Echo Cancellation Demystified

The Need for Echo Cancellation

People have been using phones as a means of distant voice communication for more than a century now. Using phones has become sort of a usual thing. We use the phones almost every day and just about everywhere: at home, at work, outside, in our cars and so on. A big thanks goes to the cellular phones, which set us free from wires!

Although we all enjoy this remarkable possibility to talk on the phone, there’s always something we’d like to be better. This something is the speech quality in the phone conversations. The speech quality has always been an issue for both the phone network service providers and their subscribers. There are several reasons for the undesirable quality degradation and the appearance of audible echoes is one of them. This kind of quality degradation is inherent in the network equipment and the end-user phone devices.

Today it is easy to implement echo cancellation on DSPs and this is what engineers are doing in their devices. However, many of them face certain difficulties with achieving echo cancellation because of incomplete understanding of the echo cancellation principles and not meeting the requirements imposed by the echo cancellers. The purpose of this article is to demystify the topic of the echo cancellation by explaining its basics and providing useful information for those engineers, who need to implement the echo cancellation in their devices. We shall see where the echoes come from, how to fight them and what the known problems with the echo cancellation are. The information provided herein is based on the experience of developing echo cancellers and supporting echo canceller customers at SPIRIT Corp.

Where Does the Echo Come From?

There are generally two kinds of the echo, which can appear when talking on the phone. The two differ by the place where they are created and by their characteristics.

Hybrid Echo

The first kind is the line echo (also known as electric or hybrid echo) and it is created by the electrical circuitry connected to the wire lines.

Let us first see a simplified version of a network with two abonents.

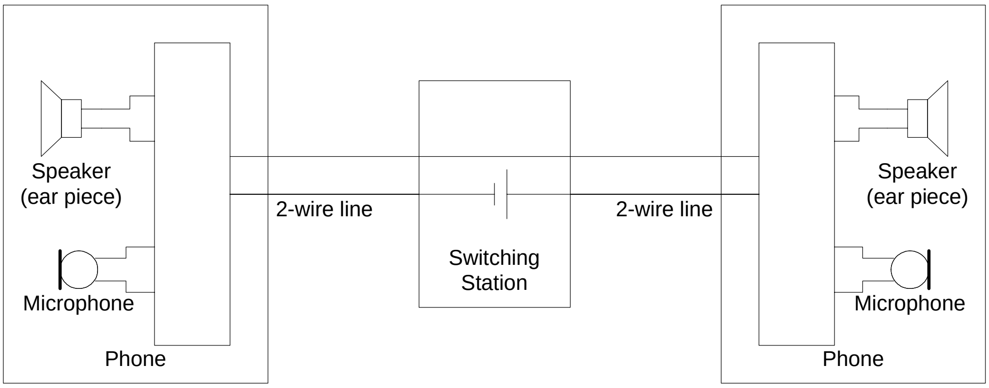

Figure 1. Simplified 2-wire phone network with 2 abonents

On the Figure 1 you can see the network using 2-wire lines to connect the abonents' phones with the switching station. Each of the 2-wire lines between a phone and the switching station carries voice signals in both directions, e.g. from one phone to the other through the station and back. The switching station provides the power supply to feed the microphones and the switching functionality, which is needed if there are more than two abonents.

The 2-wire lines are obviously cheaper than the 4-wire lines and this is why the regular phones and switching stations were designed to operate with each other over 2-wire lines.

The above network is indeed simple and it can operate very well provided the distance between the abonents is short. Now, if we want to make calls between very distant abonents, we need to do something about the signals because of their attenuation in the long analog lines. So, we need to amplify the signals. But we can't just amplify what is being sent and received over the 2-wire line because there are both signals coming in both directions at the same time. The solution to this is amplification of separated send and receive signals from the 2-wire line. Such a separation is performed by a dedicated electrical device, called a hybrid. The hybrid basically provides a conversion between 2-wire and 4-wire lines (the switching stations are now connected with 4-wire lines). If instead of the 4-wire analog lines we use digital channels, we also need to separate the signals so they can be independently digitized and efficiently compressed before transmission. Digital transmission improves the quality of the calls and increases the capacity of the phone networks, which leads to a more efficient use of the network equipment and allows more abonents.

Let us now see a phone network connecting the distant abonents:

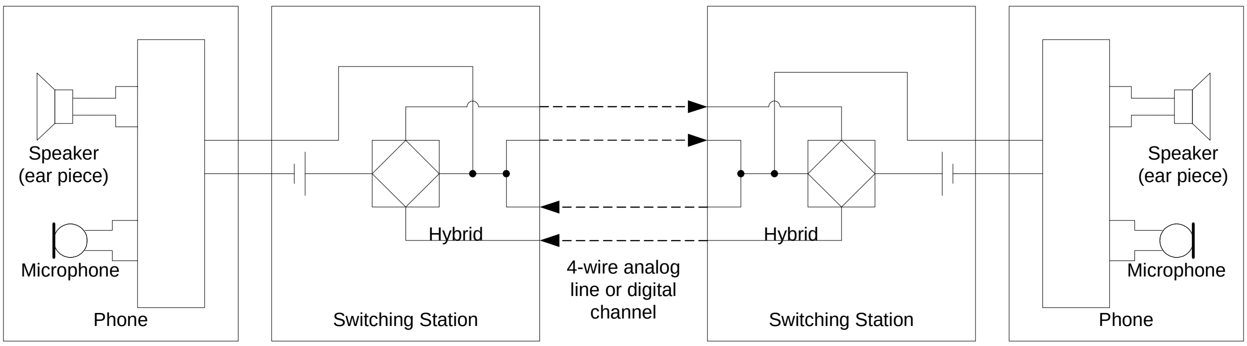

Figure 2. Long-distance phone network with 2-to-4-line conversion

Now, the interesting part is the hybrid performance because it is the hybrid where the echo can be and, in fact, is created. Ideally, the hybrid should just have a sum of the send and receive signals on the 2-wire side and these same signals separated on the 4-wire side. But in the reality, there are things like spread of equipment parameters and mismatch of line impedances, which all contribute to imperfect signal separation in the hybrid, which is the cause of the echo creation:

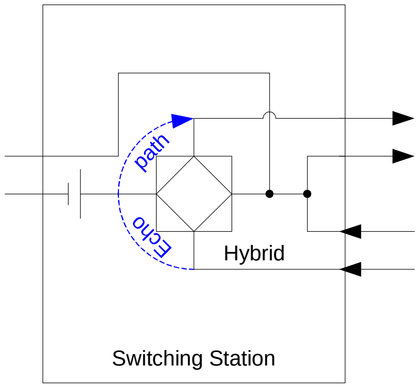

Figure 3. Hybrid echo

So part of the signal being sent to the hybrid on the 4-wire side is returned back as the echo superimposed on the signal being received from the hybrid on the 4-wire side. If, for example, the left hybrid of the Figure 2 has this kind of impairment, then the right talker will be hearing his own voice in the handset as an echo and the more the distance (and thus the signal delay) between the abonents’ phones, the better this echo will be audible.

Since the hybrid echo is inherent in the designs involving 2-to-4-wire conversion, we always need to cancel this echo in the switching stations and any other devices having this kind of conversion.

Regular phones, which are connected by 2-wire lines with the switching stations, may also have this kind of conversion, but there’s an excuse for not doing full-blown echo cancellation in the regular phones. The delay in the electrical path between the microphone and earpiece (or speaker) in the phone is essentially zero, so a cheap transformer-based attenuator can be used, and hearing your own undelayed voice of low amplitude does not cause much, if any, of the discomfort. Actually, hearing talker’s own voice is desirable as people expect to hear themselves but being not able to do so makes them think that the phone is not working.

However, using echo cancellation is required in the hands-free phones and all phones, which amplify the signal right before the earpiece or loudspeaker. Not doing echo cancellation in such devices leads not only to very well audible echoes for those who make calls to such phones (acoustic echo cancellation will be treated in the following section), but also to self-excitation of the amplifier. The self-excitation results in from non-ideal signal separation in the phone’s hybrid, e.g. part of the signal from the microphone reflects at the hybrid to the other signal path and gets amplified by the amplifier, so what the microphone is picking can be heard from the speaker. The acoustic feedback between the amplifier’s output and input effectively turns the amplifier to a generator. Therefore, the hands-free and all other amplifying phones (for example, phones for people with hearing impairments, who tend to speak louder) must include line echo cancellers.

Unlike the phones, the dialup modems and faxes always employ built-in echo cancellers to combat the local echo, because these digital devices are much more sensitive to the distortions of the received signals than humans. The same echo cancellation may be desirable in the answering machines, which record the voice from the phone line.

There are a few peculiar properties of the hybrid echoes. One is that the echo path delays are very short and each hybrid has a single echo path. Another is that the echo paths don't change or change very slowly over time because of very slow changes of the electrical circuitry parameters and wire lines parameters in the network.

Acoustic Echo

The second kind of the echo is the acoustic echo. It is easier to understand why and where this echo is created, although as we will see later, this doesn't make it easier to efficiently cancel it.

The acoustic echo is created by the loudspeaker in a phone. The sound comes out of it, bounces the walls, ceiling and other objects in the room, reflects and comes back to the phone's microphone. The same thing is possible to have not only in the buildings, but also in cars, basically, everywhere, where the sound from the loudspeaker can be reflected to the microphone, and this also includes the phone's case as the sound can and, usually does, go from the speaker to the microphone inside the hands-free phone! Similarly, if there's bad acoustic decoupling between the microphone and earpiece in the handset, the acoustic echo will exist in the handset, no matter whether it's a regular or cellular phone.

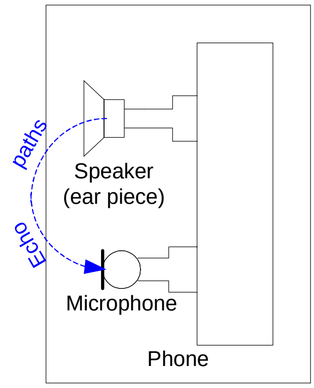

Figure 4. Acoustic echo

Often times, when making conference calls at workplace, we use the hands-free features of our phones, so all of the colleagues participating in the call can hear the other side. The acoustic echoes differ a lot from the hybrid echoes. First of all, the echo path delays aren't short (the echo path delay is the echo path length divided by the wave propagation speed. Electromagnetic waves propagate at about the speed of light in the wires, e.g. $3*10^8$ meters/second, while the sound propagation speed in the air is about $3*10^2$ meters/second. As you can see, the difference is 6 orders of magnitude!). The echo path is determined by the size of the room where the phone is used, and, obviously, the more the room size is, the longer the echo path delay is. And if we don't cancel this acoustic echo, the person who has called to the hands-free phone can hear a very annoying echo, which is delayed by the sum of the acoustic echo path delay in the room plus the round-trip delay in the network between the phones. But longer echo path delays aren't the only interesting feature of the acoustic echo. The other interesting thing, which imposes certain problems on the acoustic echo cancellation, is that there are many echo paths available in the room as the sound now can be reflected by many objects to the microphone and the paths can vary over time as the objects change their locations. Suppose you move around the room or somebody opens or closes the door in it. This makes the effective echo path change.

Approaching Echo Cancellation

As the exact network and room echo paths (and their impulse responses) are generally unknown, there's no other simple means to remove the echo but an adaptive system.

Let's look at how in general the echo cancellation can be done for one direction of transmission. We will explain this on the example of the hybrid echo cancellation, most of which also applies to the acoustic echo cancellation.

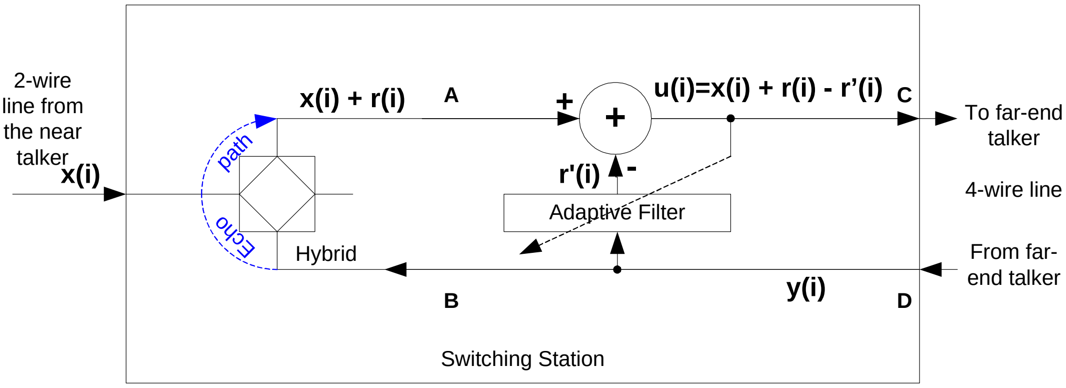

Figure 5. Switching station with hybrid echo canceller (common wires and microphone power supply not shown)

On the above figure we see the switching station with an echo canceling system integrated on the 4-wire side between points A, B, C and D. The signals (all shown as functions of the sample number, i) are as follows: x(i) is the signal from the abonent connected by a 2-wire line to the switching station (near-end talker signal), y(i) and u(i) are the signals from and to the other abonent (far-end talker), which come through the 4-wire line.

The idea of an echo canceller is simple. The signal from the far-end talker, y(i), when passing through the hybrid's echo path (between points B and A) is affected by the echo path's impulse response and is transformed to the signal r(i), which is the undesired echo. The signal from the near-end talker, x(i), is added to r(i) at point A. The adaptive filter (normally, FIR) used in the system mimics the impulse response of the hybrid's echo path and produces a replica, r'(i), of the echo signal r(i). If r(i) and r'(i) are the same, then they will cancel each other in the summer connected between points A and C and the filter's output. If r(i) and r'(i) aren't the same, the far-end talker will be hearing not only the near-end talker's x(i) signal, but also the difference of r(i) and r'(i), which is called the residual echo error signal.

The residual echo error, e(i) = r(i) – r'(i), is used to adapt the filter's coefficients. That is, the echo canceller is a tracking system with the residual echo error signal used as the feedback and the purpose if this system is to minimize this error.

Obviously, since the echo path's impulse response is unknown, some time is needed for the echo canceller to minimize the residual echo error signal below a required level. This time is called the convergence time. Note that while the far-end talker's signal y(i) is equal to zero, the echo canceller is not able to converge, because both r(i) and r'(i) are zero, thus the feedback is also zero and there's no adaptation possible. This is why the reference signal y(i) should be present in the beginning of the conversation; grabbing the handset and just saying "hello" should be more than enough for the adaptation to proceed.

Note that the adaptation is possible to do when the near-end talker's signal x(i) is close to or is zero, otherwise this signal x(i) will effectively be an additive noise in the feedback, causing the system to become unstable, diverge and stop working. This is why no filter coefficient adaptation is done at all or the adaptation is very slow during the double-talk periods, e.g. when both the near-end and far-end talkers talk simultaneously.

It is worth mentioning that such an echo cancellation scheme is naturally linear and is very sensitive to the nonlinearities in the echo path. This linear system will not be able to match the impulse response of a nonlinear echo path and therefore effectively remove the echo.

Good echo cancellation performance can be achieved by using the NLMS (Normalized Least Mean Squares) algorithm, which is also known as the normalized stochastic gradient algorithm, or its many variations. The NLMS algorithm is the most widely used one and it provides a low cost way to determine the optimum filter coefficients. The algorithm minimizes the mean square of the residual echo error signal at each adaptation step (e.g. at each sample), hence the name of the algorithm. Normalization by the signal power is used because speech is a highly non-stationary process.

Without derivations, which you can find elsewhere in the literature on adaptive signal processing, we give the general formula for the coefficient adaptation for the NLMS algorithm:

$$a_k(i+1) = a_k(i) + 2 \cdot \beta \cdot \frac{e(i) \cdot y(i-k)}{N \cdot \sigma^2}$$

Where:
i is the sample number
a_k is the k-th coefficient of the filter
N is the number of filter coefficients
β is the adaptation step, which controls the convergence time and adaptation quality
e is the residual echo error signal
y is the far-end talker signal
σ^2 is the reference signal power

The number of coefficients in the filter should be large enough to cover the echo path delay and all additional delays due to the lines and circuitry between the echo canceller and the place where the echo is created (e.g. the total delay between points A and B of the echo canceller as per Figure 5). This should also include the dispersion time due to the network elements.

The hybrid echo path delay is known to be short. The time span over which its impulse response is significant, is typically 2 to 4 milliseconds, but usually when canceling the hybrid echo, the number of filter coefficients is chosen to cover hybrid echo path delays up to 16 milliseconds, which is usually the upper bound of the hybrid echo path delay. The 16 ms path needs 128 coefficients at the sampling rate of 8 KHz.

The length of the acoustic echo path, as has already been pointed out, depends on the size of the room, where it exists. So, without going into room measurements, for acoustic echo path delays of up to 256 ms we will have to have 2048 coefficients at the sampling rate of 8 KHz.

Echo Canceller Performance

The NLMS-based echo cancellers for both hybrid and acoustic echo canceling do exist and perform well, however, acoustic echo cancellation is more complex due to the specifics of the acoustic echo paths and the need for the acoustic echo cancellers to operate in the presence of noise in the echo path (examples: noise in the car, noise in a crowded room). For these reasons a number of enhancements has been proposed and implemented in the acoustic echo cancellers (AECs) by researchers.

There are certain improvements possible when employing a frequency-domain AEC. As you should have already realized, the NLMS algorithm presented earlier entirely operates in the time domain, no work is done there on the spectrum.

The first problem with time-domain AECs is their resource requirements, MIPs. Normally, the AECs need to have many filter coefficients to efficiently cancel the acoustic echo. But doing long convolutions to generate the replica of the echo signal is expensive when doing them directly in the time domain. It is possible to modify the initial NLMS algorithm so that the filter coefficients are updated once per a block of samples y(i)…y(i+N) instead of doing that each new sample y(i). The NLMS algorithm such modified is called the block NLMS (or BNLMS) algorithm. The advantage of keeping the coefficients fixed during the block of N samples y(i) is that it is possible to replace the time-domain convolution by multiplication in the frequency domain. The direct convolution computation in the time domain has a cost proportional to N^2 (e.g. the number of multiplications, if we compute it for N samples and there are N coefficients in the filter). Frequency domain processing requires computation of several Discrete Fourier Transforms (DFTs) of the signals. The Fast Fourier Transform (FFT) is a very efficient DFT implementation and it is known to have a computational cost proportional to $N*\log_2 (N)$. So it is more efficient to use convolution by FFT for big Ns. The disadvantage of using BNLMS is that it has slower convergence because of rare adaptations. Also, such an echo canceller introduces a processing delay. Hence, with BNLMS we have a tradeoff between the convergence time, delay and cost. It should be noted, however, that FFTs require additional memory, so, we also trade memory for MIPs. Because of slower convergence, processing delay and bad noise immunity, BNLMS echo cancellers aren't popular.

In practice, other frequency-domain echo cancellers are used, such as those based on Multidelay Block Frequency Domain Adaptive Filters. This kind of sub-band processing in echo cancellation solves most (if not all) problems of time-domain NLMS algorithms and their variations. Frequency-domain echo cancellers also solve the problem of poor performance in the presence of noise, which is especially important for the AECs that need to work in the cars, crowded rooms or otherwise noisy locations. When implementing the AECs with frequency-domain analysis and synthesis blocks, it can be possible not only to reduce the computational cost, but also reduce the processing delay that would be present in BNLMS, have better noise immunity and even suppress the noise by frequency-domain noise suppressors directly integrated in such AECs. Doing so will greatly improve the quality of speech in the end. Also, with frequency-domain processing, it's possible to have better immunity to the nonlinearities in the echo paths because the AEC will adapt to the strong fundamental frequencies, while their weak harmonics can be suppressed as part of noise. This all is impossible to achieve with time-domain AECs.

Finally, it is important how the echo canceller behaves in the double-talk situations, when both talkers talk simultaneously. Obviously, the parties prefer to hear each other throughout the entire conversation and hear little or no echo during double-talks. Therefore, the echo canceller's double-talk performance should also be addressed when designing and testing echo cancellers or simply choosing which one to integrate to the phone.

Frequency-domain echo cancellers are very effective in canceling acoustic echoes. Unlike time-domain AECs, they need fewer DSP MIPs, perform better in double-talk situations, work well in presence of noise, can have embedded noise suppression almost for free and perform better with nonlinearities in the echo path. This is why frequency-domain echo cancellers should be preferred over time-domain ones for canceling acoustic echoes.

Failing to Achieve Echo Cancellation

Unfortunately, it is not uncommon for engineers, integrating echo cancellers in their devices, to make mistakes, which cause LECs and AECs to cease work. In this chapter we will describe the most typical design and integration mistakes that lead to failures in achieving echo cancellation. That is, it can be not just a question of achieving good echo cancellation, but instead it can be a question of achieving the echo cancellation at all!

It is very important to understand and meet the requirements of the echo cancellers or they will just not work. To the engineers developing applications with echo cancellers it means that they will need to redesign their product and change the integration of the echo canceller. Speaking in business terms, this will incur higher product costs and longer time to market. And all of this can and should be avoided at best!

Nonlinear Distortions in Hardware

The first thing, which can lead to echo canceller performing very poorly, is the nonlinear distortions in the echo path of the hardware of your device. The echo cancellers perform poorly or don’t work at all in systems with the net nonlinear distortions in the echo path higher than –16 dB (typical value). The smaller are distortions, the better.

The nonlinear distortions exist everywhere. Certain nonlinearity is inherent in the hybrids, microphones, speakers, amplifiers and DAC/ADCs (known as codecs). It is not recommended to make the design with parts, which are highly nonlinear such that the net nonlinear distortions of the device in the echo path are prohibitively high. If there’s some preliminary design available, already in a form of a working device, it is a good practice to measure the level of the nonlinear distortions in it. The sooner such measurements are done, the better.

Usually, in the systems, which are not strictly digital (e.g. those involving use of analog circuitry and transmitting analog signals anywhere inside), there’s an ADC/DAC available so the echo canceller implemented on a DSP can work with samples. The path between the DAC output and the ADC input has the analog circuitry, which is subject to nonlinearities. If we’re talking about using an AEC in some hands-free system, then the analog circuitry in question will include the following: microphone, microphone amplifier, the ADC/DAC itself, the loudspeaker amplifier and the loudspeaker. This entire echo path must be tested. An easy test for this would be feeding a test signal as samples to the DAC so the speaker would produce it and recording samples from the ADC, e.g. recording what the microphone is picking. The recording should then be analyzed.

To test the recording for nonlinear distortions you can use a test signal, consisting of two sinusoidal signals (for example, tones of 300 and 1800 Hz could be used). The recording must also contain these two frequencies but there will always be other frequencies in the spectrum of the recorded signal because of the tones undergoing nonlinear distortions in the aforementioned hardware. If you run this recording through spectrum analyzing software you will see all of these frequencies. Obviously, due to the distortions, there will be harmonics of each tone, e.g.

2*300=600 Hz and 2*1800=3600 Hz, and there will also be combinations of the two frequencies, e.g. sum and difference: 300+1800=2100 Hz and 1800-300=1500 Hz. We just showed nonlinearity, which of the second-order. If the nonlinearity is of a higher order, which is always the case in reality, then there will be many more frequencies in the recording. The main thing here is that the amplitudes of the original frequencies (e.g. 300 Hz and 1800 Hz) must exceed the amplitudes of all other frequencies by at least 16 dB (or equivalently, the absolute ratio of the amplitudes must be greater than $10^{16/20} = 6.31$). Presented is a very simple test and while it can reveal certain nonlinearities, a more thorough method should be used to measure the nonlinear distortions, please see the ITU-T recommendation O.42 for more information on this.

You should perform such or similar test to make sure your hardware is OK in terms of nonlinear distortions. If the testing shows that this is not the case, you must find and fix the problem before thinking of any AEC integration. To find whether or not the problem is in the ADC/DAC, use its analog loop-back mode when doing this test. Beware, the problem may have to do with incorrect ADC/DAC programming!

Also, the nonlinear distortions can be a result of limiting (clipping) the signal in either ADC/DAC or elsewhere, for example, in the amplifiers. If by expecting the waveform, which you recorded from ADC, you see that many of the samples values reach their minimum and maximum values, then you must attenuate the signal somewhere so the clipping doesn’t occur.

Note that there can be interference between the digital and analog parts in the device. The interference may be in form of additive noise superimposed on the Vcc if the power supply is overloaded or there is no good power supply decoupling. The decoupling capacitors must be placed as close to the power supply pins of the chips as possible.

Microphone and Speaker Placement

As it has already been mentioned, the acoustic echo exists between the loudspeaker and the microphone in hands-free phones inside their cases. The echo can be transmitted by both the air inside the case and by the case itself in a form of mechanical waves (vibrations) in the case parts. To reduce this form of the echo, there should be a good acoustic decoupling between the loudspeaker and the microphone. The microphone should be acoustically and mechanically insulated by a soft material, absorbing the case vibrations and sound coming out of the speaker. The microphone should not be directed to the speaker. It can be useful to have a directional microphone, so it can be directed away from the speaker.

Another important thing is the external echo path (e.g. outside the phone’s case). The external echo path is actually a number of different echo paths due to the room objects reflecting the speaker’s sound back to the microphone. It has also been noted that these echo paths vary with time as the objects or people move in the room. The changes in the echo path impulse response cause an increase in the residual echo error signal. This forces the AEC to start adapting to the new impulse response and it can even diverge, if the changes are fast or abrupt. In the installed phone, the speaker and microphone should not be directed to the path that is subject to fast changes. It is usually better to direct the speaker and microphone towards the ceiling since this echo path changes rarely.

Input and Output Signal Requirements

Besides the main hardware questions like the nonlinear distortions, there are also certain requirements on the signals, which are fed as samples to the echo cancellers.

The first requirement is that signal delays in software be as short as possible. In general, there should be no signal processing done between the codec and the echo canceller. And there must be no sample accumulation without any good reason for it. Excessive buffering will increase the effective signal delay in the echo path and therefore the utilization of the filter coefficients will be ineffective (some of the coefficients will have to cover the additional delays yet they will be zeroes). Obviously, the reference signal, y(i), delay in software must be smaller than or equal to the software delay for the signal with echo, x(i)+r(i). If this is not true, the echo canceller will not be able to converge and cancel the echo since it doesn't have the reference signal, which is to be subtracted.

Attention must be paid to the signal delays in the software in another respect. All of the delays must be constant throughout the entire session, in which echo cancellation is desirable. Changing the delays during a phone call will cause the echo canceller to diverge and stop canceling the echo until it converges again.

It is also possible to have other problems with signals in software. Echo cancellers usually process linear PCM samples, while the signals in memory or received from the codecs may be compressed to A-law or μ-law samples. Make sure the echo canceller is receiving the samples in the format, which it was designed for. And don't artificially clip the samples on the way between the echo canceller and codec. This all will only contribute to the undesired nonlinear distortions.

Incorrect Codec Synchronization

The last but not least problem with echo canceller integration can be again due to the hardware or software design mistakes. What is the problem of incorrect codec synchronization? Well, the problem is easy to understand and relatively easy to solve, provided we know the right solution to it.

Suppose we have a device, which has several different signal sources, each clocked at a different rate, and the signals from one must go through the device to the other. Where is this possible? This is possible in hands-free phones, which have a pair codecs. One of the codecs is used to interface to the phone line and the other one is used to interface to the loudspeaker and microphone.

The problem here is that if both codecs are clocked at different rates (say both have sampling rate at about 8 KHz but they're not exactly equal because they're clocked from different quartz oscillators), then we can't just take each sample from one codec, somehow process it and pass to the other codec. Eventually, the sampling rate difference will lead to either sample accumulation somewhere in the sample buffers or sample depletion, e.g. there will be nothing to take out of a buffer when a sample is needed.

The first solution is to choose the codecs such that they're clocked from the same clock source, the same quartz oscillator. This is the best solution to the problem and with little provision on the hardware design stage the problem can be completely eliminated. Even if the specifics of the application does not allow for use of the same codecs in both places, it is still better to have the same clock source for both because this will make it possible to use sample rate conversion with a constant upsampling and downsampling ratio and there will be no synchronization issues.

But if codec synchronization via the same clock source is not possible to achieve (as is the case with ISDN phones, where the data rate is not anyhow related to the codec clock), then some different solution is needed.

Often the engineers are tempted to solve this problem using one of the following solutions:

- Continuously tuning the codec's sampling rate
- Dropping samples received from the codec and repeating samples to be sent when there's nothing to send

But our experience and logical reasoning proves these solutions wrong as they fail to solve the problem they're supposed to. And here's why…

The first solution is not viable because it incurs additional nonlinear distortions in the echo path and also effectively changes the echo path delay. The second solution is not viable because using such an approach we will be abruptly changing the echo path delay. Changing the echo path reduces the quality of echo cancellation and can even force the echo canceller diverge if the residual echo error becomes too big. The worst case is the double-talk situation, e.g. when both the near and far-end talker signals are present. In such situations the echo canceller usually doesn't adapt the filter coefficients or adapts them very slowly. If the echo path remains constant during the double-talk, the echo canceller performs well, but if the echo path changes, the echo canceller will not be able to adapt to these changes and it will diverge. So, if we want the echo cancellers to operate, we can't use any of these non-solutions. Neither.

A solution to this problem is an adaptive sample rate converter, or simply an adaptive interpolator. It must be placed between the codecs (or the codec and the ISDN interface). Actually, there are two of them needed, one for each signal direction. The interpolator should be initially tuned to do upsampling or downsampling from one frequency to another if they're known to be different (for example, they can be 8 KHz and 9.6 KHz, so the interpolator will know what interpolation is done). As the time goes, it is possible to see the actual rate at which each codec transfers samples. The difference of the rates can be used as a feedback to adapt the interpolator to the actual ratio of the sampling rates.

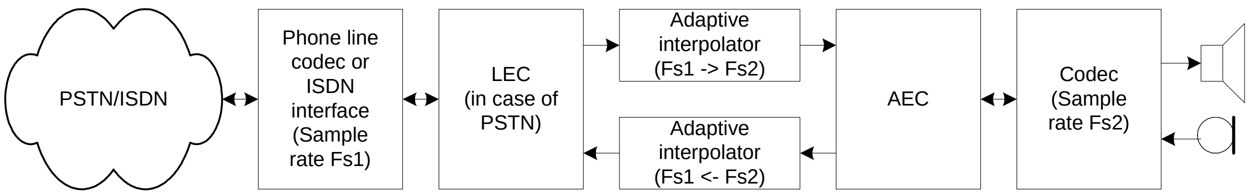

Figure 6. Codec synchronization via adaptive interpolators

This solution is schematically shown on the above Figure 6. It is important that the echo cancellers be connected to the codecs directly and running at the rate of the codec they're associated with. Placing interpolators between the codec and associated echo canceller will turn this solution into a non-solution described earlier.

Echo Cancellation on a PC

Echo cancellation on a PC, equipped with a digital sound card, microphone and active speakers should be possible to achieve in principle, but it's not always feasible. It's an interesting topic of its own; so let's see what problems can arise when trying to get the echo cancellation work on a PC and how to cope with them.

The vast majority of users of PC multimedia hardware such as sound cards, microphones, speakers and amplifiers are PC gamers. Often, the digital sound cards are used only for output in games and music players. This unfortunate practice allows making low-quality hardware, which is perfectly suitable for the outlined applications, and it is cheap. To find out whether or not the hardware is low quality, it's possible to carry out a nonlinear distortions measurement as we suggested earlier. Replacing of the microphone or active speakers with better ones can improve the overall quality of the system.

One problem with sound I/O on the PC is that the input and output can be independent and may even have different clock sources so each one can have its own sample rate. This is a very undesirable feature, which leads to changes in the echo path delay. It's somewhat similar to the problem of the codec synchronization and solutions analogous to adaptive interpolators can be used.

The hardware isn't the only place, where the problems (like nonlinear distortions and others) can appear. Another place is the software, which controls this hardware, namely, the device drivers and operating systems. The drivers for digital sound cards can be incorrectly implemented in that they may lose the samples. The end-user desktop operating system running on the PC can make a considerable contribution to this problem as well. Often, the desktop operating systems and their software, which can't work as part of a real time system and meet certain deadlines, lose responsiveness to I/O and processing requests on heavily and even moderately loaded PCs. While this can be tolerated in many applications such as games or mp3 players (a few lost samples will go unnoticed!) and the like, the echo cancellers will simply fail to do what they're supposed to. Possible solutions to this problem include closing all CPU-intensive applications and services in the operating system, using real-time operating systems (if applicable) and upgrading to a faster PC.

All of the above issues make echo cancellation on PCs problematic because of not meeting the basic requirements imposed by the echo cancellers.

Requirements Summary

Now that we have analyzed the typical problems with echo cancellation and pointed out several design and integration mistakes, we can summarize the requirements imposed by an echo canceller:

1. The nonlinear distortions in the hardware (microphones, speakers, amplifiers and codecs) must be sufficiently low, typically less than –16 dB.
2. The microphone and loudspeaker placement and echo insulation in the hands-free phones should be done carefully so as to reduce the phone's internal echo from the speaker to the microphone and be less sensitive to the echo path change outside the phone case.
3. The signal delays in the software must be fixed throughout the entire call session and they must be minimal, so the entire echo path delay is minimal and constant too.
4. Codec synchronization (if required) must be done correctly to avoid any sample losses and unbounded accumulation or depletion of samples in the software buffers leading to overflow problems and echo canceller divergence.
5. The operating system, device drivers and the rest of the software running in the system must meet the real-time requirements of the echo canceller.

Testing Echo Cancellers

It is a good practice for the customer to ask the echo canceller algorithm supplier how well their echo canceller conforms to the appropriate ITU-T recommendations (which are de-facto standards) and provide these figures alone with the resource requirements so a right decision can be made when choosing an echo canceller. The related ITU-T recommendations are: G.168 for LECs and G.167 for AECs.

It is beneficial for the customer to understand the basics of the echo cancellation and maybe even be familiar with the listed recommendations, however, it always makes sense to make a few tests of the echo canceller of interest. If a live test is possible, which is very desirable for AECs, it is good to make it. The echo canceller suppliers should provide a test or demo suit and a few test waveforms (the reference signal y(i) and the signal with the echo, x(i)+r(i) as per Figure 5), on which the echo canceller can be tested. Such a test can be carried out on either a PC or the customer's target hardware, whichever is arranged. This ensures the echo canceller operation and the suit can also be used to test the echo canceller performance on specific waveforms if the customer has any concerns about particular cases. It's also a good thing to test double-talk performance of the echo canceller to make sure the quality is delivered to the end users.

By the time the echo canceller integration is about to start, the hardware of the target device must have a sufficiently low level of nonlinear distortions. Only after having fixed all of the hardware problems, the echo canceller integration should begin. As soon as the echo canceller integration is finished, the echo canceller test can be repeated in full real-time with true I/O instead of file processing. Should there be any quality problems, the hardware and software must be checked against possible violations of the requirements imposed by the echo canceller, which have been stated earlier.

Conclusion

As we have seen in the preceding sections of this article, there are many possible problems, which can arise when designing and implementing a system with an echo canceller. But there is no black art or any other magic behind the failures. The reasons for them are well known and perfectly consistent with the echo canceller internal organization and requirements. To prevent delays in the development and reduce the costs, consider designing the system to meet the requirements at the very beginning. Redesigning the whole system at the middle or last stage because of not meeting the requirements will be expensive.

Solid understanding of the basics of the echo cancellation and meeting the general requirements imposed by echo cancellers will avoid all of the echo canceller problems and therefore shorten the development time and product costs, which is always desirable.

The engineers at SPIRIT Corp hope that this little investment in the form of an article will make a good service to all parties interested in canceling echoes in their products.

Alexey Frunze
Software engineer, SPIRIT Corp