VoIP Engine™
Voice over IP Engine Suite

RELATED PRODUCTS:

- VoIP Eng/SIP Reference Kits
- » AnVoice SDK
- » iVoIPEngine SDK
- » LnxVoice SDK
- VoIP Engine
- AnVoice
- LnxVoice
- iVoIPEngine
- LnxVoice demo - Sitara 335x

Distinguish your product! VoIP Engine SW: Sound depth and clarity unprecedented in the mobile voice app market.

Give us a listen!

» Contact us
for more information

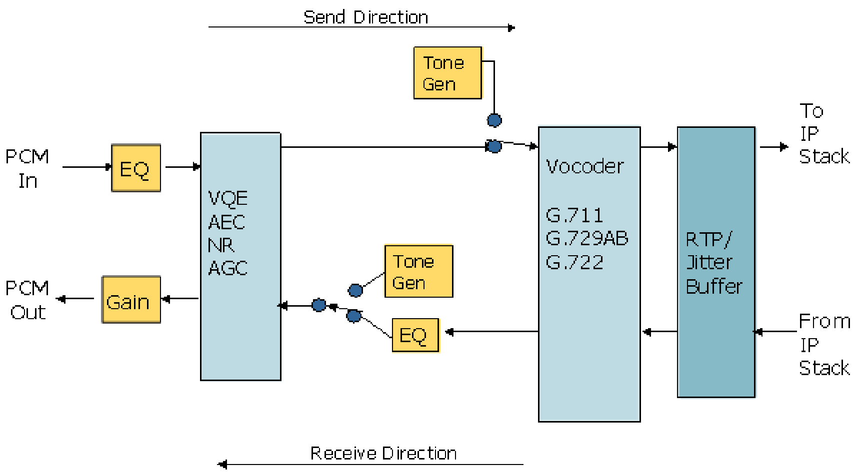
VoIP Engine™: Crystal clear voice and audio for mobile apps.

BY LEVERAGING VOIP ENGINE, DEVELOPERS CAN FOCUS ON THE FUNCTIONALITY OF THE END APPLICATION WITHOUT DEALING WITH THE COMPLEXITIES OF VOICE PROCESSING AT THE NATIVE LAYER.

VoIP Engine portable for use in conjunction with any application or operating system.

VoIP Engine (VE) is at the core of our ARM-based VoIP applications, it provides complete PCM to packet processing. The VoIP Engine software is a software engine package that handles all the voice processing from PCM to Packet and back. Its intended use is in VoIP enabled handsets or desktop phones.

VoIP ENGINE DATA FLOW



VoIP ENGINE IS PORTABLE FOR USE ON MULTIPLE H/W AND OPERATING SYSTEMS

» AnVoice™
» iPVoice™
» LnxVoice™

AVAILABLE AT VARIOUS LEVELS OF INTEGRATION

- VE source Code
- VE object Code
- VE Class Library
- VE SDK includes VE Class library and SIP Class Library
- VE Reference Kit includes Class Libraries and Sample Application code

Customizable to include multiple algorithms

Specifications | Description | Availability

The VoIP engine is purely a data processing engine. It has no interface to drivers or peripherals and performs processing solely at the request of the host application. The host application feeds the VoIP engine PCM samples from the audio input and and RTP packets from the network input. The VoIP engine in turn returns, via callbacks to the host application, PCM samples to be sent to the audio output device and RTP packets to be sent to the network interface.

SOFTWARE FEATURES

Conferencing
VoIP Handler

- G.711 with appendices 1 PLC and 2 (discontinuous transmission)
- G.729A Vocoder
- G.722 (wideband audio) with packet loss concealment
- AMR NB Vocoder
- AMR NB (G.722.2) Vocoder *
- G.726*=
- EVRC-B
- MELP
- RTP/Jitter Buffer
- SRTP
- DTMF tone relay transmit (IETF RFC2833)

PCM Front End (Independently Accessible)

- **HD** Acoustic Echo Cancellation
- Noise Reduction
- Tone Generation
- Gain Control
- Automatic Gain Control
- Diagnostics to assist in acoustic tuning
- Equalization
- SIP*

*** Optional**

Future enhancements will include:

G.729AB (with Appendix B), Plug-in Codecs

FEATURE HIGHLIGHT:

HIGH DEFINITION ACOUSTIC ECHO CANCELLATION

THE HD AEC IS ABLE TO OPERATE IN ENVIRONMENTS WHERE THE BULK DELAY (AUDIO DELAY TO THE SPEAKER AND BACK FROM THE MICROPHONE DUE TO BUFFERING) IS NOT KNOWN. THIS IS NOTABLY THE CASE IN ANDROID-BASED MOBILE PHONES.

- The HD AEC is based upon Adaptive Digital's field-tested AEC technology.
- Automatically learns about the acoustics and delay based upon normal conversation.
- Requires no manual training on a per-model basis or upon handset OS updates.
- Superior Double-Talk Performance
- Supports 8 kHz and 16 kHz sampling rates
- Able to achieve greater than 40 dB of ERLE without nonlinear processor
- Supports tail length up to 256 milliseconds
- HD AEC is a newly integrated addition to Adaptive Digital's VoIP Engine for Mobile devices.

WIDEBAND FEATURES

- Full duplex performance under a wide dynamic range of audio levels.
- Supports wideband audio (16 kHz, 32 kHz, 44.1, and 48 kHz sampling rates) with no artificial cutoff of high frequencies.
- Converges within one second regardless of tail length and sampling rate.

Today's mobile phone applications include an extraordinary amount of functionality. In the Android space in particular, writing software at the native layer is difficult not only due to the complexity of Android but also due to the anarchistic nature of open-source software in general. The best-case scenario for a developer is therefore to work at the Java layer. But for performance reasons, much functionality needs to run at the native layer.

To make mobile phone application development manageable, developers have many development kits at their fingertips to handle the native layer complexity. Adaptive Digital's VoIP Engine brings the necessary VoIP functionality to the native layer. All the developer needs to do is access the VoIP engine using a simple API, and package the supplied VoIP Engine native layer application with the end user Android application.

VoIP Engine is supplied with a sample Java application and a sample native application that in turn interfaces with the VoIP Engine software. The sample Java application interfaces with the sample native application via Java Native Interface (JNI) to setup an RTP/IP to RTP/IP VoIP connection. Android developers can incorporate the Java sample code into more complete VoIP-enabled Android applications.

The VoIP Engine API is clean and simple to use.

Adaptive Digital (**easy integration + field proven algorithms**) = **quick-to-market applications**