Network Talk: Voice Over IP

Voice over IP (VoIP) has a big place in the future of the Internet. This article tells you what software is required to support VoIP and how the system should be architected.

In traditional telephony applications, speech is digitized and delivered over a circuit-switched network. During call setup, a dedicated amount of bandwidth is reserved for each phone call. In a VoIP system, voice signals are transported as packets over a packet network with no bandwidth dedicated to their delivery. The processing involved in a VoIP system is, therefore, quite different from the circuit switched delivery operation.

This article looks at a typical VoIP system architecture and describes the various software modules that must be included. We will also look at estimates of code size and required processing power for each module, as well as some guidelines for partitioning the system between a traditional CPU and a DSP.

VoIP technology offers several advantages in the telecommunications market:

- Effective bandwidth utilization, because the same channel can be used for voice packets going to different destinations
- A lower bit rate per voice channel through vocoders. For example, G.729a results in a sub-15Kbps rate (including overhead) versus a 64Kbps A/µ-law based voice bit stream
- Simultaneous voice/data/fax transmission
- Reduced costs for long distance voice calls and facsimile transmissions

Architectural considerations

A typical VoIP system has two interfaces to the outside world, as shown in Figure 1. The first interface is to a telephone or handset and the second to a packet net-
work (LAN or WAN). The telephone interface consists of a subscriber line interface card (SLIC) and a coder/decoder (codec). The SLIC handles the interaction with the handset, while the codec performs the analog-to-digital and digital-to-analog conversion of the voice signals. The packet interface can either be a WAN interface, such as ISDN, Packet Cable, or DSL, or a LAN interface, such as Ethernet.

Before sending voice data as packets, the analog signal from the handset microphone is sampled at 8kHz to generate a PCM-encoded digital data stream. Depending on the codec hardware setting used, the resulting bit stream has a bandwidth of either 64Kbps or 128Kbps. The digital signal is then processed by an echo canceller to remove the echo of the far-end received signal. This echo-free bit stream is then converted into frames of constant length by collecting samples for a definite time period, say 10ms. The frames are passed to the application layer, where a speech coder (also known as a Vocoder), such as G.729a, compresses the data. With sufficient digital signal processing, a 64Kbps bit stream can be compressed into a 6.4Kbps stream.

The TCP/IP protocol stack is then invoked to create the actual IP packets that will travel across the packet network. The compressed voice frame becomes the payload, with appropriate headers added at the real-time protocol (RTP), UDP, and IP layers. Once the packets are created, they can be transported over any suitable WAN or LAN data link and physical layers—Ethernet, for example.

On the receiving side, voice packets arriving at the network interface are depacketized, decompressed, and then played out at the handset speaker via the same codec/SLIC interface. This entire sequence of operations must be executed in real time for one or more voice channels, depending on the features of the product.

A VoIP system requires real-time processing at two points: when the samples are collected and when they are played back. If the sampling or playback period varies even slightly, the listener will hear something other than a continuous voice signal. To reduce the load on the CPU, a buffer is used for collecting the samples and transferring them into frames. A typical frame size is 80 samples (10ms of data). This frame must be compressed and converted into an IP packet during the 10ms before the next frame is created. Because voice packets are being received at the same time, those must be depacketized and decompressed for playback within the same time slot. Thus, both DSP and CPU functions must take less than 10ms for a complete cycle of compression/decompression and packetization/depacketization for one voice channel. When multiple voice channels are provided, the system becomes even more constrained. For example, a product with two voice channels must complete each set of tasks within 5ms maximum.

Since packet networks are asynchronous and the network bandwidth might be higher than the voice collection rate, a buffer must exist to store the received packets. These packets also need to be reordered, since they may arrive at different times, as the result of network congestion. Multiple voice channels require multiple buffers, making buffer management necessary.

**Software components**

The overall processing needed for VoIP is a combination of signal processing and protocol-based control processing. Tasks such as speech compression and decompression, echo cancellation, and tone detection require signal processing capability, while network packetization and depacketization and call setup and teardown tasks require protocol-based control processing capability. The required processing capability can be realized with two different processors, a DSP for the signal processing tasks and a traditional CPU for protocol implementation. A hybrid CPU/DSP processor can also be used.

VoIP systems require simultaneous execution of at least the following software modules:

- Collection of digitized voice samples, as well as playback at regular intervals
- Call setup/teardown and call control using H.323 or H.245 standards
- Vocoder algorithm such as G.723, G.729 and G.729a, G.711, or G.728 (interoperability with other VoIP systems may require availability of many vocoders in one system)
- Telephony support modules such as DTMF, call progress tones, and line and acoustic echo cancellation
- TCP/IP protocol stack
- LAN/WAN interface driver
- RTOS that provides services such as interrupts, timers, and buffer management

A variety of tones are used on the public switched telephone network for
dialing-dual tone multi-frequency (DTMF), call setup/tear-down, and caller ID. Tones are also used to indicate the call status and line conditions. A VoIP system must recognize and perform these tones to support a phone call. A VoIP system also needs to emulate central office functions such as the generation of call-progress tones used for conveying conditions such as ringing, busy, and dialing. The ITU standards Q.23/Q.24 specify the DTMF tones, while ITU standard E.180 specifies call progress tones for North America.

Echo cancellers are required to cancel echoes that arise as the result of impedance mismatches at the hybrid networks that connect two-wire links to four-wire links. Echo is a problem in VoIP networks because the round-trip delay is significant (longer than 50ms) and unpredictable. In designing an echo cancellation function, one must consider the echo tail length, speed of convergence, double-talk performance, and algorithm complexity. ITU G.168 defines performance requirements for echo cancellers.

Vocoders—paired speech compression and decompression algorithms used in IP telephony—are ITU standards. Examples include G.711, G.723, G.729, and G.729a. The differences between these algorithms reflect trade-offs between speech quality, network bandwidth utilization, computation complexity, and latency. Other features, such as voice-activity detection and comfort noise generation, need to be included to enhance speech quality and conserve bandwidth. Table 1 shows the performance of some of the popular speech codecs.

The ITU has developed a set of standards for multimedia communications over packet-based networks. These standards are aggregated under recommendation H.323. This recommendation describes terminals, equipment, and services for multimedia communications over packet networks.

### Voice quality

The voice quality achieved by a VoIP system depends on a number of factors including the quality of the phones, network congestion, and signal processing algorithms used. In particular, it depends on the following factors:

- **Vocoder**—The choice of the speech codec affects the intelligibility of the speech. The choice is based on a trade-off between voice quality, bandwidth computation, complexity, and latency
- **Echoes**—Imperfections in the network give rise to echoes that, in turn, affect the intelligibility of the speech. Echo cancellers are required to cancel the echoes and enhance the quality of speech
- **Latency**—Because delays in packet networks are significant and unpredictable, they can have a significant impact on speech quality. Delays arise because of buffering, processing, and congestion

For consumer applications, not only must a VoIP system’s performance be adequate, but the price of the product must be affordable. System cost has to be limited to a reasonable value to stimulate demand. Therefore, VoIP implementations must utilize cost-effective components.

<table>
<thead>
<tr>
<th>Vocoder</th>
<th>Bit rate (Kbps)</th>
<th>Score (MOS)</th>
<th>Sample size (ms)</th>
<th>MIPS</th>
</tr>
</thead>
<tbody>
<tr>
<td>G.711</td>
<td>64</td>
<td>4.4</td>
<td>0.75</td>
<td>&lt;1</td>
</tr>
<tr>
<td>G.723.1</td>
<td>6.3/5.3</td>
<td>3.6</td>
<td>30</td>
<td>30</td>
</tr>
<tr>
<td>G.729</td>
<td>8</td>
<td>3.9</td>
<td>10</td>
<td>30</td>
</tr>
<tr>
<td>G.729a</td>
<td>8</td>
<td>4.0</td>
<td>10</td>
<td>17</td>
</tr>
</tbody>
</table>
Hardware requirements

Several software modules that would typically be run on a DSP, including Vocoder middleware, are shown in Table 2 with their MIPS requirement and code size.

The PCM interface module captures the voice samples at the desired rate and stores them into the memory. It also supplies the voice samples to the CODEC for playback. In our current implementation, voice samples are generated at a rate of 8,000 per second. Eight voice samples are stored in the buffer and transferred into memory every millisecond. A timer set during initialization for this purpose provides an interrupt to the RTOS to start the transfer task. This task requires less than 1 MIPS. Since the RTOS also handles other interrupts such as network packet transfer and encoder initialization, it requires about 1 to 2 MIPS. This software is not included in the table, since it would typically be run on a traditional CPU, not a DSP.

When a call is initiated by the handset, the RTOS monitors the off-hook signal and starts the appropriate DTMF task. When a call comes from the packet network, it provides the ring signal to the handset and monitors the off-hook signal again. Tasks in this group continue to provide monitoring of other calls. In total, these tasks require 1 to 2 MIPS.

The line echo cancellation (LEC) task requires 5 to 14 MIPS depending on the tail length, which can be from 4ms to 32ms of the echo. Voice samples coming from the handset include the echo of the played voice. This echo needs to be cancelled at this point, otherwise it will play back at the origin. The tail length depends on the sending and receiving points. It is generally determined during the initial set up and then remains constant during

<table>
<thead>
<tr>
<th>Software module</th>
<th>MIPS</th>
<th>Code size</th>
</tr>
</thead>
<tbody>
<tr>
<td>DTMF, Call Progress Tones, and Caller ID support</td>
<td>2 to 3</td>
<td>15KB</td>
</tr>
<tr>
<td>Line Echo Cancellation</td>
<td>5 to 14 (4ms to 32ms tail length)</td>
<td>7KB</td>
</tr>
<tr>
<td>Vocoder</td>
<td></td>
<td></td>
</tr>
<tr>
<td>G.711 Comander</td>
<td>&lt;1</td>
<td>&lt;1KB</td>
</tr>
<tr>
<td>G.721 ADPCM</td>
<td>10</td>
<td>2.1KB</td>
</tr>
<tr>
<td>G.722</td>
<td>10</td>
<td>2.6KB</td>
</tr>
<tr>
<td>G.723</td>
<td>29.7</td>
<td>56KB</td>
</tr>
<tr>
<td>G.729</td>
<td>29.5</td>
<td>35KB</td>
</tr>
<tr>
<td>G.729a</td>
<td>16.5</td>
<td>32KB</td>
</tr>
<tr>
<td>Voice Activation Detection</td>
<td>approx. 3 to 4</td>
<td>approx. 8KB</td>
</tr>
</tbody>
</table>

TABLE 2 VoIP software modules

FIGURE 2 VoIP network processing
When a call is made, the call setup module determines which vocoder is to be used. Vocoder functions perform the compression and decompression of the voice samples as necessary. Each vocoder uses a special method to compress the voice. Thus, MIPS requirements vary. For the SH3-DSP, the MIPS required for vocoder functions range from less than one to over 30.

All other MIPS data shown in Table 2 are measured quantities on the SH3-DSP hardware and its cycle accurate simulator. The voice activation detection (VAD) module is included in this table because it requires significant DSP processing. The current implementation, G.729a, is executed every 10 ms after collecting voice samples every ms.

The TCP/IP stack converts the voice frames into IP packet for the transmission over any packet network. In our case, it prepares the IP packets for the Ethernet transmission. It requires about 1 MIPS to 2 MIPS, and Ethernet control task takes less than 1 MIPS. When the VoIP application is running, overall control of the platform functions is maintained by several interrupts and by synchronous tasks using timers. An additional 1 MIPS is allocated for this overall control flow. For an incoming call, the IP packet received by the Ethernet initiates the application, and processing is performed accordingly. The call set up procedure properly starts the voice processing after all steps for timers and interrupt priority are performed.

The relationship between all tasks, input data, and output data is shown in Figure 3. The voice samples come from the codec. These samples are compressed, packetized using TCP/IP, and then delivered to the network interface for transfer over the LAN/WAN. Processing begins when the off-hook condition is detected from the code. Samples are collected every millisecond, compressed at 10 ms intervals, and delivered asynchronously.

The total MIPS needed for a single channel depends heavily on the type of vocoder used and on the tail length of the line echo. Assuming G.729a and a 4 ms tail length, the total MIPS required by one voice channel is about 27 MIPS. When G.711 is used, less processing power is needed, about 10 MIPS. At the worst (assuming a 32 ms tail length and adding some MIPS for coordination), one channel generally requires a maximum of 35 to 38 MIPS.

These estimates can be used to design a system with adequate DSP and CPU processing power to support from one to any number of VoIP channels. The breakdown of software modules should also help you to understand what you’ll need to implement to include VoIP support in your system.

If you already have an RTOS and TCP/IP stack, I think you’ll find that the additional code space requirements are not a major issue. However, the complexity of adding a DSP to a system and managing the flow of data between that processor and the CPU can make for some long hours in the lab.

Yashvant Jani received his PhD in physics from University of Texas-Dallas in 1976. He currently works for Hitachi on residential gateways and applications of SH3-DSP, SH4, and similar devices. Yashvant has also supported embedded controller designs using SH and H8 microcontrollers, and developed embedded system architectures for PDAs, hard disk drives, and voice pagers. You can e-mail him at yashvant.jani@hsa.hitachi.com

References
The International Telecommunication Union (ITU) Web site has information available about all standards mentioned here—www.itu.int. For all relevant information regarding G-series standards, go to www.itu.int/publications/itu-t/itutg.htm.
The following Web sites have detailed information regarding voice over IP: www.telogy.com
www.vocaltec.com
www.voxware.com
www.iptelephony.org
www.micom.com
www.micom.com