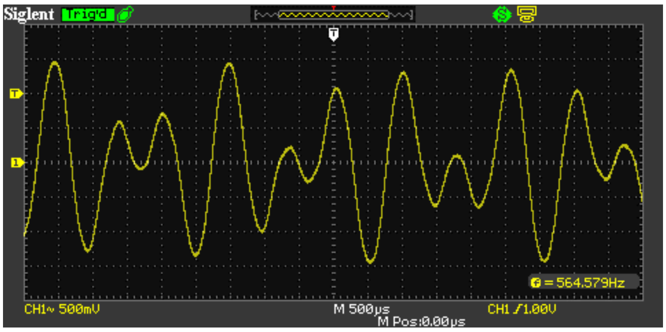


Figure 18. 852 Hz Square Wave

Figure 19. 1477 Hz Square Wave

Once all the signal frequencies were verified, we measured the output of the filter corresponding to the sum of a Low Group Signal and a High Group Signal. Figure 20 shows the sum of 770 Hz and 1209 Hz, while Figure 21 shows the sum of 941 Hz and 1633 Hz.



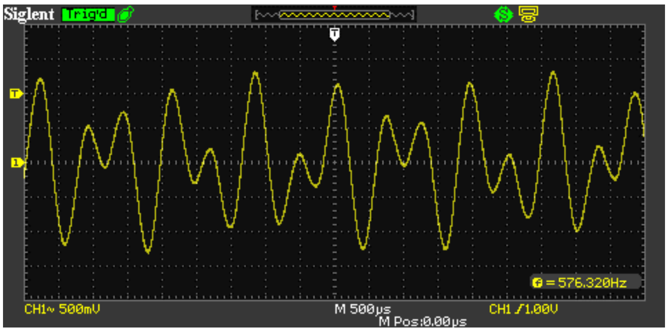


Figure 20. 770 Hz and 1209 Hz DTMF Signal

Figure 21. 941 Hz and 1633 Hz DTMF Signal

Conclusion

In this application note, we implemented a DTMF Tone Generator using a GreenPAK SLG46620V and a SLG88104V OpAmp. The user must set the frequencies for the two DTMF groups and provide a pulse which enables the generation and defines the signal duration. This design needs only 5 signals from the user to generate the corresponding DTMF signal.

IMPORTANT NOTICE AND DISCLAIMER

RENESAS ELECTRONICS CORPORATION AND ITS SUBSIDIARIES (“RENESAS”) PROVIDES TECHNICAL SPECIFICATIONS AND RELIABILITY DATA (INCLUDING DATASHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES “AS IS” AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING, WITHOUT LIMITATION, ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT OF THIRD-PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for developers who are designing with Renesas products. You are solely responsible for (1) selecting the appropriate products for your application, (2) designing, validating, and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, or other requirements. These resources are subject to change without notice. Renesas grants you permission to use these resources only to develop an application that uses Renesas products. Other reproduction or use of these resources is strictly prohibited. No license is granted to any other Renesas intellectual property or to any third-party intellectual property. Renesas disclaims responsibility for, and you will fully indemnify Renesas and its representatives against, any claims, damages, costs, losses, or liabilities arising from your use of these resources. Renesas' products are provided only subject to Renesas' Terms and Conditions of Sale or other applicable terms agreed to in writing. No use of any Renesas resources expands or otherwise alters any applicable warranties or warranty disclaimers for these products.

(Disclaimer Rev.1.01)

Corporate Headquarters

TOYOSU FORESIA, 3-2-24 Toyosu,
Koto-ku, Tokyo 135-0061, Japan
www.renesas.com

Trademarks

Renesas and the Renesas logo are trademarks of Renesas Electronics Corporation. All trademarks and registered trademarks are the property of their respective owners.

Contact Information

For further information on a product, technology, the most up-to-date version of a document, or your nearest sales office, please visit www.renesas.com/contact-us/.