Introduction to VisualAudio

Presented by:
Paul Beckmann
Analog Devices CPSG
About this Module

- This module gives an overview of VisualAudio, a tool for rapid development of audio processing software. Examples and demonstrations will be based on the ADSP-BF533 EZ-KIT. You will learn about:
  - The primary features of VisualAudio and how the tool can accelerate product development
  - How to design audio processing layouts using the graphical editor
  - The underlying DSP software architecture

- Target Audience
  - Embedded product developers
  - Some experience with audio
  - Some familiarity with Blackfin processors and the VisualDSP++ development environment

- A separate module – aimed specifically at audio algorithm developers - discusses VisualAudio’s advanced features in more detail
Module Outline

- VisualAudio Overview
- Live Demonstration
- DSP Software Architecture
  - Relationship to VisualDSP++
  - The audio module library
  - Real-time platforms
- Conclusion
VisualAudio Overview
What Is VisualAudio?

- A tool for streamlining audio product development
- Consists of:
  - The VisualAudio Designer - graphical audio processing design application
  - Audio Module Library - commonly used audio functions
  - Example Platforms - real-time frameworks with audio I/O
- Designed for product development engineers
- Provides most of the standard software components found in audio products
- Generates MIPs and memory optimized code
Supports Both Blackfin and SHARC Processor Families from ADI

- **Blackfin**
  - 2x16-bit SIMD fixed-point DSP (VisualAudio uses 32-bit fixed-point for all audio)
  - Rich set of microcontroller features
  - Full external memory interface

- **SHARC**
  - 32-bit floating-point DSP with SIMD capabilities
  - External memory support varies among processor versions

- **Both architectures come in a variety of models with integrated audio peripherals**
  - Serial ports
  - S/PDIF transceiver
  - Hardware sampling rate converters

- **Both processor families are supported by similar platforms, and complementary sets of audio modules and decoders.**
Blackfin vs. SHARC Selection Guide

- The SHARC is ideal for products whose primary function is audio or if there is a significant amount of audio processing
  - Audio/video receivers
  - Professional audio systems
  - High-end automotive audio systems

- The Blackfin is ideal for products that have functions in addition to audio
  - Portable media players
  - Automotive head units & telematics
  - Networked media nodes
  - Mass market pro audio
  - Mid-end automotive amplifiers

- As a rule of thumb, the SHARC is 3 to 4 times as efficient as the Blackfin in processing audio, per MIP
EZ-KIT Evaluation Hardware Supported by VisualAudio

- **ADSP-21262 EZ-KIT**
  - 2-in 8-out analog I/O
  - 1 S/PDIF input
- **ADSP-21364 EZ-KIT**
  - 2-in 8-out analog I/O
  - S/PDIF input and output
- **ADSP-21369 EZ-KIT**
  - 2-in 8-out analog I/O
  - S/PDIF input

- **Audio extender card is coming soon which provides 8-in and 16-out analog I/O**

- **ADSP-BF533 EZ-KIT**
  - 4-in 6-out analog I/O
- **ADSP-BF537 EZ-KIT**
  - 2-in 2-out analog I/O
- **ADSP-BF537 EZ-KIT with audio extender card**
  - 8-in 16-out analog I/O
  - S/PDIF input and output
Key Benefits

◆ For product developers
  ● Provides a starting point and methodology for audio product development
  ● Reduces development time, cost, and risk
  ● Allows engineers to focus on differentiating their products rather than implementing standard features
  ● Provides access to audio IP in a consistent format

◆ For IP developers
  ● Streamlines IP development
  ● Serves as a demonstration platform
  ● Provides a consistent format to deliver audio IP
Live Demo

- Creating a New System
- Designing the Layout
- Generating Code
- Building the Executable
- Real-time Tuning
Demo Overview

◆ Hardware Setup
  ● BF533 EZ-KIT
  ● HPUSB emulator (recommended, but you can use the built-in USB emulator)
  ● Line-level audio source
  ● Powered speakers

◆ Software Setup
  ● VisualAudio
  ● VisualDSP++

◆ Steps
  ● Create an audio processing design using the graphical editor
  ● Generate code
  ● Build and run the executable on the EZ-KIT
  ● Tune the system in real-time
Migrate to Your Target Hardware

- Begin with a reference platform – source code provided
- Write drivers for your target hardware
- Create a platform file that describes your hardware to VisualAudio
- Continue to use VisualAudio on your target hardware
DSP Software Architecture
VisualAudio and VisualDSP++

Platform sources and libs

Audio processing functions

Layout Support Library

“Platform”

Platform XML File

VisualDSP Project File (.dpj)

VisualDSP++

Audio Module XML Files

Layout File (.VAL)

System File (.VAS)

Generated Layout Source Files

Executable
Audio Module Library

- Subroutines for processing PCM audio
- 89/(94) “standard” modules provided in the current Blackfin/(SHARC) release:
  - Mixers
  - Filters
  - Delays
  - Tone controls
  - Basic math
  - Faders / balance
  - Volume controls
  - Compressor
  - Limiter
  - Etc

- Optimized for SIMD execution
- Some have separate versions for mono and stereo inputs
- Source code provided
- Customers can write their own modules as well
Block Processing

- Each audio module operates on a block of data – rather than sample-by-sample
- The number of samples per block is fixed and is called the “TickSize”
- All modules operate at the same TickSize.
- The TickSize is adjustable through the User Interface
- Block processing is a natural fit for audio decoders which output blocks of data (e.g., Dolby Digital outputs 256 sample blocks)
- Block processing yields an efficient implementation
Example Computation
10th order IIR filter

![Graph showing comparison between Blackfin and SHARC for clock cycles/sample vs. tick size (samples).]
Audio Module Interconnections / Wires

- Mono wires contain TickSize audio samples
- Stereo wires hold interleaved data and contain 2*TickSize audio samples
- Control wires contain a single value

- 4 frequency domain wire types
  - Real half spectrum
  - Complex half spectrum
  - Real full spectrum
  - Complex full spectrum
VisualAudio Platforms

- Lightweight interrupt driven real-time frameworks
- Platforms provide
  - Double-buffered DMA-driven audio I/O
  - An interface to VisualAudio generated audio processing
  - A separate non-real-time control thread
  - Tuning
  - Communication with a host micro-controller (if any)
- Several application-specific variants
  - “Basic” - General purpose, PCM I/O
  - AVR (for home theater products with decoders)
  - Automotive
“Basic” Platform

- Targeted at PCM-based audio products without decoders
- Platform divided into
  - A common core framework
  - Platform specific drivers
- Double-buffered DMA-driven block-based audio I/O
- Layout executes at interrupt level
- Tuning, host communication and user control code execute at non-interrupt level

![Diagram showing Multichannel codecs or S/PDIF transceiver connected to SPORT Interface and Audio Processing with User Control Code]
“AVR” Platform

- S/PDIF Interface
- SPORT Interface
- Bitstream Detector
- Audio Decoders
- Audio Processing
- Multi-channel codecs
- SPORT Interface
- User Control Code
Platform / Layout Interface

- Real-time Audio I/O
- VisualAudio Layout Support Library
- Audio Module Render Functions
- Audio Module data structures generated by the VisualAudio Designer

"Platform"

Audio input and output buffers

"Layout"
Conclusion

- VisualAudio accelerates the development of embedded audio applications
- An intuitive graphical user interface allows audio processing to be easily designed and configured
- Supports both the Blackfin and SHARC families of processors and many different EZ-KIT development platforms
- Generates efficient code

- A separate training module covers the VisualAudio environment in more depth
  - Advanced user interface features
  - Writing custom audio modules
  - Interfacing to external design applications
For Additional Information

- A free download is available at the VisualAudio product page

- Additional examples and tutorials can be found at the VisualAudio Developer’s Web site:
  - www.visualaudiodeveloper.com

- Specific technical questions can be sent to:
  - visualaudio.support@analog.com

- Click the “Ask A Question” button