Simple and Generic Tone Generator for IP Phone

M. Senthil Kumar
Project Leader
senthil@hyd.hellosoft.com

P. Chandra Sekhar
Director
chandra@hyd.hellosoft.com

M. Nathan
Member Technical
mnathan@hyd.hellosoft.com

Aditya Chebiyyam
Member Technical
caditya@hyd.hellosoft.com

ABSTRACT

IP Phones need to handle Call Progress Tones (CPT) generation to provide an audible indication of the state of the call similar to the PSTN phone. Additionally, in case of systems having a keypad interface, DTMF tones need to be generated based on the key activated, either during call setup or during mid-call session. So, very low complexity generic Tone generator (TG) is required for any application, which needs to generate tones. The digital oscillator algorithm is used for generation of a single sinusoidal wave frequency. Both amplitude and frequency modulated tones are generated by different sine or cosine frequency separately mixed as per requirement of modulation scheme used. This algorithm is tested with a large number of other signaling tones (such as MF-R1, MF-R2, Caller ID Alerting Signal (CAS) etc), and the results establish that the method employed is a good compromise in terms of performance, memory requirements, and precision. Design guidelines and implementation details of generic TG are presented to further emphasize the mathematical effectiveness of the method employed.

Keywords

Call Progress Tones, Tone Parameters, Oscillator co-efficients, DTMF tones.

1 INTRODUCTION

Multiple developments have taken place in packet based communication systems such as IP phones. To provide an audible indication of the state of the call similar to PSTN call in the IP phones, the Call Progress Tones (CPT) such as, dial tone, busy tone, ring tone, ring-back tone etc. are to be generated during call setup process. Besides, in case of systems having a keypad interface, Dual tone Multi Frequency (DTMF) tones are to be generated based on the key activated, either during call setup or during mid-call session. In either case, the DTMF [1] tone is generated locally, based on the key activated. During mid-call, in addition to local tone generation, the activated key information is passed to the other party as per RFC2833 [2]. But, in case of systems having analog phone interfaced through passive SLIC/DAA, indication of the information about the calling or called party through the telephone line can be given during the ringing or call waiting phase of a telephone call. So Caller ID (Identification) module, which generates calling or called user information such as date and time, caller name, telephone number etc. is embedded within CPT module. The tone parameters such as number of frequencies, value of each frequency, cadence and power level of each frequency in a tone will vary for different countries [3]. Similarly, various Caller ID standards are available in the world such as British Telecom, NTT, ETSI, and AUSTEL etc. The tone parameters of the alerting tone or signal to be transmitted to the Analog Telephone Adapter (ATA) vary for different standards. So, a generic tone generator is required and that should generate a tone with configurable tone parameters.

In this paper, design of a simple and generic tone generator is presented with test results. This generic tone generator can generate tones with configurable tone parameters. Also, the design provides low MIPS consumption (approx. 0.5 MIPS) on ARM9E core. As mentioned, additional tones or country specific tones can be easily added or modified by just adding or modifying the required parameters in the tone table which holds tone parameters for different tones with their IDs (denoted by tone_id).

2 THEORY OF DIGITAL OSCILLATOR

The digital oscillator algorithm is used for generation of a single sinusoidal wave. A Tone, which has more than one frequency at a time is resolved into different sine or cosine components and is generated separately using different oscillators. All tones generated from the different oscillators are mixed for the resultant wave. Generation of a single tone basically implies generating samples of a sine/cosine wave. The Z-transform of a sine wave is given in equation (1).

$$Z(\sin\theta) = \frac{Y(z)}{X(z)} = \frac{z\sin\theta}{z^2 - 2z\cos\theta + 1} \quad (1)$$

The impulse response of the above transform in equation (1) will generate a sine wave of frequency θ. Thus the above equation is translated to equation (2) as below.

[1] VoIP – DSP Group, HelloSoft, 8-2-703, Road No. 12, Banjara Hills, Hyderabad, India. Tel: +91-40-23327062.

$$Y(z) = \frac{z^{-1}\sin\theta}{1 - 2z^{-1}\cos\theta + z^{-2}} \quad (2)$$

Equation (2) can be rewritten in a difference equation form as follows:

$$y(n) - 2y(n-1)\cos\theta + y(n-2) = x(n-1)\sin\theta \quad (3)$$

Thus, the equation for the oscillator is given as follows:

$$x(n) = 2\cos(2\pi f / f_s)x(n-1) - x(n-2) \quad (4)$$

where, $x(n)$ is a sine wave signal, f is the required frequency, f_s is the sampling frequency and $2\cos(2\pi f / f_s)$ is the resonator constant or oscillator co-efficient. The amplitude and phase of the sine wave are determined by the initial states $x(-2)$ and $x(-1)$. The amplitude A of the tone is estimated from the power level required as per equation (5). The initial conditions for generating a sine wave are given in the equation (6) and for the cosine wave generation are given in the equation (7).

$$A = 10^{Level(dBm0)/20} \quad (5)$$

$$x(-2) = \pm A\sin(2\pi f / f_s),\ \ x(-1) = 0 \quad (6)$$

$$x(-2) = \pm A,\ \ x(-1) = \pm A\cos(2\pi f / f_s) \quad (7)$$

For a pre-defined number of target frequencies, the initialization values can be pre-computed. However, a generic tone generation requires the computation of the initial conditions with high precision for any frequency. To compute oscillator co-efficient polynomial approximation is used. This method uses the Taylor's expansion of a cosine given as:

$$\cos x = \sum_{k=0}^{\infty} a_k x^{2k},\ \ a_k = (-1)^k / ((2k)!) \quad (8)$$

Replacing x with the $2\pi f / f_s$ and defining the normalized frequency as follows yields $f_n = f / f_s$

$$\cos(2\pi f_n) = \sum_{k=0}^{\infty} b_k f_n^{2k},\ b_k = (2\pi)^{2k} a_k \quad (9)$$

The actual implementation uses a factorized version of this formula with a finite number of elements and normalized coefficients. A sixth-order polynomial provides sufficient precision for 16-bit coefficients; furthermore, the same polynomial can be used to compute either sine or cosine. Thus, the final polynomial approximation is given in the equation (10) and (11).

$$\cos(2\pi f_n) = 2^m (((c_3 f_n^2 + c_2) f_n^2 + c_1) f_n^2 + c_0 \quad (10)$$

$$c_k = 2^{-m} b_k \quad (11)$$

The corresponding normalized coefficients to be stored in memory are scaled by 2^{-7} (that is, m = 7) and listed in following Table 1.

Table 1. Values of Polynomial Co-efficients

Co-efficient	Values
c_0	0.0078125
c_1	–0.1542053
c_2	0.5073242
c_3	-0.6676025

3 SYSTEM ARCHITECTURE

In the design of tone generation process, tone to be generated is decomposed into elementary components. So, each tone cycle consists of concatenated multiple tone components with or without silence and each component has its own tone parameters. The maximum number of components allowed could be limited based on the different country specifications. A maximum of four components are required in a tone cycle and two frequencies are required in a component to support most of the specifications of different countries in the world [3]. The tone cycle can be repeated as often as needed to form the signaling tone.

Each component is defined by the following parameters: (i) *Control Register:* Defines whether the two sinusoids are to be added or modulated, the initial phase (0 or 180), and the frequency unit (f or $f / 3$). This flag is stored in a byte. The meaning of each bit in the control register is given in Table 2, (ii), *Component cycles:* This specifies the number of times the tone component (ON + OFF) in a tone cycle must be generated to complete the component, (iii) *ON duration:* Indicates the duration of the tone ON (in ms), (iv) *OFF duration:* Indicates the duration of the silence period (in ms) that follows the ON period, (v) *Frequencies* (in Hz) and (vi). *Levels* (in dBm0).

The high level architecture of a tone component generation is shown in Figure 1. The Tone Parameter Extraction block will extract the tone parameters (frequencies, cadence, modulation, phase and levels) from the tone tables, corresponding to the 'tone_id' of the tone to be generated. The Tone Component Controller block decrement the component count to be generated by 1 whenever ON and OFF samples to be generated is equal to 0. If number of components to be generated is equal to 0, then Tone Cycle Controller checks whether any more cycles need to be generated. But, if component count is greater than 0 then Parameter Estimation & Setting blocks will initialize the next tone component to be generated.

Table 2. Description of Control Register

MSB : 7	6	5	4	3	2	1	0: LSB
Reserved	Unit 2	Unit 1	Phase 2	Phase 1	Modulation		

Bit position	Bit description	Value	Description for the chosen value
Bit 0-2	Modulation type	0	No modulation, addition for dual frequency tone or single tone only
		1	Amplitude modulation with suppressed carrier
		2	Amplitude modulation with carrier
		3	Amplitude modulation with upper side band only
		4	Amplitude modulation with lower side band only
		5	Narrow band Frequency modulation
		6	Reserved for future
		7	Reserved for future
Bit 3	Phase 1	0	Frequency1 initialized with SINE wave
		1	Frequency1 initialized with COS wave
Bit 4	Phase 2	0	Frequency2 initialized with SINE wave
		1	Frequency2 initialized with COS wave
Bit 5	Unit 1	0	Frequency1 used without frequency division
		1	Frequency1 divided by 3. The unit bits produce frequencies with 1/3 Hz. For example, 16 2/3 Hz is generated with 50 Hz when the unit bit is set (16 Hz = 50/3 Hz).
Bit 6	Unit 2	0	Frequency2 used without frequency division
		1	Frequency2 divided by 3
Bit 7	Reserved	Reserved	Reserved

The following parameters are initialized for the generation of a tone component (i) *generate_cycles* is initialized with number of cycles to be generated, (ii) *component count* is initialized with number of tone components in the tone cycle, (iii) *repeat_comp*, which describe number of times tone component need to be repeated in a tone cycle, is initialized with number of times the current component in generation need to be repeated for completing the component (iv) *Resonator constants* (hereafter called as oscillator co-efficients) are estimated for the each frequency of the tone component and are initialized, (v) *ON* and *OFF* samples to be generated are initialized with converting ON and OFF duration of tone component into number of samples (vi) *Amplitude* is initialized based on the power level of the frequency. The amplitude conversion is done from dBm0 value as per equation (5), (vii) *Modulation scheme*, waveform type (sin or cos) and dual tone generation are initialized based on values of appropriate bits in the control register and (viii) *Oscillator initial memory parameter* $x(n-2)$ and $x(n-1)$ are initialize based on phase angle of each frequency.

Tone Component Generation block calls respective tone generation algorithm depending on number of frequencies in a tone component for generating tone samples during ON time and fills zero for generating silence during OFF time. Component Cycle Control block decrement the repeat_comp by 1 and signal Parameter Estimation & Setting block to initialize the state for present tone component for repeating if the repeat_comp is greater than 0 with ON and OFF samples to be generated is 0. If the repeat_comp and also ON and OFF samples to be generated are equal to 0, the Tone Cycle Control block decrement the component count. If number of components to be generated is equal to 0, then Tone Cycle Controller decrement generate_cycles and checks whether any more cycles need to be generated. If generate_cycles is greater than 0, it signals Parameter Estimation & Setting block to initialize the state with first parameters of tone component.

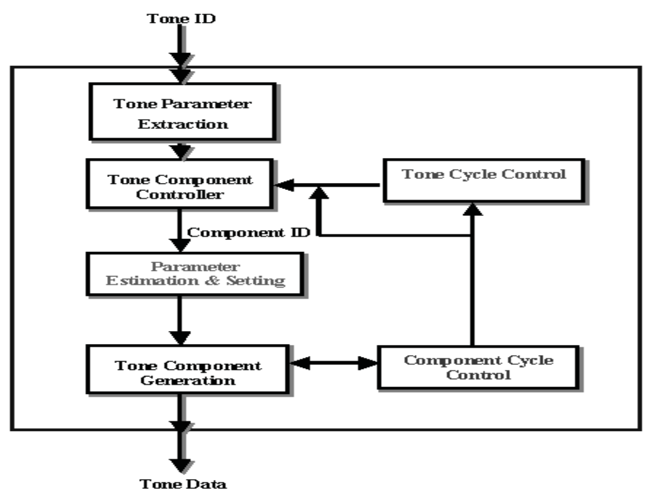

Figure 1. Top-level architecture of generic TG

4 PERFORMANCE EVALUATION

The performance of the algorithm is evaluated by performing the following tests: (i) Frequency accuracy test, (ii) Power level accuracy tests, (iii) ON-OFF time tests, (iv) RFC events generation tests and (vi) RFC tone generation tests.

A Hanning-windowed Fast Fourier Transform (FFT) is used to measure the output frequency. The typical frequency accuracy requirement is about ±1% of the nominal frequency. For an 8 kHz sampling rate, at least 4096 samples (512 ms) are required to ensure a good frequency resolution. The valid range of power level can be 0 dBm0 to –63 dBm0. If power level is 63 (representing –63 dBm0), the filter states are initialized with zeroes so that no tone samples are generated for this frequency. If a single frequency is generated with the modulation flag set, the resulting signal will be silence. For a continuous tone the component is repeated with a longest possible duration for performance reasons. For tone consisting of two or more frequencies, the effective level will not be equal to a simple addition of the two levels. If N sources generating the same sound level, are combined, the overall Sound Pressure Level (SPL) will increase by 10 log N dBm0. When one tone level in a combination is greater that the others, the resultant can be estimated from the following Table 3.

The frequency accuracy of the algorithms designed is shown in Figure 2 & 3. The worst-case frequency precision observed is 0.3%. The test results confirm the high accuracy of the algorithm design.

Table 3. Effective power level of combination tone

Difference in the levels	Add to Higher Level (dBm0)
0 – 1	3
2 – 3	2
4 – 9	1
More that 10	0

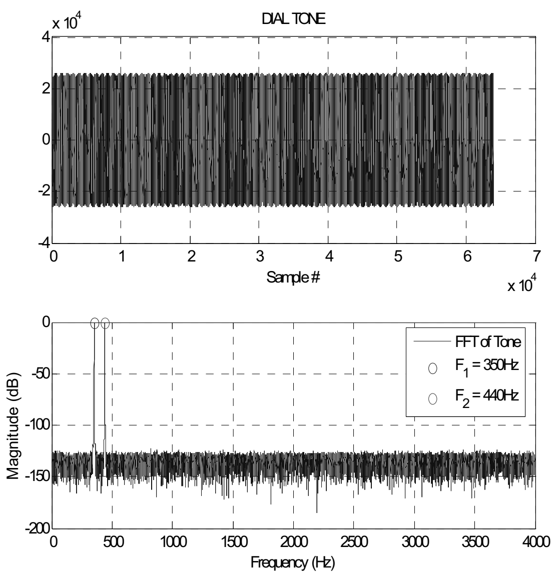

Figure 2. Dial tone (a) Time domain (b) Spectrum

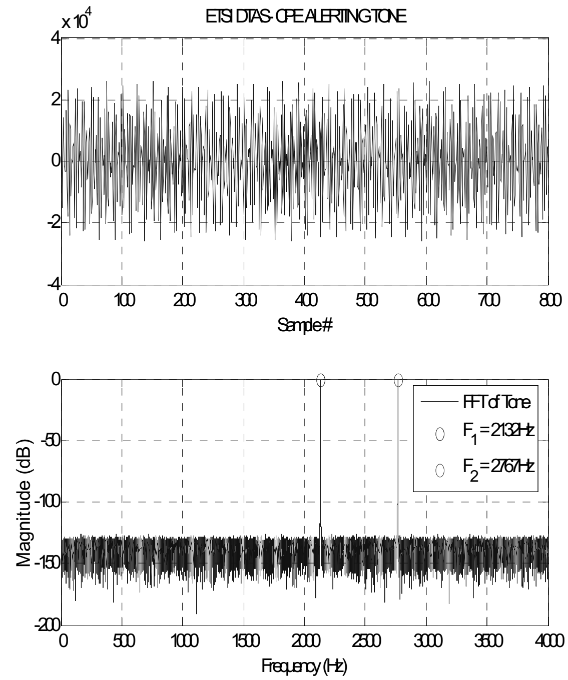

Figure 3. ETSI DTAS (Dual Tone Alerting Signal) Tone (a) time domain and (b) spectrum

5 CONCLUSION

A simple and generic tone generator design is discussed in detail and performance evaluation tests confirm the high accuracy of the design. The implementation on ARM9E core confirms low power consumption of the tone generator designed. All the features of tone generator are tested and found to be in compliance with the respective tone's ITU-T guidelines. The generic tone generator may be configured easily for any kind of tone generation in the telephony frequency band, due to the versatility and simplicity of the digital oscillator.

6 REFERENCES

[1] ITU-T Recommendation E.180/Q.35, “Technical characteristics of tones for the telephone service”.

[2] ITU-T Recommendation E.182 “Application of tones and recorded announcements”, Vol. II.

[3] ITU-T Recommendation E.181 “Customer recognition of foreign tones”, Vol. II.

[4] CCITT Recommendation V.23 (1988): "600/1 200-baud modem standardized for use in the general switched telephone network".

[5] ETS 300 001 (1996): "Attachments to the Public Switched Telephone Network (PSTN); General technical requirements for equipment connected to an analogue subscriber interface in the PSTN".