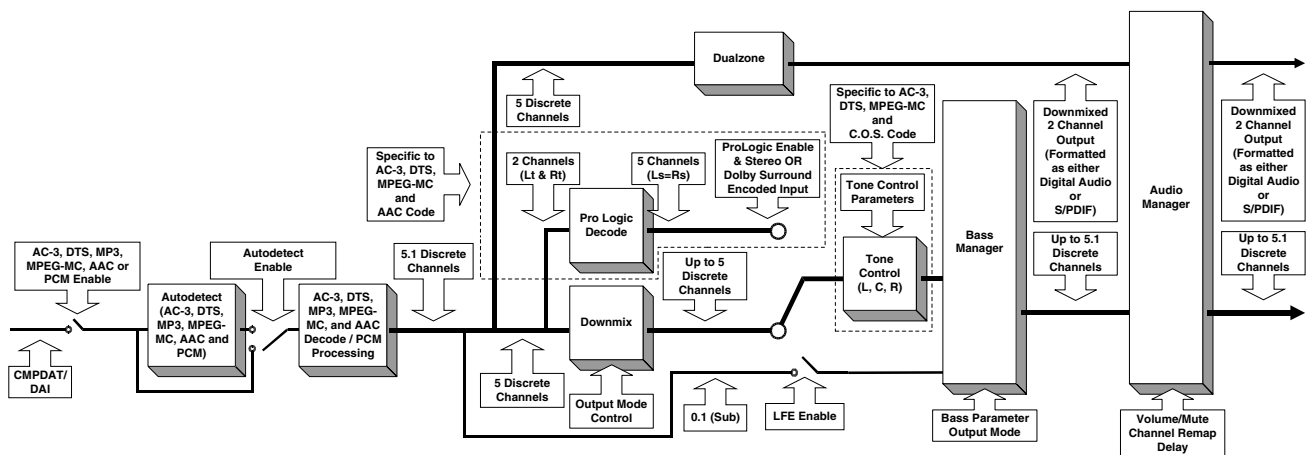# CRYSTAL
from CIRRUS LOGIC

AN163

APPLICATION NOTE

AVR/OUTBOARD DECODER SYSTEMS: APPLICATION CODE USER'S GUIDE FOR THE CS4932X FAMILY

Contents

- Dolby Digital™ (AC-3™) Description
- DTS Digital Surround™ Description
- MPEG-1, Audio Layer 3 (MP3) Description
- MPEG Multichannel Description
- AAC Description
- Crystal Original Surround Description
- Software Naming Convention
- Document Strategy
- Understanding Application Messaging
- Hardware Configurations Supported by the CS4932X and AC-3, DTS, MP3, MPEG Multichannel, AAC and Crystal Original Surround

DOLBY DIGITAL ®

DIGITAL ® dts SURROUND

MPEG)||(Multichannel ®

AAC

- How to Control Common Application Modules such as:
  — Audio Manager
  — Generalized Bass Manager
  — Pro Logic Module
  — Tone Control
- How to Control Specific Application Modules such as:
  — **AC-3**: AC-3 Manager, PCM Processor
  — **DTS**: DTS Manager
  — **MP3**: MP3 Manager
  — **MPEG Multichannel**: MPEG Multichannel Manager
  — **AAC**: AAC Manager
  — **Crystal Original Surround**:
     C.O.S. Effects Manager
     C.O.S. PCM Processor

DIGITAL SOUND

CRYSTAL™
PROCESSING

TABLE OF CONTENTS

Contacting Cirrus Logic Support

For a complete listing of Direct Sales, Distributor, and Sales Representative contacts, visit the Cirrus Logic web site at:
http://www.cirrus.com/corporate/contacts

LIST OF FIGURES

LIST OF TABLES

1. OVERVIEW

The CS4932X is a sub-family of the CS49300 Family DSPs which are system-on-a-chip solutions for Multichannel audio decompression and digital signal processing. Since the part is RAM-based, a download of application software is required each time the CS4932X is powered up.

The CS49300 Family is generally targeted at three different market segments. The broadcast market, where audio/video (A/V) synchronization is always required, the DVD Player market where audio/video (A/V) synchronization is sometimes required, and the outboard decoder markets where audio/video synchronization is not required. Another important differentiation between these applications is the format in which the data will be received by the CS49300 Family. In systems where A/V synchronization is required from the CS4931X, the incoming data is typically PES encoded. In an outboard decoder application, the data typically comes in the IEC61937 format (as specified by the DVD consortium). In DVD players, where A/V synchronization is not required, data is typically delivered in an elementary stream format. An important point to remember is that the CS49300 Family can support all three environments, but different device numbers and downloads are required depending on the input data type.

Broadcast applications include (but are not limited to) digital cable and satellite set-top box applications and digital TVs. Data input can be ES or PES in format, while data delivery is either serial or parallel and is typically bursty in nature.

DVD-Audio Players are unique in that they may have the additional requirement of MLP® (Meridian Lossless Packing) Technology and other compressed audio decoding algorithms that are not needed in the Broadcast market, yet may or may not require A/V synchronization. Data delivery can be ES, PES, or IEC61937 depending on the implementation. Often times, a system may be a hybrid between an outboard decoder and a broadcast system depending on its functionality.

Outboard decoder applications include stand-alone decoders and audio/video receivers. Where data is delivered by way of a S/PDIF cable where the data is formatted IEC61937 (for compressed data) or IEC60958 (for PCM data).

This User's Guide covers application code that provides decoding of compressed data streams such as:

- Dolby Digital (AC-3), delivered via an IEC61937 stream
- DTS and DTS-ES, delivered via an IEC61937 stream; DTS Format-16 and DTS Format-14 (via IEC60958)
- MPEG-1, Layer 3 (MP3) delivered via an IEC61937 stream
- MPEG-2, Audio Layer 2 Stereo and Multichannel (with and without extension), delivered via an IEC61937 stream
- MPEG-2, AAC (Advanced Audio Coding) delivered via an IEC61937 stream.

This guide also covers application code that provides effects processing of Linear PCM delivered in an IEC60958 stream, called Crystal Original Surround.

1.1 Document Strategy

Multiple documents are needed to fully define, understand and implement the functionality of the CS4932X. They can be split up into two basic groups: hardware and application code documentation. Please note that hardware and application code are co-dependent and one can not successfully use the part without an understanding of both. The "ANXXX" notation denotes the application note number under which the respective user's guide was released.

1.1.1 Hardware Documentation

CS49300 Family Data Sheet

This document describes the electrical characteristics of the device from timing to base functionality. This is the hardware designers tool to learn the part's electrical and systems requirements.

1.1.2 CS493XX Application Code User's Guides

The following application notes describe the application codes used with the CS493XX. Whenever an application code user's guide is referred to, it should be assumed that one or more of the below documents are being referenced. This list will grow with each new application released. For the most current list of released user's guides and application codes please refer to the CS4931x/CS49330 Part Matrix vs. Code Matrix and the CS4932x/CS49330 Part Matrix vs. Code Matrix available from the CS49300 Page. All appendices covering proprietary algorithms such as: DTS-ES™, DTS Neo:6, Dolby Digital with Dolby Pro Logic II, Virtual Dolby Digital™, VMAx, SRS CircleSurround™, HDCD®, Logic 7™, SRS TruSurround™, and THX®/THX® Surround EX™, in addition to the corresponding device-specific codes are available from Cirrus Logic. *All application codes are only available for a 90 day evaluation period upon signing of a Crystal Ware Software Evaluation Agreement (CWSEA) after the proper IP rights have been confirmed (i.e. - the end-customer has a Dolby Digital License, DTS License, etc.). A Crystal Ware Licensing Agreement (CWLA) must be signed before these application codes may be used in a production-level design.*

AN161 — *DVD Audio/Video Systems: Application Code User's Guide for the CS4930X*

This guide describes all application code (e.g. Dolby Digital; DTS; MPEG-1, Layer 3 (MP3); MPEG-2, Layer 3 (a.k.a. MP3); MPEG-2, Layer 2 Multichannel.) intended specifically for DVD Players. This guide also describes the additional application codes designed for DVD Audio Players such as: Meridian Lossless Packing (with both elementary and DVD Audio Pack support), all supported LPCM formats, in addition to a discussion of audio/video synchronization considerations. *All application codes referred to in AN161 are only available for a 90 day evaluation period upon signing of a Crystal Ware Software*

Evaluation Agreement (CWSEA) after the proper IP rights have been confirmed (i.e. - the end-customer has a MLP License, Dolby Digital License, DTS License, and/or AAC License). A Crystal Ware Licensing Agreement (CWLA) must be signed before these application codes may be used in a production-level design.

AN162 — *Broadcast Systems: Application Code User's Guide for the CS4931X*

This guide describes all application code (e.g. Dolby Digital; Stereo MPEG-1, Layers 1 and 2, Stereo MPEG-2, Layer 2; and MPEG-2, AAC) designed for broadcast systems such as HDTV and set-top box receivers. This document also provides a discussion of broadcast system considerations and dependencies, such as bursty data delivery, and audio/video synchronization. *All application codes referred to in AN162 are only available for a 90 day evaluation period upon signing of a Crystal Ware Software Evaluation Agreement (CWSEA) after the proper IP rights have been confirmed (i.e. - the end-customer has a Dolby Digital License, and/or AAC License). A Crystal Ware Licensing Agreement (CWLA) must be signed before these application codes may be used in a production-level design.*

AN162A — **Appendix A to Application Note 162:** *PCMP (SMPTE 302M) User's Guide for the CS4931X Family and CS49330*

The PCMP application code is a SMPTE 302M decoder designed for broadcast applications where PCM is delivered in an MPEG-2 transport stream PES packet. It provides support for AV synchronization, channel change, bitstream parameter reporting and PLL control, among other features offered in other broadcast-based application codes discussed in AN162. In addition, a mutually exclusive PCM pass-through application mode is available with the PCMP release. This feature is similar to the AVR-based AC-3 code that is capable of passing PCM via I²S data delivery. *PCMP application code and AN162A are available for a 90 day evaluation period upon signing of a Crystal Ware Software Evaluation Agreement (CWSEA). A Crystal Ware*

Licensing Agreement (CWLA) must be signed before these application codes may be used in a production-level design.

AN162B — **Appendix B to Application Note 162:** *AC-3 + PCM Mixer (AC3N) User's Guide for the CS4931X*

AC3N - Dolby Digital decoder supporting PES parsing and AV Sync for data that is delivered according to the "PES over I²S" Cirrus Logic proprietary packing technique. Additionally, an IEC60958 Lt, Rt output or IEC61937 compressed output is supported from the AUDAT3/XMT958 pin. *AC3N application code and AN162B are available for a 90 day evaluation period upon signing of a Crystal Ware Software Evaluation Agreement (CWSEA). A Crystal Ware Licensing Agreement (CWLA) must be signed before these application codes may be used in a production-level design.*

AN162C — **Appendix C to Application Note 162:** *Interfacing the TeraLogic TL8XX Digital TV ICs to the CS4931X Audio Decoder ICs using AACT or AAST*

AN162C includes a brief description of hardware configurations and software feature set supported by the Cougar DTV Development Platform in combination with the CS4931X when using two separate AAC application codes developed exclusively for TeraLogic. These two codes are AACT and AAST. The AACT code offers up to 5.1 Channel decoded outputs for an 5.1 Channel AAC Digital Audio Source (Low Complexity, ADTS, Elementary Stream Data packed according to the IEC61937 specification) delivered via an I 2 S (or LJ) output from the TL85X. The AACT code is also capable of simultaneously mixing PCM via the DAI port on the CS4931X. The AAST code offers up to 2.0 Channel (Lt, Rt downmix) decoded outputs for a 5.1 Channel AAC Digital Audio Source (Low Complexity, ADTS, Elementary Stream Data packed according to the IEC61937 specification) delivered via an I 2 S (or LJ) output from the TL85x. *AACT and AAST application codes and AN162C are available for a 90 day*

evaluation period upon signing of a Crystal Ware Software Evaluation Agreement (CWSEA). A Crystal Ware Licensing Agreement (CWLA) must be signed before these application codes may be used in a production-level design.

AN163 — *AVR/Outboard Decoder Systems: Application Code User's Guide for the CS4932X Family*

This guide describes all application code (e.g. Dolby Digital; DTS; MPEG-1, Layer 3 (MP3); MPEG-2, Layer 3 (a.k.a. MP3); MPEG-2, Layer 2 Multichannel; MPEG-2, AAC; and Crystal Original Surround) designed for outboard decoder and A/V Receiver systems. *All application codes referred to in AN163 are only available for a 90 day evaluation period upon signing of a Crystal Ware Software Evaluation Agreement (CWSEA) after the proper IP rights have been confirmed (i.e. - the end-customer has a Dolby Digital License, DTS License, and/or AAC License). A Crystal Ware Licensing Agreement (CWLA) must be signed before these application codes may be used in a production-level design.*

AN163A — **Appendix A to Application Note 163:** *Virtual Dolby Digital™ User's Guide for the CS4932X Family*

This appendix offers how to control the software-specific features displayed in the Virtual Dolby Digital Manager. Release of this document and code requires confirmation of an approved Virtual Dolby Digital license. Please contact Dolby Laboratories Licensing Corporation for licensing information. *Virtual Dolby Digital application code and AN163A are available for a 90 day evaluation period upon signing of a Crystal Ware Software Evaluation Agreement (CWSEA) after Cirrus Logic has confirmed that the end-user has a Virtual Dolby Digital License and is in good standing with Dolby Laboratories. A Crystal Ware Licensing Agreement (CWLA) must be signed before this application code may be used in a production-level design.*

AN163B — Appendix B to Application Note 163: *VMAx VirtualTheater*™ *User's Guide for the CS4932X Family*

This appendix offers how to control the software-specific features displayed in the VMAx VirtualTheater Manager. Please contact Harman International and Dolby Laboratories Licensing Corporation for licensing information. *VMAx VirtualTheater application code and AN163B are available for a 90 day evaluation period upon signing of a Crystal Ware Software Evaluation Agreement (CWSEA) after Cirrus Logic has confirmed that the end-user has a VMAx License as well as a Virtual Dolby Digital License and is in good standing with both Harman International and Dolby Laboratories Licensing Corporation. A Crystal Ware Licensing Agreement (CWLA) must be signed before this application code may be used in a production-level design.*

AN163C — Appendix C to Application Note 163: *SRS CircleSurround*™ *User's Guide for the CS4932X Family and CS49330*

This appendix offers how to control the software-specific features displayed in the SRS CircleSurround Manager. Release of this document and code is currently only authorized to licensees of SRS CircleSurround technology. Please contact SRS Labs for licensing information. *SRS CircleSurround application code and AN163C are available for a 90 day evaluation period upon signing of a Crystal Ware Software Evaluation Agreement (CWSEA) after Cirrus Logic has confirmed that the end-user has a SRS CircleSurround License, and is in good standing with SRS Labs. A Crystal Ware Licensing Agreement (CWLA) must be signed before these application codes may be used in a production-level design.*

AN163D — Appendix D to Application Note 163: *Pacific Microsonics*™ *HDCD*® *User's Guide for the CS4932X Family and CS49330*

This appendix offers how to control the software-specific features displayed in the HDCD Manager. Release of this document and code is currently only

authorized to licensees of HDCD technology. Please contact Pacific Microsonics for licensing information. *HDCD application code and AN163D are available for a 90 day evaluation period upon signing of a Crystal Ware Software Evaluation Agreement (CWSEA) after Cirrus Logic has confirmed that the end-user has an HDCD License and is in good standing with Pacific Microsonics. A Crystal Ware Licensing Agreement (CWLA) must be signed before this application code may be used in a production-level design.*

AN163E — Appendix E to Application Note 163: *Logic 5*™ *5.1 Channel PCM Enhancement User's Guide for the CS4932X Family*

This appendix offers how to control the software-specific features displayed in the Logic 5 Manager. Release of this document and code is currently only authorized to licensees of Logic 5 technology. Please contact Lexicon, Inc. for licensing information. *Logic 5 application code and AN163E are available for a 90 day evaluation period upon signing of a Crystal Ware Software Evaluation Agreement (CWSEA) after Cirrus Logic has confirmed that the end-user has a Logic 5 License and is in good standing with Lexicon, Inc. A Crystal Ware Licensing Agreement (CWLA) must be signed before this application code may be used in a production-level design.*

AN163F — Appendix F to Application Note 163: *SRS TruSurround*™ *User's Guide for the CS4932X Family*

This appendix offers how to control the software-specific features displayed in the SRS TruSurround Manager. Release of this document and code requires confirmation of an approved SRS TruSurround license. Please contact SRS Labs for licensing information. Furthermore, if the version of code that is to be released is based on Virtual Dolby Digital technology, a Virtual Dolby Digital License will be required. Please contact Dolby Laboratories Licensing Corporation for licensing information. If the version of this code is based on DTS technology, a DTS license will be required. Please contact Digital Theater Systems for licensing information. If the version of this code is

based on MPEG-2, Layer 2 Multichannel technology, no additional licensing is required other than the SRS TruSurround license. *All SRS TruSurround application codes and AN163F are available for a 90 day evaluation period upon signing of a Crystal Ware Software Evaluation Agreement (CWSEA) after Cirrus Logic has confirmed that the end-user has a SRS TruSurround License, Virtual Dolby Digital License and DTS Virtual 5.1 License and is in good standing with SRS Labs, Dolby Laboratories Licensing Corporation, and DTS Corporation. A Crystal Ware Licensing Agreement (CWLA) must be signed before these application codes may be used in a production-level design.*

AN163G — Appendix G to Application Note 163: *Crystal 5.1 Channel Digital Post-Processor User's Guide for the CS49330*

This appendix offers how to control the software-specific features for a non-license based multichannel PCM digital post-processing code. This code offers the following features: Multichannel PCM input, LR2LsRs copy, Test Noise and Impulse Generator, PCM Mixing, Downmixing, Tone Control and 3-band Parametric EQ control of up to 5 full-bandwidth input channels, Bass Manager and Delay. *Crystal 5.1 Channel Digital Post-Processor application code and AN163G are available for a 90 day evaluation period upon signing of a Crystal Ware Software Evaluation Agreement (CWSEA). A Crystal Ware Licensing Agreement (CWLA) must be signed before this application code may be used in a production-level design.*

AN163H — Appendix H to Application Note 163: *THX®/THX® Surround EX™ 5.1 Channel Post-Processor User's Guide for the CS49330*

This appendix offers how to control the software-specific features for THX and THX Surround EX application codes. Release of this document and code(s) requires confirmation of an approved THX license status. Please contact the Mr. Mark Paddack at the THX Division of Lucasfilm, Ltd. for licensing information. *Both Home THX Cinema and/or THX Surround EX 5.1 Channel Post-Processing application codes and AN163H are available for a 90 day evaluation period upon signing of a Crystal Ware Software Evaluation Agreement (CWSEA) after Cirrus Logic has confirmed that the end-user has a Home THX Cinema and/or THX Surround EX License and is in good standing with the THX Division of Lucasfilm, Ltd. A Crystal Ware Licensing Agreement (CWLA) must be signed before these application codes may be used in a production-level design.*

AN163I — Appendix I to Application Note 163: *Crystal P.D.F. User's Guide for the CS4932X*

This appendix offers how to control the software-specific features for Crystal P.D.F. (Pro Logic Double-Frequency) application code. This code allows for stereo or Pro Logic encoded PCM Stereo streams with sampling rates up to 96kHz. This code also allows for 1:2 upsampling for all input sampling frequencies up to 48kHz. Digital deemphasis for all input sampling frequencies is also available with dual-precision bass management. A Dolby Pro Logic License is required. *Crystal P.D.F. application code and AN163I are available for a 90 day evaluation period upon signing of a Crystal Ware Software Evaluation Agreement (CWSEA) after Cirrus Logic has confirmed that the end-user has a Dolby Pro Logic License and is in good standing with Dolby Laboratories Licensing Corporation. A Crystal Ware Licensing Agreement (CWLA) must be signed before this application code may be used in a production-level design.*

AN163J — Appendix J to Application Note 163: *Digital/Multimedia Speaker User's Guide for the CS4932X Family*

This appendix describes the Digital/Multimedia Speaker (D.M.S.) licensed software that can process up to two PCM stereo streams, as well as perform an AC-3 decode or DTS (depending on the version of code requested). It also controls independent 5-channel tone control and 3-band parametric EQ on all full-bandwidth channels. With the appropriate Dolby licenses, codes for

various virtualization modules are available that support Virtualization of Dolby Digital or Dolby Pro Logic decoded output down to two channels. A Dual-Precision Bass Manager is also available with this code. A Dolby Digital, DTS or additional virtualizer technology license may be required, depending on which application code is required. *All Digital/Multimedia application codes and AN163J are available for a 90 day evaluation period upon signing of a Crystal Ware Software Evaluation Agreement (CWSEA) after Cirrus Logic has confirmed that the end-user has the proper IP owner's licenses and is in good standing with all respective companies. A Crystal Ware Licensing Agreement (CWLA) must be signed before these application codes may be used in a production-level design.*

AN163K — Appendix K to Application Note 163: *5-Channel, 5-Band Graphic Equalizer User's Guide For the CS4932X Family*

This appendix describes how to control the licensed application code that runs on the CS4932X Family DSP and offers the designer a 5-channel, 5- band graphic equalizer. This graphic equalizer features 24-bits of precision at the backend of an AC-3 decoder (which offers additional Pass-Through). The application code also offers a DTS decoder or an MPEG Multichannel decoder, instead of a Dolby Digital decoder. *All application codes with a 5-Band, 5 Channel Graphic EQ and AN163K are available for a 90 day evaluation period upon signing of a Crystal Ware Software Evaluation Agreement (CWSEA) after Cirrus Logic has confirmed that the end-user has a Dolby Digital License and DTS Digital Surround License and is in good standing with Dolby Laboratories Licensing Corporation, and the DTS Corporation. A Crystal Ware Licensing Agreement (CWLA) must be signed before these application codes may be used in a production-level design.*

AN163L — Appendix L to Application Note 163: *DTS-ES Discrete 6.1 and Matrix 6.1 User's Guide for the CS4932X Family*

This appendix describes the licensed software designed to run on: the CS49326 and CS49329 which provides DTS Digital Surround 5.1 channel decoding in addition to DTS-ES Discrete 6.1 and DTS-ES Matrix 6.1 decoding. Descriptions for software configuration and in-depth descriptions of application control modules are also included in this document. *DTS-ES Discrete 6.1 and Matrix 6.1 application code and AN163L are available for a 90 day evaluation period upon signing of a Crystal Ware Software Evaluation Agreement (CWSEA) after Cirrus Logic has confirmed that the end-user has a DTS-ES License, and is in good standing with the DTS Corporation. A Crystal Ware Licensing Agreement (CWLA) must be signed before these application codes may be used in a production-level design.*

AN163M — Appendix M to Application Note 163: *Dolby Pro Logic II* ™ *PCM Enhancement User's Guide for the CS4932X Family*

This appendix describes how to control the licensed software designed to run on: the CS4932X Family, which provides Dolby Pro Logic II decoding of 2 channel PCM material and AC-3 DTS, MPEG. To find out about the licensing requirements for using these Dolby features, contact Dolby Laboratories Licensing Corporation for licensing information. *All application codes that contain Dolby Pro Logic II decoding technology and AN163M are available for a 90 day evaluation period upon signing of a Crystal Ware Software Evaluation Agreement (CWSEA) after Cirrus Logic has confirmed that the end-user has a Dolby Pro Logic II License in addition to a Dolby Digital License and DTS Digital Surround License and is in good standing with Dolby Laboratories Licensing Corporation, and the DTS Corporation. A Crystal Ware Licensing Agreement (CWLA) must be signed before these application codes may be used in a production-level design.*

AN163N — Appendix N to Application Note 163: *DTS Neo:6 PCM Enhancement User's Guide for the CS4932X Family*

lo

This appendix describes the licensed software designed to run on: the CS49326 and CS49329 which provides DTS NEO:6 decoding for 2-channel PCM. This appendix documents how to enable the DTS NEO:6 functionality. *DTS Neo:6 and AN163N are available for a 90 day evaluation period upon signing of a Crystal Ware Software Evaluation Agreement (CWSEA) after Cirrus Logic has confirmed that the end-user has a DTS Neo:6 License and is in good standing with the DTS Corporation. A Crystal Ware Licensing Agreement (CWLA) must be signed before these application codes may be used in a production-level design.*

AN163O — Appendix H to Application Note 163: THX®/THX® Surround EX™ 7.1 Channel Post-Processor User's Guide for the CS49330

This appendix offers how to control the software-specific features for Home THX Cinema and THX Surround EX 7.1 Channel Post-Processor application codes. This application code is capable of post-processing DTS-ES Discrete 6.1 channel streams. Release of this document and code(s) requires confirmation of an approved THX license status. Please contact Mr. Mark Paddack at the THX Division of Lucasfilm, Ltd. for licensing information. *Both Home THX Cinema and/or THX Surround EX 7.1 Channel Post-Processing application codes and AN163O are available for a 90 day evaluation period upon signing of a Crystal Ware Software Evaluation Agreement (CWSEA) after Cirrus Logic has confirmed that the end-user has a Home THX Cinema and/or THX Surround EX License and is in good standing with the THX Division of Lucasfilm, Ltd. A Crystal Ware Licensing Agreement (CWLA) must be signed before these application codes may be used in a production-level design.*

AN163P — Appendix P to Application Note 163: Crystal PL2_2FS User's Guide for the CS4932X

This appendix offers how to control the software-specific features for Crystal PL2_2FS (Pro Logic II Double-Frequency) application code. This code allows for stereo or Pro Logic encoded PCM Stereo streams with sampling rates up to 96kHz. This code also allows for 1:2 upsampling for all input sampling frequencies up to 48kHz. Digital deemphasis for all input sampling frequencies is also available with dual-precision bass management. A Dolby Pro Logic II License is required. *Crystal PL2_2FS application code and AN163P are available for a 90 day evaluation period upon signing of a Crystal Ware Software Evaluation Agreement (CWSEA) after Cirrus Logic has confirmed that the end-user has a Dolby Pro Logic II License and is in good standing with Dolby Laboratories Licensing Corporation. A Crystal Ware Licensing Agreement (CWLA) must be signed before this application code may be used in a production-level design.*

AN163Q — Appendix Q to Application Note 163: Crystal 7.1 Channel Digital Post-Processor User's Guide for the CS49330

This appendix offers how to control the software-specific features for a non-license based multichannel PCM digital post-processing code. This code offers the following features: Multichannel PCM input, LR2LsRs copy, Test Noise and Impulse Generator, PCM Mixing, Downmixing, Tone Control and 3-band Parametric EQ control of up to 7 full-bandwidth input channels, Bass Manager and Delay. *Crystal 7.1 Channel Digital Post-Processor application code and AN163Q are available for a 90 day evaluation period upon signing of a Crystal Ware Software Evaluation Agreement (CWSEA). A Crystal Ware Licensing Agreement (CWLA) must be signed before this application code may be used in a production-level design.*

AN163R — Appendix R to Application Note 163: Crystal Extra Surround 6.1/7.1 Channel Decoder Enhancement User's Guide for the CS4932X

This appendix offers how to control all application codes which offer the Crystal Extra Surround 6.1/7.1 Channel Decoder Enhancement. Crystal Extra Surround (CES) algorithm processes the

surround channels (Ls and Rs) to produce the extra surround channel (BsC) or channels (BsL, BsR) using proprietary matrix techniques developed by Cirrus Logic. The Crystal Extra Surround module can operate in "Matrix Encoded Mode" (e.g. - Surround EX™ encoded Dolby Digital™ streams or DTS-ES™ Extended Surround streams) or "Non-Matrix Encoded Mode" (e.g. - normal Dolby Digital, DTS Digital Surround™, or MPEG Multichannel streams) which is user controlled. These two different matrix processing modes are available to help provide optimum surround channel steering performance depending on whether the input surround channels (Ls and Rs) are matrix encoded or not. *All Crystal Extra Surround enabled application codes and AN163R are available for a 90 day evaluation period upon signing of a Crystal Ware Software Evaluation Agreement (CWSEA). A Crystal Ware Licensing Agreement (CWLA) must be signed before these application codes may be used in a production-level design.*

AN163S — Appendix S to Application Note 163: *Logic 7™ 7.1 Channel PCM Enhancement User's Guide for the CS4932X Family*

This appendix offers how to control the software-specific features displayed in the Logic 7 Manager. Release of this document and code is currently only authorized to licensees of Logic 7 technology. Please contact Lexicon, Inc. for licensing information. *Logic 7 application code and AN163S are available for a 90 day evaluation period upon signing of a Crystal Ware Software Evaluation Agreement (CWSEA) after Cirrus Logic has confirmed that the end-user has a Logic 7 License and is in good standing with Lexicon, Inc. A Crystal Ware Licensing Agreement (CWLA) must be signed before this application code may be used in a production-level design.*

AN163T — Appendix T to Application Note 163: *SRS CircleSurround II™ User's Guide for the CS4932X Family and CS49330*

This appendix offers how to control the software-specific features displayed in the SRS CircleSurround II Manager. Release of this document and code is currently only authorized to licensees of SRS CircleSurround II technology. Please contact SRS Labs for licensing information. *SRS CircleSurround II application code and AN163T are available for a 90 day evaluation period upon signing of a Crystal Ware Software Evaluation Agreement (CWSEA) after Cirrus Logic has confirmed that the end-user has a SRS CircleSurround II License, and is in good standing with SRS Labs. A Crystal Ware Licensing Agreement (CWLA) must be signed before this application code may be used in a production-level design.*

AN163U — Appendix U to Application Note 163: *Crystal DVD Audio Bass Management User's Guide for the CS4932X Family and CS49330*

This appendix offers how to control the software-specific features of the Crystal DVD Audio Bass Manager application code in addition to to showing a system block diagram of how to deliver 6 channels of 96kHz, 24-bit audio or 2 channels of 192kHz, 24-bit audio into the DSP from 3 stereo, high-performance A/Ds. *Crystal DVD Audio Bass Management application code and AN163U are available for a 90 day evaluation period upon signing of a Crystal Ware Software Evaluation Agreement (CWSEA). A Crystal Ware Licensing Agreement (CWLA) must be signed before this application code may be used in a production-level design.*

1.2 Using the CS4932X

No matter what application is being used on the chip, the following four steps are always followed to use the CS4932X in system. Please refer to the associated documents (seen below in parentheses).

1) Reset and/or Download Code. (CS49300 Family Datasheet)

2) Hardware Configuration. (CS49300 Family Datasheet)

3) Application Configuration. (AN163)

4) Kickstart - This is the "Go" command to the CS4932X once the system is properly configured. (AN163)

1.3 Software Naming Convention

To keep up with the different applications, parts, hardware revisions and software revisions the following naming convention has been adopted for the CS4932X software:

ANAME_CCCCCC_VV.LD

where:

ANAME = Application Name = 2...12 chars (can include "_"s)

CCCCCC = Chip Number = 6 chars
NOTE: This is exactly what is marked on the chip except "x" wherever wildcard applies

VV = Version Number = 2 digits

This document uses the terms Dolby Digital and AC-3 interchangeably. For the AC-3 Section, CS49325, 326, and 329 has been replaced with CS4932X. For the DTS Section, CS49326 and 329 have been replaced with CS4932X. For the MPEG Multichannel Section, CS49325, 326, 329 and 330 have been replaced with CS493XX. For the MP3 Section, CS49325, 326, 329, and 330 have been replaced with CS493XX. For the AAC Section, CS49329 is the only device applicable. For the C.O.S. Section, CS49325, 326, 329, and 330 have been replaced with CS493XX. Unless otherwise specified, CS4932X should be interpreted as applying to the CS49325, 326, 328 and 329. Please refer to Figure 1 and Figure 2 for more detailed information.

It should be noted that some AC-3 codes (which also handle PCM Pass-Through), DTS codes and C.O.S. codes have an "I" after the first three characters, such as ac3i_493263_05.LD, dtsi_493263_05.LD, and effi_4932xx_04.ld. This "I" stands for IBA or Internal Boot Assist. This is due to the fact that the code for Dolby Digital, and/or DTS (along with DTS Tables), and Crystal Original Surround Effects code are already located in the ROM of the CS49325 (AC-3 and COS only),

CS49326, and CS49328 (DTS and COS only). The IBA code must be downloaded just like any other full download code following the instructions in the CS49300 Family Datasheet to initiate the respective application. Please see Section 1.4 "Understanding the Differences between IBA Application Codes and Full Download Application Codes" on page 15.

1.4 Understanding the Differences between IBA Application Codes and Full Download Application Codes

The Internal Boot Assist (IBA) code simply loads the internal code from ROM into active memory, whereas the Full Download downloads directly to active memory. The typical size of an IBA code is around 300 bytes, where a Full Download code can be as large as 32 Kbytes. IBA codes are only for applications that accept IEC60958 or IEC61937 streams. The system designer should be aware that while both codes may be downloaded from the host microcontroller or autobooted from external memory, there are some fundamental feature set differences between the two types of application codes for AC-3, DTS and C.O.S.

Please see below for all major feature differences between AC3_4932XX_VV.LD vs. AC3I_4932XX_VV.LD:

AC3_4932XX_VV.LD has:

1) Fs-dependent digital deemphasis for PCM input. This means that the deemphasis coefficients are embedded inside the code (for 48 kHz, 44.1 kHz and 32 kHz), so that the user need only select the appropriate Fs, and when digital deemphasis is enabled, the response will be correct. For PCM streams with pre-emphasis and an Fs higher than 48 kHz, additional steps must be taken to enable deemphasis. Once the host determines that the incoming PCM stream has changed to 96 kHz, 99.2 kHz, or 64 kHz from the S/PDIF receiver and pre-emphasis was applied earlier on in the mastering process, the host must either reset the DSP (due to the Fs

change on the DIR) and re-download the ac3_4932xx_vv.ld code with appropriate Fs digital deemphasis coefficient set OR perform an Application Restart, and then download the appropriate Fs digital deemphasis coefficient set.

Note: This extra message patch download is only necessary if the host wants to enable digital deemphasis.

2) Tone Control.

3) Audio_Change_Config_Notification for the Dolby Digital stream-specific variable of timecod2 (also know as dsurmodex). This means that if a stream changes from just a Dolby Digital stream to one that is Dolby Digital Surround EX encoded, the DSP will perform an unsolicited message.

AC3I_4932XX_VV.LD has:

1) There exists a message patch for 96 kHz, 88.2 kHz, 64 kHz, 48 kHz, 44.1 kHz, and 32 kHz Fs digital deemphasis coefficients. This means that once the host determines that the incoming PCM stream has changed to 96 kHz, 99.2 kHz, 64 kHz, 48 kHz, 44.1 kHz, or 32 kHz from the S/PDIF receiver and pre-emphasis was applied earlier on in the mastering process, the host must either: (a) Reset the DSP (due to the Fs change on the DIR), and re-download the AC3I32XX.LD code with appropriate Fs digital deemphasis coefficient set. (b) Or, perform an Application Restart, and then download the appropriate Fs digital deemphasis coefficient set.

Note: This extra message patch download is only necessary if the host wants to enable digital deemphasis.

Please see below for all major feature differences between DTS_32XX.LD vs. DTSI32XX.LD:

DTS_32XX.LD has:

1) Tone Control.

2) Pro Logic decoding support for AMODE = 2, 3,

or 4 type DTS streams.

3) Audio_Change_Config_Notification for the DTS stream-specific variable of pcmr. This means that if a stream changes from just a DTS 5.1 stream to one that is DTS-ES Discrete 6.1 or DTS-ES Matrix 6.1 encoded, the DSP will perform an unsolicited message.

DTSI32XX.LD does not incorporate the features of DTS_32XX.LD.

Please see below for all major feature differences between EFF_32XX.LD vs. EFFI32XX.LD:

EFF_32XX.LD has:

1) Tone Control.

2) Fs-dependent digital deemphasis for PCM input. This means that the deemphasis coefficients are embedded inside the code (for 48 kHz, 44.1 kHz, and 32 kHz), so that the user need only select the appropriate Fs, and when digital deemphasis is enabled, the response will be correct. For PCM streams with pre-emphasis and an Fs higher than 48 kHz, additional steps must be taken to enable deemphasis. Once the host determines that the incoming PCM stream has changed to 96 kHz, 99.2 kHz, or 64 kHz from the S/PDIF receiver and pre-emphasis was applied earlier on in the mastering process, the host must either: (a) Reset the DSP (due to the Fs change on the DIR), and re-download the EFF_32XX.LD code with appropriate Fs digital deemphasis coefficient set. (b) Or, perform an Application Restart, and then download the appropriate Fs digital deemphasis coefficient set.

Note: This extra message patch download is only necessary if the host wants to enable digital deemphasis.

EFFI32XX.LD has:

1) There exists a message patch for 96 kHz, 88.2 kHz, 64 kHz, 48 kHz, 44.1 kHz, and 32 kHz Fs digital deemphasis coefficients. This means that once the host determines that the

incoming PCM stream has changed to 96 kHz, 99.2 kHz, 64 kHz, 48 kHz, 44.1 kHz or 32 kHz from the S/PDIF receiver and pre-emphasis was applied earlier on in the mastering process, the host must either: (a) Reset the DSP (due to the Fs change on the DIR), and re-download the EFFI32XX.LD code with appropriate Fs digital deemphasis coefficient set. (b) Or, perform an Application Restart, and then download the appropriate Fs digital deemphasis coefficient set.

Note: This extra message patch download is only necessary if the host wants to enable digital deemphasis.

1.5 Understanding the Feature Set of a Device vs. the Application Code

Please refer to CS4932X/CS49330 Part Matrix vs. Code Matrix PDF available from the CS49300 Page.

1.6 Dolby Digital Description

Dolby Digital, also known as AC-3, is a digital audio compression algorithm developed by Dolby Laboratories for up to 5.1 channels of audio. This User's Guide covers the software that decodes Dolby Digital, and is designed to run on the CS49325, CS49326, CS49328, and CS49329 devices. This User's Guide covers all code that is named with AAAA == AC3_, AC3D, AC3X, AC3R and CCC == 325, 326 and 329 in addition to AAAA == AC3I and CCC == 325 and 326, as described in the previous section.

Along with Dolby Digital, this document describes how to use the code for Dolby Pro Logic decode, tone control, pink/white noise generation, and bass management, as well as other features. It is assumed that the reader is familiar with the requirements and features of Dolby Digital as specified in:

• **Dolby AC-3 Specification:** *Digital Audio Compression Standard (AC-3)*, Advanced Television Systems Committee, Dec. 20, 1995

• **Licensee Information Manual:** *Dolby Digital Multichannel Digital Audio Decoding for Consumer Products, Licensee Information Manual*, Version 2.0, April, 1997

Special versions of the Dolby Digital decoder exist with different features, such as Virtual Dolby Digital™, VMAx VirtualTheater™, and SRS TruSurround™. Dolby Digital and/or Pro Logic decoding must be enabled in order for proper operation. For each such version of Dolby Digital decode, a separate appendix covers the incremental changes with respect to this application note. For commands specific to Virtual Dolby Digital software, please refer to AN163A. For commands specific to VMAx VirtualTheater software, please refer to AN163B. For commands specific to SRS TruSurround software, please refer to AN163F.

Figure 1 displays a functional block diagram of the Dolby Digital application code when configured for PCM pass-through or Pink/White Noise Generation. It gives an idea of the interaction between the various application modules when PCM is enabled. Figure 2 displays a functional block diagram when the part is configured for decoding AC-3.

AC-3 and PCM should be considered as separate applications when using this code.
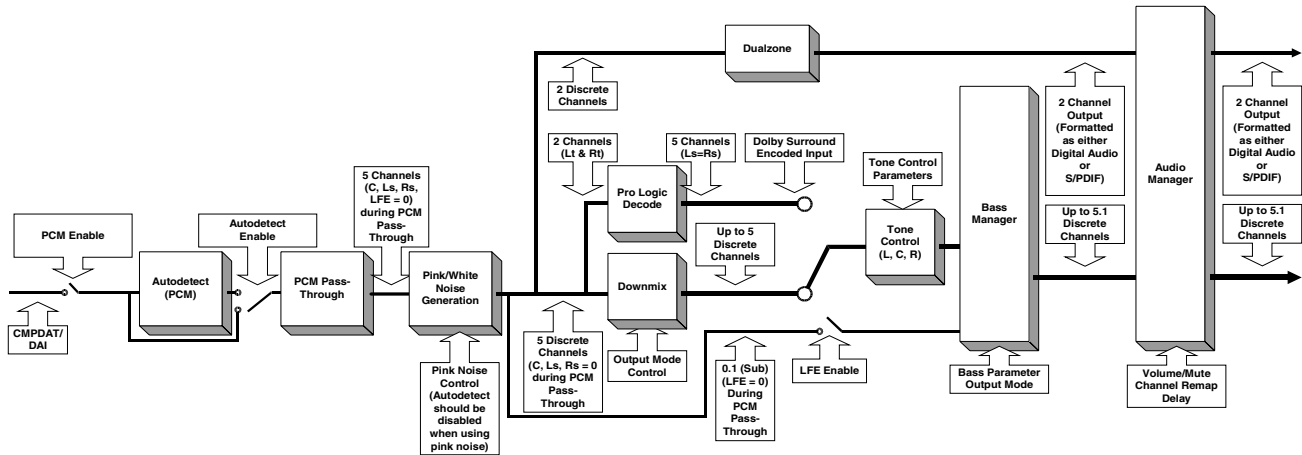
Figure 1. Dolby Digital Block Diagram, Configured for PCM Pass-Through/Pink Noise Generation
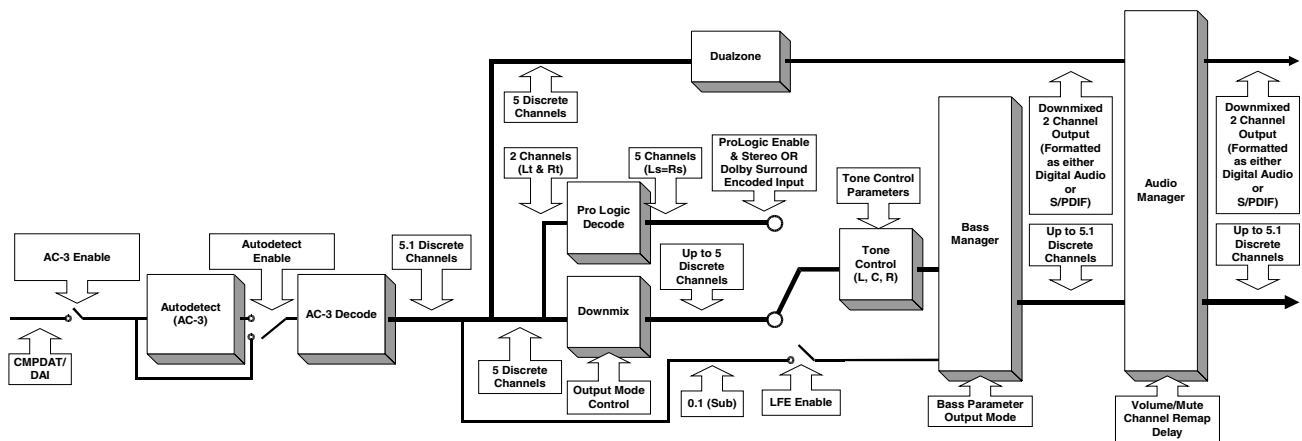
Figure 2. Dolby Digital Block Diagram, Configured for AC-3 Decoding

1.7 DTS Digital Surround Description

DTS Digital Surround is a digital audio compression algorithm developed by Digital Theater Systems for up to 5.1 channels of audio. This User's Guide covers the software that decodes DTS and is designed to run on the CS49326, CS49328, and CS49329 devices. This user's guide covers all code that is named with AAAA == DTSI and DTS_, and CCC==326, 328 and 329, as described in Section 1.3 "Software Naming Convention" on page 15.

Along with DTS, this document describes how to use the code for Dolby Pro Logic decode, tone control and bass management, as well as other features. It is assumed that the reader is familiar with the requirements and features of DTS as specified in:

• **DTS Specification:** *DTS Coherent Acoustics Decoder*, DTS Technology, Version 1.0, January 20, 1998

• **Licensee Manual:** *DTS Master Quality Multichannel Digital Audio Decoding System for Consumer Products Licensee Manual*, Version 1.0, July 1998

A special version of the DTS decoder exists with a virtualization feature called SRS TruSurround™. DTS and/or Pro Logic decoding must be enabled in order for proper operation. For commands specific to SRS TruSurround software, please refer to AN163F.

Figure 3 displays a functional block diagram of the DTS application code and gives an idea of the interaction between the various application modules.
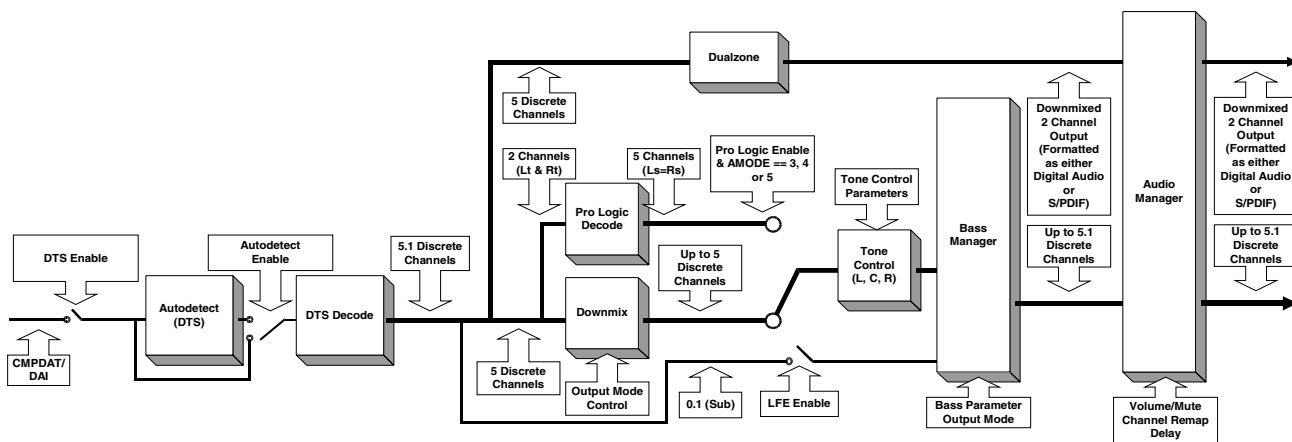
Figure 3. DTS Block Diagram

1.8 MP3 Description

MPEG-1, Audio Layer 3 (also known as MP3) is a digital audio compression algorithm developed by the Moving Pictures Experts Group for up to 2 discrete decoded channels of audio. This User's Guide covers the software that decodes MPEG-1, Audio Layer 3 and is designed to run on the CS49325, CS49326, CS49328, and CS49329 devices. This User's Guide covers all code that is named with AAAA == MP3_ and CCC==325, 326, 328 and 329, as described in the previous Section 1.3 "Software Naming Convention" on page 15.

Along with MP3, this document describes how to use the code for bass management, as well as other features. It is assumed that the reader is familiar with the features of MP3 as specified in:

> ISO/IEC 11172-3, *Information technology - Coding of moving pictures and associated audio for digital storage media at up to about 1.5 Mbit/s - Part 3: Audio*

Figure 4 displays a functional block diagram of the MP3 application code and gives you an idea of the interaction between the various application modules.
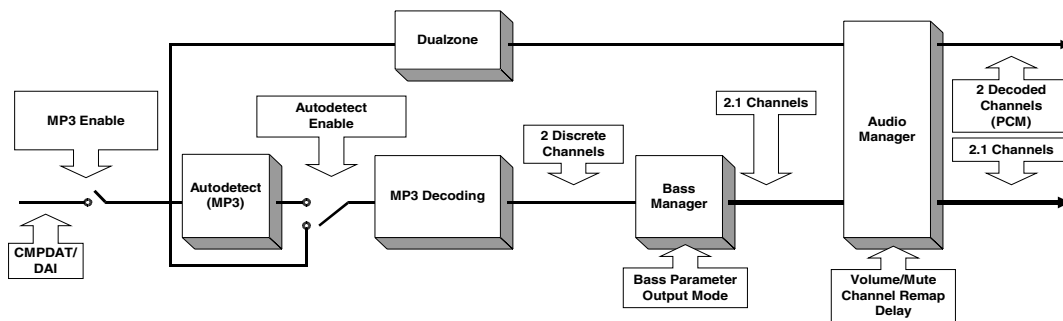
Figure 4. MP3 Block Diagram

1.9 MPEG Multichannel Description

MPEG Multichannel is a digital audio compression algorithm developed by the Moving Pictures Experts Group for multiple discrete encoded channels of audio. This User's Guide covers the software that decodes MPEG-1, Audio Layer 2 as well as MPEG-2, Audio Layer 2 (with and without extension) Stereo and Multichannel streams with a decoded output of up to 5.1 discrete channels and is designed to run on the CS49325, CS49326, CS49328, and CS49329. This User's Guide covers all code that is named with AAAA == MPG_ and CCC==325, 326, 328 and 329, as described in Section 1.3.

Along with MPEG Multichannel this document describes how to use the code for Dolby Pro Logic

decode, tone control and bass management, as well as other features. It is assumed that the reader is familiar with the features of MPEG Multichannel as specified in:

ISO/IEC 13818-3, *MPEG Audio Coding Standard*

A special version of the MPEG Multichannel decoder exists with a virtualization feature, called SRS TruSurround™. MPEG Multichannel and/or Pro Logic decoding must be enabled in order for proper operation. For commands specific to SRS TruSurround software, please refer to AN163F.

Figure 5 displays a functional block diagram of the MPEG Multichannel application code and gives an idea of the interaction between the various application modules.
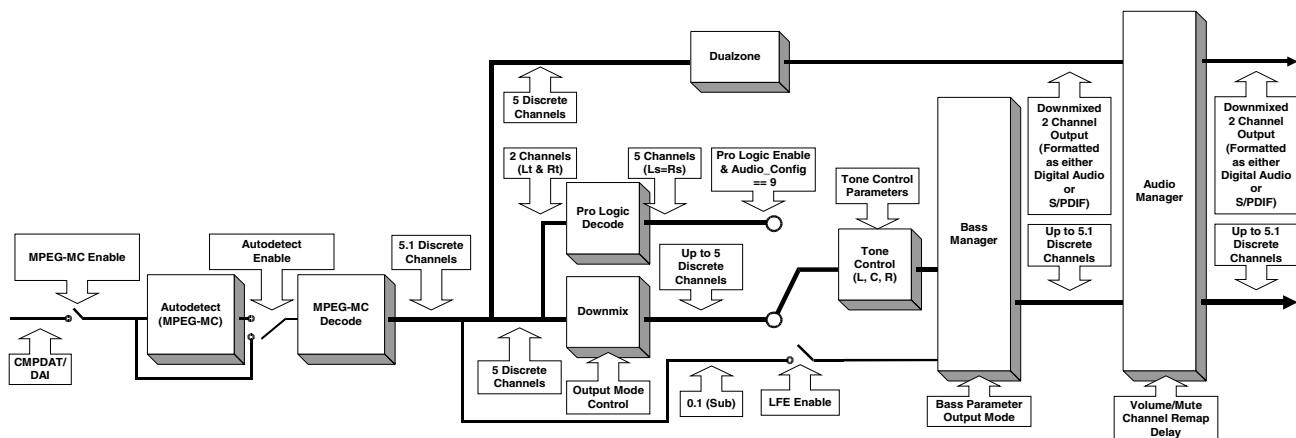
Figure 5. MPEG Multichannel Block Diagram

1.10 AAC Description

AAC (Advanced Audio Coding) is an "MPEG-2-standard-based" digital audio compression algorithm (offering up 5.1 discrete decoded channels for this implementation) collaboratively developed by AT&T, the Fraunhofer Institute, Dolby Laboratories, and the Sony Corporation. This User's Guide covers the software that decodes and is designed to run on the CS49329. This User's Guide covers all code that is named with AAAA == AAC_ and CCC==329 as described in the previous section.

Along with AAC this document describes how to use the code for Dolby Pro Logic decode and bass management, as well as other features. It is assumed that the reader is familiar with the features of AAC as specified in:

IS 13818-7, *MPEG-2 Advanced Audio Coding (AAC) Specification*, April, 1997

Figure 6 displays a functional block diagram of the AAC application code and gives you an idea of the interaction between the various application modules.
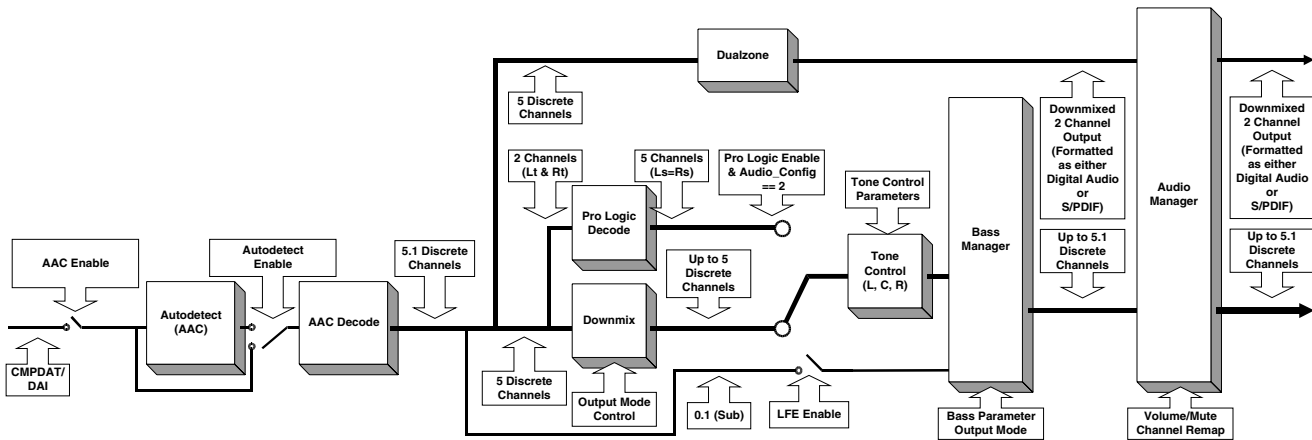
Figure 6. AAC Block Diagram

1.11 Crystal Original Surround Description

Crystal Original Surround is an application code designed to take process PCM data and convert a two-channel source into 5.1 channels with effects processing. This User's Guide covers the software that processes PCM using the C.O.S. Application Code designed to run on the CS49325, CS49326, CS49328, CS49329, and CS49330. This User's Guide covers all code that is named with AAAA == EFF_, and CCC==325, 326, 328, 329, and 330, as described in the previous section.

It is assumed that the reader is familiar with effects processing, however a brief description of reverb, echo, and other filtering techniques are covered.

Along with Crystal Original Surround, this document describes how to use the code for tone control and bass management, as well as other features. Figure 7 displays a functional block diagram of the Crystal Original Surround application code and gives you an idea of the interaction between the various application modules.

Other Stereo PCM processing application codes are available and are discussed in separate appendices to AN163. They are: SRS Circle Surround (AN163C), Pacific Microsonics HDCD (AN163D) and Logic 7 (AN163E).
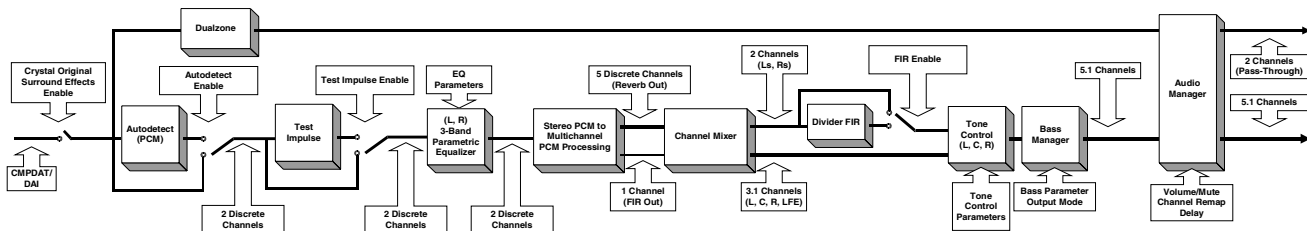
Figure 7. Crystal Original Surround Block Diagram

2. HARDWARE CONFIGURATION

After download or soft reset, and before kickstarting the application (please see Section 4.1 "Audio Manager for Dolby Digital, DTS, MP3, MPEG Multichannel, AAC and Crystal Original Surround" on page 33 for more information on kickstarting), the host has the option of changing the default hardware configuration. Address checking for serial communication and data type and format for digital data input and output can be changed through hardware configuration settings. The entire hardware configuration can only be changed immediately after download or soft reset. The input and output port configurations, however, can also be changed following a special run-time restart message (please see Section 4.1 "Audio Manager for Dolby Digital, DTS, MP3, MPEG Multichannel, AAC and Crystal Original Surround" on page 33 for more information about the Application Restart message).

2.1 Supported Input/Output Modes

The CS4932X has two input ports and one output port. The *CS49300 Family Datasheet* describes the digital audio formats supported by the ports and gives a description of the ports themselves. The capabilities of each port are presented in the *CS49300 Family Datasheet*. Table 1 and Table 2 specify the exact input/output modes supported by each application. Please refer to the *CS49300 Family Datasheet* for more information on these hardware configurations.

The hardware modes are supported when the AC-3, DTS, MPEG Multichannel, or AAC code is active:

The hardware modes are supported by the PCM pass-through portion of the AC-3 application or when Crystal Original Surround Audio Effects Processing Code is loaded.

Hardware Configuration	Description	Supported Modes
INPUTA	Input Data Type	0,1
INPUTB	Input Data Format	0,1
INPUTC	Input SCLK/Data Edge	0,1
OUTPUTA	Output Master/Slave Clock Setup	0,1
OUTPUTB	Output Data Format	0,1,2,3
OUTPUTC	Output MCLK Rate	0,1,2,3
OUTPUTD	Output SCLK Rate	0,1,2
OUTPUTE	Output SCLK/Data Edge	0,1

Table 1. Hardware Configurations Supported By: AC-3, DTS, MP3, MPEG Multichannel, and AAC

Hardware Configuration	Description	Supported Modes
INPUTA	Input Data Type	0,1,2
INPUTB	Input Data Format	0,1
INPUTC	Input SCLK/Data Edge	0,1
OUTPUTA	Output Master/Slave Clock Setup	0,1
OUTPUTB	Output Data Format	0,1,2,3
OUTPUTC	Output MCLK Rate	0,1,2,3
OUTPUTD	Output SCLK Rate	0,1,2
OUTPUTE	Output SCLK/Data Edge	0,1

Table 2. Hardware Configurations Supported By: PCM Pass-Through (AC-3 Code) and Crystal Original Surround

2.1.1 DCLK = 86MHz

The switching characteristics of the Motorola and Intel parallel communication modes, discussed in the *CS49300 Data Sheet*, have notes that state "It should be noted that DCLK for the internal clock mode is application specific. The application code user's guide should be checked to confirm DCLK for the particular application." For all the application codes discussed in this application code user's guide, DCLK is equal to 86 MHz.

2.1.2 Configuring the XMT958/AUDAT3 Data Format

The following commands must be sent to configure the data present on the XMT958/AUDAT3 pin to S/PDIF (IEC60958) format that is driven from the supplied MCLK at 256 Fs:

> 0x800271
> 0xf7ffff
> 0x800171
> 0x021000

The following commands must be sent to configure the data present on the XMT958/AUDAT3 pin to S/PDIF (IEC60958) format that is driven from the supplied MCLK at 512 Fs:

> 0x800271
> 0xf7ffff
> 0x800171
> 0x061000

The following commands must be sent to configure the data present on the XMT958/AUDAT3 pin to I^{2}S 24-bit format that is driven from the supplied MCLK (256Fs/512Fs/128Fs/384Fs) set by the OUTPUT C Parameter in the CS49300 Data Sheet:

> 0x800271
> 0xF80FFF
> 0x800270
> 0xF01701

The following commands must be sent to configure the data present on the XMT958/AUDAT3 pin to I^{2}S 20-bit format that is driven from the supplied MCLK (256Fs/512Fs/128Fs/384Fs) set by the OUTPUT C Parameter in the CS49300 Data Sheet:

> 0x800271
> 0xF80FFF
> 0x800270
> 0xF01301

The following commands must be sent to configure the data present on the XMT958/AUDAT3 pin to Left Justified 24-bit format that is driven from the supplied MCLK (256Fs/512Fs/128Fs/384Fs) set by the OUTPUT C Parameter in the CS49300 Data Sheet:

> 0x800271
> 0xF80FFF
> 0x800270
> 0xF01700

The following commands must be sent to configure the data present on the XMT958/AUDAT3 pin to Left Justified 20-bit format that is driven from the supplied MCLK (256Fs/512Fs/128Fs/384Fs) set by the OUTPUT C Parameter in the CS49300 Data Sheet:

> 0x800271
> 0xF80FFF
> 0x800270
> 0xF01300

One of the configurations listed above must be sent before kickstart.

2.1.3 S/PDIF (AES/EBU) Channel Status Block Control

The CS4932X, in combination with all of the codes mentioned in this document, offers the "minimum" professional implementation of control over the IEC60958 channel status bits. According to the *IEC60958 Specification*, the host has control over the setting of the "PRO" bit in the channel status word (Channel Status Byte 0, Bit 0).

"Professional use of the channel status block" will be implemented if the host sets the PRO bit to a state of logic 1. These additional channel status bits may also be set by the host: Channel Status Byte 0, Bits 1-5; Channel Status Byte 1, Bits 0-7; and Channel Status Byte 3, Bits 0 and 1. Please refer to Table 3 for the IEC60958 definition for the setting of each bit or set of bits.

"Consumer use of the channel status block" will be implemented if the host sets the PRO bit to logic state 0. These additional channel status bits may also be set by the host: Channel Status Byte 0, Bits 1-5; Channel Status Byte 1, Bits 0-7; and Channel Status Byte 3, Bits 0 and 1. Please refer to Table 4 for the IEC6096 definition for the setting of each bit or set of bits.

The channel status bits map to the DSP Write Data Word in the following manner (please refer to Table 5):

- Channel Status Byte 0, Bit 0 = maps to bit 8 in the DSP Write Data Word.

- Channel Status Byte 0, Bits 1:5 = map to bits 9:13 in the DSP Write Data Word.

- Channel Status Byte 1, Bits 0:7 = map to bits 14:21 in the DSP Write Data Word.

- Channel Status Byte 3, Bits 0 and 1 = map to bits 22 and 23 in the DSP Write Data Word.

Please refer to the *IEC60958 Specification* for more a more detailed explanation of the channel status block definitions.

For example, the following command sets the Channel Status Block of the Left Subframe (Channel A) for: Consumer Mode (Byte 0, Bit0 = 0); 48kHz Fs (Byte 3, Bit 0 and 1 = 01); Category Code = General (Byte 1, Bits 0:7 = 000000000); 2 Audio Channels without Pre-Emphasis and Copy Prohibited (Byte 0, Bits 1:5 = 00000):

> 0x800272
> 0x0000FF
> 0x800172
> 0x800000

For example the following command sets the Channel Status Block of the Right Subframe (Channel B) for: Consumer Mode (Byte 0, Bit 0 = 0), 32kHz Fs (Byte 3, Bits 0 and 1 = 11); Category Code = General (Byte 1, Bits 0:7 = 000000000); 2 Audio Channels without Pre-Emphasis and Copy Prohibited (Byte 0, Bits 1:5 = 00000):

> 0x800273
> 0x0000FF
> 0x800173
> 0xC00000

The first word performs a logical "AND" of the register with the second word, which acts as a mask. The values located in the bit locations "AND'ed" with a one are preserved, while the locations "AND'ed" with a zero are cleared. The third word performs a logical "OR" of the same register with the fourth word, which contains the necessary CSW bits that the user wants to set.

2.1.4 S/PDIF (AES/EBU) Validity Bit Control

In addition to the Channel Status Block Control, the Validity bit in the IEC60958 stream may also be set by the host to indicate if the according audio sample is fit for conversion to analog.

The following command must be sent to CLEAR the IEC60958 Validity bit:

> 0x800271
> 0xFFBFFF

The following command must be sent to SET the IEC60958 Validity bit:

> 0x800171
> 0x004000

BYTE 0		
Bit	**0**	**PRO = 1**
	0	Consumer use of channel status block
	1	Professional use of channel status block
Bit	**1**	**Audio**
	0	Normal Audio
	1	Non-Audio
Bits	**2 3 4**	**Encoded audio signal emphasis**
	0 0 0	Emphasis not indicated. Receiver defaults to no emphasis with manual override enabled
	1 0 0	None. Receiver manual override disabled
	1 1 0	50/15 μS. Receiver manual override disabled
	1 1 1	CCITT J.17. Receiver manual override disabled
	X X X	All other states of bits 2-4 are reserved
Bit	**5**	**Lock: Source Sample Frequency**
	0	Locked - default
	1	Unlocked
bits	**6 7**	**Fs: Sample Frequency**
	0 0	Not indicated. Receiver default to 48 kHz and manual override or auto set enabled
	0 1	48 kHz. Manual override or auto disabled
	1 0	44.1 kHz. Manual override or auto disabled
	1 1	32 kHz. Manual override or auto disabled

BYTE 1				
Bits	**0 1 2 3**			**Channel Mode**
	0 0 0 0			Mode not indicated. Receiver default to 2-channel mode. Manual override enabled
	0 0 0 1			Two-channels. Manual override disabled
	0 0 1 0			Single channel. Manual override disabled
	0 0 1 1			Primary/Secondary (Ch. A is primary) Manual override disabled
	0 1 0 0			Stereophonic. (Ch. A is left) Manual override disabled.
	0 1 0 1			Reserved for user-defined application
	0 1 1 0			Reserved for user-defined application
	1 1 1 1			Vector to byte 3. Reserved
	X X X X			All other states of Bits 0-3 are reserved.
Bits	**4 5 6 7**			**User bits management**
	0 0 0 0			Default, no user information indicated
	0 0 0 1			192-bit block structure Preamble "Z" starts block
	0 0 1 0			Reserved
	0 0 1 1			User-defined application
	X X X X			All other states of Bits 4-7 are reserved.

BYTE 3		
Bits	**0-7**	**Vectored target byte**
	XXXXXX	Reserved

Table 3. Professional Channel Status bytes 0, 1, and 3

3. APPLICATION MESSAGING

While using the CS4932X it may be necessary to control or monitor the application to take full advantage of the rich feature set employed by the CS4932X and its software. By controlling application messaging, the user controls the application. Whether it is configuring the part after download (e.g. enabling MPEG decode), or changing run-time parameters (e.g. adjusting channel delays), the host uses application messaging to communicate with the CS4932X.

While communicating with the CS4932X using indexed modules, a strict software protocol must be used in conjunction with the hardware protocol discussed in the *CS49300 Family Data Sheet*. This section will cover both the format of the messages and the different configuration modules available for use with the CS4932X.

Caution: The host must strictly adhere to the hardware and software protocols described in the software and hardware documentation for the CS4932X part to ensure successful communication.

3.1 Indexed Module Communication Protocol

Each indexed module of the application can be thought of as a block of software registers or variables. The index identifies a unique variable within the module. When the opcode for a module and an index are combined, a unique variable can be read or written.

This section covers how to communicate with the CS4932X using indexed modules. The software protocol is presented for the following types of messages: Write, Solicited Read, Read Response, and Unsolicited Read Response.

When the protocol presented in this section is used with the application modules in Section 4 "Common Application Modules" on page 32, the host will be able to fully configure the application running on the CS4932X.

3.1.1 Write Session

A write session with the CS4932X consists of one 6-byte message from the host to the CS4932X. The write message consists of a Command word followed by an associated data word.

Table 5 shows the format of a Write Message:

Write Command Word:

23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OPCODE[7:0]								INDEX[15:0]															

Write Data Word:

23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DATA[23:0]																							

OPCODE[7:0] – 8-bit (1-byte) field containing opcode for targeted application module. This field will choose which module is written.

INDEX[15:0] – 16-bit (2-byte) field containing the index for the desired variable in the module chosen by *OPCODE*. This field will choose the actual variable to be altered.

DATA[23:0] – 24-bit (3-byte) data word to be written into the variable specified by *INDEX* in the module specified by *OPCODE.*

Table 5. Write Message Format

3.1.2 Solicited Read Message Format

A solicited read session consists of one 3-byte Read Request message from the host to the CS4932X, followed by a 6-byte Read Response message from the CS4932X to the host.

The Read Request message simply consists of a Read Command word whose format is shown in Table 6.

After the host sends the Read Command word it should wait for the INTREQ line to fall. See the *CS49300 Family Data Sheet* for more on hardware communication with the CS4932X.

After INTREQ falls, the host should read out the 6-byte Read Response message which consists of a 3-byte Read Response Command word followed by the requested data word. The format of the Read Response message is shown in Table 7.

Read Command Word:

23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OPCODE[7:0]								INDEX[15:0]															

> OPCODE[7:0] – 8-bit (1-byte) field containing opcode for targeted application module. This field will choose the module to be read.

> INDEX[15:0] – 16-bit (2-byte) field containing the index for the desired variable in the module chosen by *OPCODE*. This field will choose the actual variable to be read.

Table 6. Read Command Message Format

Read Response Command Word:

23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OPCODE[7:0]								INDEX[15:0]															

Read Response Data Word

23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DATA[23:0]																							

> OPCODE[7:0] – 8-bit (1-byte) field containing the Read Response opcode for the targeted application module. This field will show which module was read.

> INDEX[15:0] – 16-bit (2-byte) field containing the index for the desired variable in the module chosen by *OPCODE*. This field will show the actual variable that was read.

> DATA[23:0] – 24-bit (3-byte) data word that was read from the variable specified by *INDEX* in the module specified by *OPCODE.*

Table 7. Read Response Message Format

3.1.3 Unsolicited Read Message Format

Unsolicited messages will typically be used in systems where INTREQ can generate interrupts. These messages will come from the CS4932X to indicate a change in the system that must be addressed. One example is when the part is in autodetect mode and detects a new stream. An Unsolicited Read message will be sent by the CS4932X to indicate the new stream type.

The 6-byte unsolicited read messages from the CS4932X consist of a 3-byte Read Command word

which defines the type of unsolicited message (as described in Section 4.3 "Unsolicited Messages (Read-Only)" on page 54and a 3-byte associated data word that contains more information describing a system condition. Every time the existence of a message is detected (by sensing that INTREQ has gone LOW), the host should read out the 6-byte Unsolicited Read message.

Table 8 shows the format of an Unsolicited Read Message:

Unsolicited Read Command Word:

23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OPCODE[7:0]								INDEX[15:0]															

Unsolicited Read Data Word:

23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DATA[23:0]																							

OPCODE[7:0] – 8-bit (1-byte) field containing opcode for the Unsolicited Read message

INDEX[15:0] – 16-bit (2-byte) field containing the index for the variable in the module designated by *OPCODE*.

DATA[23:0] – 24-bit (3-byte) data word that corresponds to the variable specified by *INDEX* in the module specified by *OPCODE.*

Table 8. Unsolicited Read Message Format

4. COMMON APPLICATION MODULES

Both the Generalized Audio Manager and Generalized Bass Manager are generic in their use and apply to Dolby Digital, DTS, MP3, MPEG Multichannel, AAC, and Crystal Original Surround code, in addition to the codes covered in AN163 appendices.

The Generalized Pro Logic Manager is applicable to all codes with the exception of MP3 and C.O.S. codes in addition to all IBA-Based codes.

The Generalized Tone Control Manager is applicable to all codes with the exception of MP3 and AAC. The block diagram on the cover page of this document portrays the interaction between the various application modules and the AC-3, DTS, MP3, MPEG Multichannel, AAC and Crystal Original Surround code.

The following should be noted about all values in the application modules:

• Variables marked by "*Default**" will only be initialized after download. These variables will retain their values after a soft reset (or Application Restart, where applicable).

• Variables marked by *'Default'* will be re-initialized to the values shown in this application note after download, soft reset (or Application Restart, where applicable).

• Variables marked by '†' *can* be modified during runtime of any application code. Variables NOT marked by '†' *can NOT* be modified during runtime of any application code.

• Only those values which are presented as valid or which fall within the specified range should be written to the application module variables. If a value which falls outside the stated range is written to a variable, functionality of the application is not guaranteed and erroneous output could result.

• All bits that are not defined should be considered reserved and written with 0's unless specified otherwise.

• For variables such as volume and scaling factors, the real number range of 0.0-1.0 is written as 0x000000-0x7FFFFF. This range is linear, i.e. 1/4 volume, or -12 dB, is represented as 0x1FFFFF and is equivalent to a value of 0.25.

• The formula for converting variable settings from a hexadecimal number into dB is as follows:

20 * log (variable setting/max value for variable)

e.g. 20 * log (0x1FFFFF/0x7FFFFF) == -12 dB

Numbers preceded by 0x should be interpreted as hex, and numbers followed by 'b' should be interpreted as binary. All values sent to the CS4932X should be converted to hexadecimal. Likewise all values read from the part are in hexadecimal.

4.1 Audio Manager for Dolby Digital, DTS, MP3, MPEG Multichannel, AAC and Crystal Original Surround

Write Opcode = 0x88; Read Opcode = 0x09; Read Response Opcode = 0x89

Write Message = 0x8800HH 0xhhhhhh

Read Request Message = 0x0900HH;

Read Response Message = 0x8900HH 0xhhhhhh

0xHH = index, 0xhhhhhh = data value

Index	Variable	Dataword Content
0x00	AUDIO_MGR_CONTROL	**Bit 15** = Audio_Config_Change_Notification_Enable = 0/1= Disable/Enable unsolicited message notification of any change in audio configuration of any input bitstream.
		Note: This bit is applicable only for the following algorithms: AAC, AC-3, MPEG and DTS. The exact fields are specified for each application separately. Please refer to Section 4.1.1 "Audio_Control: (Index 0x00)" on page 37 for more detail.
		Bit 14 = Startup_Autodetect_Bypass_Enable= 0/1= Disable/Enable bypassing of autodetection at start-up, i.e. for the first time after download/reset. (Valid only with Autodetect_Enable==1).
		Note: If the above mode is enabled, during readback this bit is reserved and may read 0 or 1. Setting of this bit is not recommended with DTS code.
		Bit 13 = Null_Pause_Report_Enable = 0/1= Disable/Enable reporting of Null and Pause IEC61937 data types (Valid only with Autodetect_Enable==1).
		Note: Care should be taken when enabling this bit as it could cause multiple interrupts to be taken by the host. Using this bit is not suggested unless IEC61937 streams and null/pause reporting are understood in the context of how it will affect the host microcontroller and interrupts. Use of null/pause reporting is not required for a working system.
		Bit 12 = Autodetect_Enable = 0/1= Disable/Enable autodetect functionality.
		Bit 9 = Delay_Granularity 0/1 = ms/16 sample.
		Note: Not applicable to AAC code.
		Bit 8 = Application_Restart_Enable = 0/1= Disable/Enable restart of software application.
		Note: This setting is ONLY applicable for AC-3 and C.O.S. application codes. This bit should be set to 0 for all other application codes.
		Bit 4 = PLL_Enable = 0/1 = Disable/Enable Phase Locked Loop generation of DSP clock. It is mandatory that the PLL be enabled for the CS4932X.
		Bit 0 = Kickstart_Enable = 0/1 = Disable/Enable Kickstart of application.
		Default = 0x000000

Table 9. Audio Manager

Index	Variable	Dataword Content
0x01	NOISE_CONTROL	**Bit 6** = High_Pass_Filtered_White_Noise_Enable = 0/1 = Disable/Enable. † **Note:** Bit 6 and Bit 4 are mutually exclusive. Set only one HIGH. **Bit 5** = Level_Select = 0/1 = Dolby Level / High Level White Noise output. Dolby Level must be used for Dolby Certification. Refer to Section 4.1.2. † **Note:** Bit 5 should only be set HIGH if Bit 4 or Bit 6 has been set HIGH. **Bit 4** = White_Noise_Enable = 0/1 = Pink/White Noise output. † **Note:** Bit 6 and Bit 4 are mutually exclusive. Set only one HIGH. **Bits 3:0** = Output _Channel = 0...5 = L, C, R, Ls, Rs, LFE channel output. Disabled if > 5. † **Note:** Only applicable when AC-3 application code is loaded and configured for PCM pass-through. Autodetection must be disabled. *Default = 0x7FFFFF*
0x02	C_DELAY	0...15 ms Delay if Delay_Granularity = 0 0...45 * 16 sample Delay if Delay_Granularity = 1 **Note:** Not applicable to AAC Code. *Default* = 0* †
0x03	PCM_PRECISION	1...24 = Precision of output PCM. *Default* = 24*
0x04	LS_DELAY	0...40 = mS delay (automatic +15ms for Pro Logic) if Delay_Granularity = 0 0...120 * 16 sample Delay (automatic +15ms for Pro Logic) if Delay_Granularity = 1 **Note:** AAC Code only supports 15ms automatic + user defined (up to 15ms) delay for left/right surround channels when Prologic is switched on. However, when Prologic is switched off, the user settings for left/right surround delays are ignored (i.e. AAC Code does not support delays for any channels during normal operation). *Default* = 0* †
0x05	RS_DELAY	0...40 = mS delay (automatic +15ms for Pro Logic) if Delay_Granularity = 0 0...120 * 16 sample Delay (automatic +15ms for Pro Logic) if Delay_Granularity = 1 **Note:** AAC Code only supports 15ms automatic + user defined (up to 15ms) delay for left/right surround channels when Prologic is switched on. However, when Prologic is switched off, the user settings for left/right surround delays are ignored (i.e. AAC Code does not support delays for any channels during normal operation). *Default* = 0* †
0x06	MASTER_VOLUME	0x800000-0x7FFFFF (-1.0 to 1.0). *Default* = 0x7FFFFF* †
0x07	L_VOLUME	0x800000-0x7FFFFF (-1.0 to 1.0). *Default* = 0x7FFFFF* †
0x08	C_VOLUME	0x800000-0x7FFFFF (-1.0 to 1.0). *Default* = 0x7FFFFF* †
0x09	R_VOLUME	0x800000-0x7FFFFF (-1.0 to 1.0). *Default* = 0x7FFFFF* †
0x0a	LS_VOLUME	0x800000-0x7FFFFF (-1.0 to 1.0). *Default* = 0x7FFFFF* †
0x0b	RS_VOLUME	0x800000-0x7FFFFF (-1.0 to 1.0). *Default* = 0x7FFFFF* †
0x0c	LFE_VOLUME	0x800000-0x7FFFFF (-1.0 to 1.0). *Default* = 0x7FFFFF* †

Table 9. Audio Manager (Continued)

Index	Variable	Dataword Content
0x0d	MUTE	0/1 = Unmute/mute audio. This is a soft mute. *Default = 0* †
0x0e	DAO0_CHANNEL	0...7 = Channel type[a] for Digital Audio Output 0. Any channel may be mapped to any valid number of outputs. *Default* = 0(L)* † (note a)
0x0f	DAO1_CHANNEL	0...7 = Channel type[a] for Digital Audio Output 1. Any channel may be mapped to any valid number of outputs. *Default* = 2(R)* † (note a)
0x10	DAO2_CHANNEL	0...7 = Channel type[a] for Digital Audio Output 2. Any channel may be mapped to any valid number of outputs. *Default* = 3(Ls)* † (note a)
0x11	DAO3_CHANNEL	0...7 = Channel type[a] for Digital Audio Output 3. Any channel may be mapped to any valid number of outputs. *Default* = 4(Rs)* † (note a)
0x12	DAO4_CHANNEL	0...7 = Channel type[a] for Digital Audio Output 4. Any channel may be mapped to any valid number of outputs. *Default* = 1(C)* † (note a)
0x13	DAO5_CHANNEL	0...7 = Channel type[a] for Digital Audio Output 5. Any channel may be mapped to any valid number of outputs. *Default* = 5(LFE)* † (note a)
0x14	SAMPLING_FREQUENCY_CODE	0..2 = Sampling Frequency code to be set by host in case of PCM-only input applications. Irrelevant for non-PCM processing. **Note:** This setting affects both the Pro Logic and Bass Manager filter coefficients and Surround Channel Delays, as well as the deemphasis filter applied when digital deemphasis is enabled. Please refer to Section 5.2.1 "PCM_CONTROL" on page 80 for more on using digital deemphasis with IBA codes. 0 = 48 KHz (note b) 1 = 44.1 KHz (note c) 2 = 32 KHz (note d) *Default* = 0*
0x15	PCM_AUTODETECT_SILENCE_THRESHOLD	0..8388606 = Number of samples of silence after which CS4932X will declare Out-Of-PCM while playing PCM with autodetection enabled. 8388607 (0x7FFFFF) = Disable silence detection when playing PCM. CS4932X will never declare Out-Of-PCM once it starts playing PCM (equivalent to infinite threshold). **Note:** Valid only for AC-3 code when configured for PCM pass-through, in addition to all other PCM processing application codes. It is recommended that system designers set this value large enough to avoid inter-track silence from PCM Compact Discs. *Default* = 48000*
0x16	LAST_UNSOLICITED_MESSAGE	Last Unsolicited Message Value (see Section 4.3) ***READ ONLY*** *Default=0x000000*
0x17	PLL_REGISTER_1	Register 1 value for Phase Locked Loop (no Fs generation). The following values should be used for each corresponding CLKIN frequency: *0x05F1BF (CLKIN = 12.288MHz, DCLK = 86MHz, Output A1)* *0x0BF1BF (CLKIN = 24.576MHz, DCLK = 86MHz, Output A1)* *0x0d214f (CLKIN = 27MHz, DCLK = 86MHz, Output A1)* *Default* = 0x05F1BF*

Table 9. Audio Manager (Continued)

Index	Variable	Dataword Content
0x18	PLL_REGISTER_2	Register 2 value for Phase Locked Loop (no Fs generation). The following values should be used for each corresponding CLKIN frequency: *0x002F7D (CLKIN = 12.288MHz, DCLK = 86MHz, Output A1)* *0x002F7D (CLKIN = 24.576MHz, DCLK = 86MHz, Output A1)* *0x00237D (CLKIN = 27MHz, DCLK = 86MHz, Output A1)* *Default* = 0x002F7D*
0x19	L_DELAY	0...15 ms Delay if Delay_Granularity = 0 0...45 * 16 sample Delay if Delay_Granularity = 1 **Note:** Not applicable to AAC Code. *Default* = 0* †
0x1a	R_DELAY	0...15 ms Delay if Delay_Granularity = 0 0...45 * 16 sample Delay if Delay_Granularity = 1 **Note:** Not applicable to AAC Code. *Default* = 0* †
0x1b	LFE_DELAY	0...10 ms Delay if Delay_Granularity = 0 0...30 * 16 sample Delay if Delay_Granularity = 1 **Note:** Not applicable to AAC Code. *Default* = 0* †
0x1c	LZ_VOLUME	0x800000-0x7FFFFF (-1.0 to 1.0). *Default* = 0x7FFFFF* †
0x1d	RZ_VOLUME	0x800000-0x7FFFFF (-1.0 to 1.0). *Default* = 0x7FFFFF* †
0x1e	DAO6_CHANNEL	0...7 = Channel type[a] for Digital Audio Output 6. Any channel may be mapped to any valid number of outputs. *Default* = 6(Lz)* † (note a)
0x1f	DAO7_CHANNEL	0...7 = Channel type[a] for Digital Audio Output 7. Any channel may be mapped to any valid number of outputs. *Default* = 7(Rz)* † (note a)
0x20	DUALZONE_ATTENUATION	0x800000-0x7FFFFF (-1.0 to 1.0). *Default* = 0x7FFFFF* †

Table 9. Audio Manager (Continued)

Notes: a. Channel type 0...7 corresponds to Channels L, C, R, Ls, Rs, LFE, Lz, and Rz respectively. Lz and Rz refer dual-zone outputs. If the source made available to the decoder is a 5.1 Channel AC-3 stream, these outputs will be Pro Logic encoded and should be considered to be Lt and Rt. However, for a 2.0 Channel AC-3 stream, these outputs will be simply L and R.

b. Pre-loaded coefficients will provide a 80 Hz Crossover if input Fs is 48 kHz. A different set of coefficients must be downloaded for 96 kHz or 192 kHz Fs support.

c. Pre-loaded coefficients will provide a 80 Hz Crossover if input Fs is 44.1 kHz. A different set of coefficients must be downloaded for 88.2 kHz or 176.4 kHz Fs support.

d. Pre-loaded coefficients will provide a 80 Hz Crossover if input Fs is 32 kHz. A different set of coefficients must be downloaded for 64 kHz or 128 kHz Fs support.

†   See definition on .

"Default" vs. "Default*"   See definition .

4.1.1 Audio_Control: (Index 0x00)

Bit_15: *Audio_Config_Change_Notification_Enable*

Setting this bit HIGH will cause the CS4932X to generate interrupts when a change to a specific stream definition variable occurs. Stream definition variables are embedded in the compressed bit stream.

Caution: The host must wait at least 20 ms before attempting to read any of the stream definition variables after receiving an Unsolicited Message indicating that there has been a change to one of the variables. Variable values read from the DSP before this time has passed may not be valid.

For the AC-3 code if any of *acmod, bsid, lfeon, dsurmod, dialnorm* or *timecod2* variables change in the incoming stream, CS4932X will issue an unsolicited message (see Section 4.3 "Unsolicited Messages (Read-Only)" on page 54).

For the DTS code if any of *amode, lff,* or *pcmr* variables change in the incoming stream, CS4932X will issue an unsolicited message (see Section 4.3 "Unsolicited Messages (Read-Only)" on page 54).

For the MP3 code if the *mode* variable changes in the incoming stream, CS4932X will issue an unsolicited message (see Section 4.3 "Unsolicited Messages (Read-Only)" on page 54).

For the MPEG code if any of *mode, center, surround,* or *lfe* variables change in the incoming stream, CS4932X will issue an unsolicited message (see Section 4.3 "Unsolicited Messages (Read-Only)" on page 54).

For the AAC code if the *channel_configuration* variable changes in the incoming stream, CS4932X will issue an unsolicited message (see Section 4.3 "Unsolicited Messages (Read-Only)" on page 54).

    0 - Disables Audio_Config_Change_Notification
    1 - Enables Audio_Config_Change_Notification

Bit 14: *Startup_Autodetect_Bypass_Enable*

Maintains the CS4932X in autodetect mode (if Bit 12 if set), but bypasses autodetection (directly enters input data processing) at start-up, i.e., for the first time after download/reset. This bit should only be changed after download, soft reset, (or Application Restart, where applicable). The state of autodetect should not be modified during run-time.

    0 - Disables Startup_Autodetect_Bypass
    1 - Enables Startup_Autodetect_Bypass

No unsolicited message is generated by CS4932X if the processing is successful. However, if the data format is subsequently found to be the wrong type, autodetect messaging will be generated by CS4932X as usual.

The Startup_Autodetect_Bypass mode above is useful in saving the autodetect latency when the controller already knows that the current input data format is decodable by the downloaded application. The controller is usually aware of this due to the autodetection message from the previous (most recent) downloaded application.

Notes: 1. If this mode is enabled, Bit 14 readback value is reserved (either 0 or 1 may be received and should be ignored).

2. The autodetection module in the MPEG code and AAC code will assume a particular format (Pc = 5 in MPEG and Pc = 28 in AAC) and will start decoding. It is important to note that autodetection will be ALWAYS bypassed, however, an unsolicited message will be sent TWICE if the incoming data format does not have the assumed Pc. More specifically, if MPEG material has been detected while decoding a different format stream, the host will download the MPEG code with *Startup_Autodetect_Bypass_Enable* set HIGH, and will receive the Pc = 5 message. If the MPEG stream happens to have Pc = 6, then that will be reported to the host as well. Similarly, in the AAC case, the default Pc is chosen to be 28, but the Pc = 7 will be detected and reported as well.

Bit 12: *Autodetect_Enable*

Places the CS4932X into autodetect mode (see Section 4.3 "Unsolicited Messages (Read-Only)" on page 54). This bit should only be changed after download, soft reset, (or Application Restart, where applicable). The state of autodetect should not be modified during run-time.

> 0 - Disables Autodetect
> 1 - Enables Autodetect

Bit 9: *Delay Granularity*

Delay Granularity should be setup at kickstart (not after). This setting is used to choose the delay granularity as mS or in 16-sample units.

> 0 - The delay indices are interpreted as mS
>
> 1 - The delay indices are interpreted as the number of 16-sample-units to delay. In other words, the individual channel delay setting is multiplied by 16 to determine the amount of delay in samples.

Bit 8: *Application_Restart_Enable*

Used along with *Kickstart_Enable* in only the AC-3 and C.O.S. application codes, restarts the downloaded software application.

> 0 - Disables Application_Restart
> 1 - Enables Application_Restart

When enabled, this action allows input unit settings to be changed on the fly without resetting the output unit (which may cause clicks on the DAC if in master mode). This action also allows changes of processing parameters (such as filter coefficients) without resetting the output unit. Application Restart effectively places the CS4932X in a start-up state as if it had been issued a hardware reset plus a soft reset, except that the preset output and control port configuration is maintained. Except for issuing output and host initialization messages, after an Application_Restart, the host should treat the CS4932X as it would at start-up, i.e., issue input unit initialization, setup application parameters, and then kickstart.

After sending an Application Restart, the host should wait a certain amount of time after sending

Application Restart before sending any further commands to the CS4932X.

For example, here are several delay sequences that should be followed: If the AC-3 Application Code was processing an AC-3 Stream, the host must wait 35 ms. If the C.O.S. application code or AC-3 application code was processing PCM, the host must wait 12 ms.

Note: Just as with Soft Reset, all *Default** settings are untouched by Application Restart.

Bit 4: *PLL_Enable*

Enables the use of the Phase Locked Loop (PLL) to generate internal DSPCLK from the desired external CLKIN.

> 0 - Disables PLL
> 1 - Enables PLL

The register values in PLL_Register_1 (0x17) and PLL_Register_2 (0x18) define the desired DSP clock and reference CLKIN used. The default values at download, set up a 86 MHz DSPCLK from a 12.288 MHz external CLKIN. These register values should be downloaded prior to Kickstart with PLL_Enable. Use the corresponding PLL values located in Table 9 for 24.576 MHz external CLKIN or 27 MHz external CLKIN.

For other system requirements, please contact the factory for appropriate register values.

Bit 0: *Kickstart Enable*

Puts CS4932X into run-time mode. Indicates that all hardware and software configuration activities have been completed and CS4932X can start the application.

> 0 - Application continues waiting for kickstart
> 1 - Application is kickstarted

4.1.2 Noise_Control (Index 0x01): (Applicable to AC-3 Code Only)

Switches ON Pink/White Noise in Channels L, C, R, Ls, Rs and LFE. Pink/White noise will not be enabled unless the AC-3 application code is configured for PCM pass-through and the Output_Channel is set to a value in the range

0x0 - 0x5. Autodetect should not be enabled when using Pink/White Noise.

Bit 5: *Level_Select*

Level_Select sets up output level.

> 0 - Default level (Dolby-compatible)
> 1 - 10.5 dB boosted with respect to default level.

Bit 4: *White_Noise_Enable*

White_Noise_Enable enables White noise if set (otherwise output is Pink noise).

> 0 - Pink Noise
> 1 - White Noise

Bits 3:0: *Output Channel*

Output Channel selects channel for noise output

> 0x0 - Left
> 0x1 - Center
> 0x2 - Right
> 0x3 - Left Surround (This value controls both Ls and Rs in Pro Logic Mode)
> 0x4 - Right Surround (This value has no effect in Pro Logic mode)
> 0x5 - LFE
> 0x6 - Lz
> 0x7 - Rz
> 0x8-0xf - Disabled

4.1.3 Sampling Frequency Code (Index 0x14)

This parameter controls the sampling frequency for PCM applications. This must be set by the host during while a PCM application is loaded, but not during runtime.

Note: For PCM input, the DSP does NOT know the sampling frequency setting. If the system expects to receive PCM from a S/PDIF receiver, it is up to the host to inquire about the sampling frequency information from the receiver, and then inform the DSP.

For Pro Logic processing of PCM streams or when digital deemphasis is enabled (refer to Section 5.2 "PCM Manager (for AC-3 Application Code)" on page 78), the correct sampling frequency must also be given to the Pro Logic Manager for proper operation of filters. For compressed inputs, such as AC-3, DTS, MPEG, or AAC, the decoder application will set up sampling frequency information based on the information parsed from the bitstream; consequently, this field does not need to be set.

Bits 2:0: *Sampling_Frequency_Code*

> 0x0 - 48kHz
> 0x1 - 44.1kHz
> 0x2 - 32kHz

For 192 kHz or 96 kHz Stereo PCM support, the host should set the Fs to 48 kHz and download the correct 192 kHz or 96 kHz coefficients in the Generalized Bass Manager in the place of the 48 kHz coefficients.

For 176.4 kHz or 88.2 kHz Stereo PCM support, the host should set the Fs to 44.1 kHz and download the correct 176.4 kHz or 88.2 kHz coefficients in the Generalized Bass Manager in the place of the 44.1 kHz coefficients.

For 128 kHz or 64 kHz Stereo PCM support, the host should set the Fs to 32 kHz and download the correct 128 kHz or 64 kHz coefficients in the Generalized Bass Manager in the place of the 32 kHz coefficients.

Please contact your FAE for the appropriate 192 kHz, 176.4 kHz, 128 kHz, 96 kHz, 88.2 kHz or 64 kHz coefficient files, along with the desired crossover frequencies you need. These coefficients only effect the Generic Bass Manager module and do not effect the generic Pro Logic module. Currently, the Pro Logic module only supports sampling frequencies up to 48 kHz for the standard application code releases of AC-3, DTS, MPEG Multichannel, and AAC.

4.2 Generalized Bass Manager for Dolby Digital, DTS, MP3, MPEG Multichannel, AAC, and Crystal Original Surround

Write Opcode = 0x94; Read Opcode = 0x15; Read Response Opcode = 0x95

Write = 0x9400HH 0xhhhhhh

Read Request = 0x1500HH;

Read Response = 0x9500HH 0xhhhhhh

0xHH = index, 0xhhhhhh = data value

Index	Variable	Dataword Content
0x00	BASS_MGR_CONTROL	**Bit 16** = LFE_To_Sub_Enable = 0/1 = Disable/Enable LFE Only to subwoofer. † **Bit 12** = LPF_Enable = 0/1 = Disable/Enable 4th Order LPF at the output of summer. † **Bit 8** = HPF_L_Enable = 0/1/ = Disable/Enable L channel 2nd Order HPF. † **Bit 7** = HPF_C_Enable = 0/1/ = Disable/Enable C channel 2nd Order HPF. † **Bit 6** = HPF_R_Enable = 0/1/ = Disable/Enable R channel 2nd Order HPF. † **Bit 5** = HPF_Ls_Enable = 0/1/ = Disable/Enable Ls channel 2nd Order HPF. † **Bit 4** = HPF_Rs_Enable = 0/1/ = Disable/Enable Rs channel 2nd Order HPF. † **Bit 0** = Bass_Mgr_Enable = 0/1 = Disable/Enable Bass Manager post-processing. † All other bits should be 0. *Default = 0x000000*
0x01	BASS_MGR_INPUT_ L_LEVEL	0.0-1.0 = Level adjustment for input L channel pass-through. † *Default* = 0x7FFFFF*
0x02	BASS_MGR_INPUT_ C_LEVEL	0.0-1.0 = Level adjustment for input C channel input pass-through. † *Default* = 0x7FFFFF*
0x03	BASS_MGR_INPUT_ R_LEVEL	0.0-1.0 = Level adjustment for input R channel input pass-through. † *Default* = 0x7FFFFF*
0x04	BASS_MGR_INPUT_ LS_LEVEL	0.0-1.0 = Level adjustment for input Ls channel pass-through. † **Note:** Setting not applicable when Pro Logic is Enabled. Bass information from surround channels will never be input. *Default* = 0x7FFFFF*
0x05	BASS_MGR_INPUT_ RS_LEVEL	0.0-1.0 = Level adjustment for input Rs channel pass-through. † **Note:** Setting not applicable when Pro Logic is Enabled. Bass information from surround channels will never be input. *Default* = 0x7FFFFF*
0x06	BASS_MGR_INPUT_ LFE_LEVEL	0.0-1.0 = Level adjustment for input LFE channel pass-through. † *Default* = 0x7FFFFF*
0x07	BASS_MGR_SUM_ OUTPUT_L_LEVEL	0.0-1.0 = Level adjustment for bass summer output contribution to L channel output. † *Default* = 0x000000*

Table 10. Bass Manager

Index	Variable	Dataword Content
0x18	BASS_MGR_LPF_48_B1	0.0-1.0 = b1 coefficient for Low Pass Filter, Fs = 48 kHz. † (Note a) *Default* = 0x0000e4*
0x19	BASS_MGR_LPF_48_B2	0.0-1.0 = b2 coefficient for Low Pass Filter, Fs = 48 kHz. † (Note a) *Default* =0x000072*
0x1a	BASS_MGR_LPF_48_A1	0.0-1.0 = a1 coefficient for Low Pass Filter, Fs = 48 kHz. † (Note a) *Default* = 0x7f0d13*
0x1b	BASS_MGR_LPF_48_A2	0.0-1.0 = a2 coefficient for Low Pass Filter, Fs = 48 kHz. † (Note a) *Default* = 0xc0f124*
0x1c	BASS_MGR_HPF_48_B0	0.0-1.0 = b0 coefficient for High Pass Filter, Fs = 48 kHz. † (Note a) *Default* = 0x3f8745*
0x1d	BASS_MGR_HPF_48_B1	0.0-1.0 = b1 coefficient for High Pass Filter, Fs = 48 kHz. † (Note a) *Default* = 0x80f176*
0x1e	BASS_MGR_HPF_48_B2	0.0-1.0 = b2 coefficient for High Pass Filter, Fs = 48 kHz. † (Note a) *Default* = 0x3f8745*
0x1f	BASS_MGR_HPF_48_A1	0.0-1.0 = a1 coefficient for High Pass Filter, Fs = 48 kHz. † (Note e) *Default* = 0x7f0da6*
0x20	BASS_MGR_HPF_48_A2	0.0-1.0 = a2 coefficient for High Pass Filter, Fs = 48 kHz. † (Note a) *Default* = 0xc0f092*
0x21	BASS_MGR_LPF_44_B0	0.0-1.0 = b0 coefficient for Low Pass Filter, Fs = 44.1 kHz. † (Note b) *Default* = 0x000087*
0x22	BASS_MGR_LPF_44_B1	0.0-1.0 = b1 coefficient for Low Pass Filter, Fs = 44.1 kHz. † (Note b) *Default* = 0x00010e*
0x23	BASS_MGR_LPF_44_B2	0.0-1.0 = b2 coefficient for Low Pass Filter, Fs = 44.1 kHz. † (Note b) *Default* = 0x000087*
0x24	BASS_MGR_LPF_44_A1	0.0-1.0 = a1 coefficient for Low Pass Filter, Fs = 44.1 kHz. † (Note b) *Default* = 0x7ef797*
0x25	BASS_MGR_LPF_44_A2	0.0-1.0 = a2 coefficient for Low Pass Filter, Fs = 44.1 kHz. † (Note b) *Default* = 0xc1064b*
0x26	BASS_MGR_HPF_44_B0	0.0-1.0 = b0 coefficient for High Pass Filter, Fs = 44.1 kHz. † (Note b) *Default* = 0x3f7ca3*
0x27	BASS_MGR_HPF_44_B1	0.0-1.0 = b1 coefficient for High Pass Filter, Fs = 44.1 kHz. † (Note b) *Default* = 0x8106ba*
0x28	BASS_MGR_HPF_44_B2	0.0-1.0 = b2 coefficient for High Pass Filter, Fs = 44.1 kHz. † (Note f) *Default* = 0x3f7ca3*
0x29	BASS_MGR_HPF_44_A1	0.0-1.0 = a1 coefficient for High Pass Filter, Fs = 44.1 kHz. † (Note f) *Default* = 0x7ef838*
0x2a	BASS_MGR_HPF_44_A2	0.0-1.0 = a2 coefficient for High Pass Filter, Fs = 44.1 kHz. † (Note b) *Default* = 0xc105ad*
0x2b	BASS_MGR_LPF_32_B0	0.0-1.0 = b0 coefficient for Low Pass Filter, Fs = 32 kHz. † (Note c) *Default* = 0x000100*

Table 10. Bass Manager (Continued)

Index	Variable	Dataword Content
0x2c	BASS_MGR_LPF_32_B1	0.0-1.0 = b1 coefficient for Low Pass Filter, Fs = 32 kHz. † (Note c) *Default* = 0x000200*
0x2d	BASS_MGR_LPF_32_B2	0.0-1.0 = b2 coefficient for Low Pass Filter, Fs = 32 kHz. † (Note c) *Default* =0x000100*
0x2e	BASS_MGR_LPF_32_A1	0.0-1.0 = a1 coefficient for Low Pass Filter, Fs = 32 kHz. † (Note c) *Default* = 0x7e939f*
0x2f	BASS_MGR_LPF_32_A2	0.0-1.0 = a2 coefficient for Low Pass Filter, Fs = 32 kHz. † (Note c) *Default* = 0xc1685f*
0x30	BASS_MGR_HPF_32_B0	0.0-1.0 = b0 coefficient for High Pass Filter, Fs = 32 kHz^c. † (note c) *Default* = 0x2a77ee*
0x31	BASS_MGR_HPF_32_B1	0.0-1.0 = b1 coefficient for High Pass Filter, Fs = 32 kHz. † (note c) *Default* = 0xf8c361*
0x32	BASS_MGR_HPF_32_B2	0.0-1.0 = b2 coefficient for High Pass Filter, Fs = 32 kHz. † (note g) *Default* = 0x2a77ee*
0x33	BASS_MGR_HPF_32_A1	0.0-1.0 = a1 coefficient for High Pass Filter, Fs = 32 kHz. † (note c) *Default* = 0x7e947c*
0x34	BASS_MGR_HPF_32_A2	0.0-1.0 = a2 coefficient for High Pass Filter, Fs = 32 kHz. † (note c) *Default* = 0xc16787*

Table 10. Bass Manager (Continued)

Notes: a.   Pre-loaded coefficients will provide a 80 Hz Crossover if input Fs is 48 kHz. A different set of coefficients must be downloaded for 96 kHz or 192 kHz Fs support.

        b.   Pre-loaded coefficients will provide a 80 Hz Crossover if input Fs is 44.1 kHz. A different set of coefficients must be downloaded for 88.2 kHz or 176.4 kHz Fs support.

        c.   Pre-loaded coefficients will provide a 80 Hz Crossover if input Fs is 32 kHz. A different set of coefficients must be downloaded for 64 kHz or 128 kHz Fs support.

        †   See definition on .

       "Default" vs. "Default*"   See definition .

Figure 9. Subwoofer Summing Module
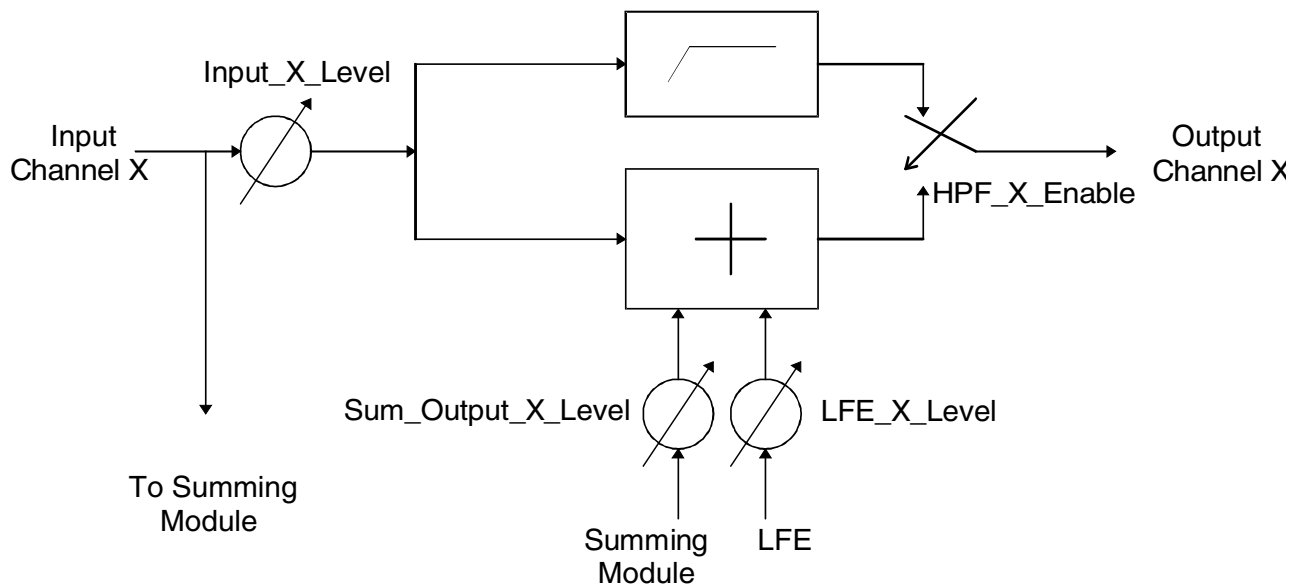
Figure 10. Bass Manager Processing Unit

4.2.2 Dolby Digital Bass Management Configurations

Tables 11 through 18 show the values that need to be written for the different Dolby Digital Bass Management configurations. All variables (except filter coefficients) can be changed during runtime. The following tables are in no way inclusive of all of the different modes possible, they are just

presented to show the different configurations required for Dolby Digital Certification. Different configurations are required depending on the level of certification that is desired. Any questions regarding certification should be directed towards Dolby Laboratories.

Bass Manager Variables	Values	Write Opcode	HEX Value
BASS_MGR_CONTROL	LFE_ONLY_SUB DISABLED LPF DISABLED HPF_L DISABLED HPF_C DISABLED HPF_R DISABLED HPF_LS DISABLED HPF_RS DISABLED BASS_MGR ENABLED	0x940000	0x000001
INPUT_L_LEVEL	0 dB	0x940001	0x7FFFFF
INPUT_C_LEVEL	0 dB	0x940002	0x7FFFFF
INPUT_R_LEVEL	0 dB	0x940003	0x7FFFFF
INPUT_LS_LEVEL	0 dB	0x940004	0x7FFFFF
INPUT_RS_LEVEL	0 dB	0x940005	0x7FFFFF
INPUT_LFE_LEVEL	0 dB	0x940006	0x7FFFFF
SUM_OUTPUT_L_LEVEL	-∞ dB	0x940007	0x000000
SUM_OUTPUT_C_LEVEL	-∞ dB	0x940008	0x000000
SUM_OUTPUT_R_LEVEL	-∞ dB	0x940009	0x000000
SUM_OUTPUT_LS_LEVEL	-∞ dB	0x94000A	0x000000
SUM_OUTPUT_RS_LEVEL	-∞ dB	0x94000B	0x000000
LFE_L_LEVEL	-∞ dB	0x94000C	0x000000
LFE_C_LEVEL	-∞ dB	0x94000D	0x000000
LFE_R_LEVEL	-∞ dB	0x94000E	0x000000
LFE_LS_LEVEL	-∞ dB	0x94000F	0x000000
LFE_RS_LEVEL	-∞ dB	0x940010	0x000000
SUM_INPUT_L_LEVEL	-15 dB	0x940011	0x16C311
SUM_INPUT_C_LEVEL	-15 dB	0x940012	0x16C311
SUM_INPUT_R_LEVEL	-15 dB	0x940013	0x16C311
SUM_INPUT_LS_LEVEL	-15 dB	0x940014	0x16C311
SUM_INPUT_RS_LEVEL	-15 dB	0x940015	0x16C311
SUM_INPUT_LFE_LEVEL	-5 dB	0x940016	0x47FACD

Table 11. Dolby Configuration 0

Bass Manager Variables	Values	Write Opcode	HEX Value
BASS_MGR_CONTROL	LFE_Only_Sub Disabled LPF Enabled HPF_L Enabled HPF_C Enabled HPF_R Enabled HPF_Ls Enabled HPF_Rs Enabled Bass_Mgr Enabled	0x940000	0x0011F1
INPUT_L_LEVEL	0 dB	0x940001	0x7FFFFF
INPUT_C_LEVEL	0 dB	0x940002	0x7FFFFF
INPUT_R_LEVEL	0 dB	0x940003	0x7FFFFF
INPUT_LS_LEVEL	0 dB	0x940004	0x7FFFFF
INPUT_RS_LEVEL	0 dB	0x940005	0x7FFFFF
INPUT_LFE_LEVEL	0 dB	0x940006	0x7FFFFF
SUM_OUTPUT_L_LEVEL	-∞ dB	0x940007	0x000000
SUM_OUTPUT_C_LEVEL	-∞ dB	0x940008	0x000000
SUM_OUTPUT_R_LEVEL	-∞ dB	0x940009	0x000000
SUM_OUTPUT_LS_LEVEL	-∞ dB	0x94000A	0x000000
SUM_OUTPUT_RS_LEVEL	-∞ dB	0x94000B	0x000000
LFE_L_LEVEL	-∞ dB	0x94000C	0x000000
LFE_C_LEVEL	-∞ dB	0x94000D	0x000000
LFE_R_LEVEL	-∞ dB	0x94000E	0x000000
LFE_LS_LEVEL	-∞ dB	0x94000F	0x000000
LFE_RS_LEVEL	-∞ dB	0x940010	0x000000
SUM_INPUT_L_LEVEL	-15 dB	0x940011	0x16C311
SUM_INPUT_C_LEVEL	-15 dB	0x940012	0x16C311
SUM_INPUT_R_LEVEL	-15 dB	0x940013	0x16C311
SUM_INPUT_LS_LEVEL	-15 dB	0x940014	0x16C311
SUM_INPUT_RS_LEVEL	-15 dB	0x940015	0x16C311
SUM_INPUT_LFE_LEVEL	-5 dB	0x940016	0x47FACD

Table 12. Dolby Configuration 1

Bass Manager Variables	Values	Write Opcode	HEX Value
BASS_MGR_CONTROL	LFE_Only_Sub Disabled LPF Enabled HPF_L Disabled HPF_C Enabled HPF_R Disabled HPF_Ls Enabled HPF_Rs Enabled Bass_Mgr Enabled	0x940000	0x0010B1
INPUT_L_LEVEL	-12 dB	0x940001	0x2026F2
INPUT_C_LEVEL	ADJ, default = 0 dB	0x940002	0x7FFFFF
INPUT_R_LEVEL	-12 dB	0x940003	0x2026F2
INPUT_LS_LEVEL	ADJ, default = 0 dB	0x940004	0x7FFFFF
INPUT_RS_LEVEL	ADJ, default = 0 dB	0x940005	0x7FFFFF
INPUT_LFE_LEVEL	0 dB	0x940006	0x7FFFFF
SUM_OUTPUT_L_LEVEL	-1.5 dB	0x940007	0x6BB2D5
SUM_OUTPUT_C_LEVEL	-∞ dB	0x940008	0x000000
SUM_OUTPUT_R_LEVEL	-1.5 dB	0x940009	0x6BB2D5
SUM_OUTPUT_LS_LEVEL	-∞ dB	0x94000A	0x000000
SUM_OUTPUT_RS_LEVEL	-∞ dB	0x94000B	0x000000
LFE_L_LEVEL	-∞ dB	0x94000C	0x000000
LFE_C_LEVEL	-∞ dB	0x94000D	0x000000
LFE_R_LEVEL	-∞ dB	0x94000E	0x000000
LFE_LS_LEVEL	-∞ dB	0x94000F	0x000000
LFE_RS_LEVEL	-∞ dB	0x940010	0x000000
SUM_INPUT_L_LEVEL	-∞ dB	0x940011	0x000000
SUM_INPUT_C_LEVEL	-15 dB	0x940012	0x16C311
SUM_INPUT_R_LEVEL	-∞ dB	0x940013	0x000000
SUM_INPUT_LS_LEVEL	-15 dB	0x940014	0x16C311
SUM_INPUT_RS_LEVEL	-15 dB	0x940015	0x16C311
SUM_INPUT_LFE_LEVEL	-5 dB	0x940016	0x47FACD

Table 13. Dolby Configuration 2

Bass Manager Variables	Values	Write Opcode	HEX Value
BASS_MGR_CONTROL	LFE_Only_Sub Disabled LPF Enabled HPF_L Disabled HPF_C Enabled HPF_R Disabled HPF_Ls Enabled HPF_Rs Enabled Bass_Mgr Enabled	0x940000	0x0010B1
INPUT_L_LEVEL	0 dB	0x940001	0x7FFFFF
INPUT_C_LEVEL	0 dB	0x940002	0x7FFFFF
INPUT_R_LEVEL	0 dB	0x940003	0x7FFFFF
INPUT_LS_LEVEL	0 dB	0x940004	0x7FFFFF
INPUT_RS_LEVEL	0 dB	0x940005	0x7FFFFF
INPUT_LFE_LEVEL	0 dB	0x940006	0x7FFFFF
SUM_OUTPUT_L_LEVEL	-∞ dB	0x940007	0x000000
SUM_OUTPUT_C_LEVEL	-∞ dB	0x940008	0x000000
SUM_OUTPUT_R_LEVEL	-∞ dB	0x940009	0x000000
SUM_OUTPUT_LS_LEVEL	-∞ dB	0x94000A	0x000000
SUM_OUTPUT_RS_LEVEL	-∞ dB	0x94000B	0x000000
LFE_L_LEVEL	-∞ dB	0x94000C	0x000000
LFE_C_LEVEL	-∞ dB	0x94000D	0x000000
LFE_R_LEVEL	-∞ dB	0x94000E	0x000000
LFE_LS_LEVEL	-∞ dB	0x94000F	0x000000
LFE_RS_LEVEL	-∞ dB	0x940010	0x000000
SUM_INPUT_L_LEVEL	-∞ dB	0x940011	0x000000
SUM_INPUT_C_LEVEL	-15 dB	0x940012	0x16C311
SUM_INPUT_R_LEVEL	-∞ dB	0x940013	0x000000
SUM_INPUT_LS_LEVEL	-15 dB	0x940014	0x16C311
SUM_INPUT_RS_LEVEL	-15 dB	0x940015	0x16C311
SUM_INPUT_LFE_LEVEL	-5 dB	0x940016	0x47FACD

Table 14. Dolby Alternative Configuration 2

Bass Manager Variables	Values	Write Opcode	HEX Value
BASS_MGR_CONTROL	LFE_Only_Sub Disabled LPF Enabled HPF_L Disabled HPF_C Enabled HPF_R Disabled HPF_Ls Disabled HPF_Rs Disabled Bass_Mgr Enabled	0x940000	0x001081
INPUT_L_LEVEL	-8 dB	0x940001	0x32F52C
INPUT_C_LEVEL	-8 dB	0x940002	0x32F52C
INPUT_R_LEVEL	-8 dB	0x940003	0x32F52C
INPUT_LS_LEVEL	-8 dB	0x940004	0x32F52C
INPUT_RS_LEVEL	-8 dB	0x940005	0x32F52C
INPUT_LFE_LEVEL	-8 dB	0x940006	0x32F52C
SUM_OUTPUT_L_LEVEL	-4.5 dB	0x940007	0x4C3EA7
SUM_OUTPUT_C_LEVEL	$-\infty$ dB	0x940008	0x000000
SUM_OUTPUT_R_LEVEL	-4.5 dB	0x940009	0x4C3EA7
SUM_OUTPUT_LS_LEVEL	$-\infty$ dB	0x94000A	0x000000
SUM_OUTPUT_RS_LEVEL	$-\infty$ dB	0x94000B	0x000000
LFE_L_LEVEL	-8 dB	0x94000C	0x32F52C
LFE_C_LEVEL	$-\infty$ dB	0x94000D	0x000000
LFE_R_LEVEL	-8 dB	0x94000E	0x32F52C
LFE_LS_LEVEL	-8 dB	0x94000F	0x32F52C
LFE_RS_LEVEL	-8 dB	0x940010	0x32F52C
SUM_INPUT_L_LEVEL	$-\infty$ dB	0x940011	0x000000
SUM_INPUT_C_LEVEL	-8 dB	0x940012	0x32F52C
SUM_INPUT_R_LEVEL	$-\infty$ dB	0x940013	0x000000
SUM_INPUT_LS_LEVEL	$-\infty$ dB	0x940014	0x000000
SUM_INPUT_RS_LEVEL	$-\infty$ dB	0x940015	0x000000
SUM_INPUT_LFE_LEVEL	$-\infty$ dB	0x940016	0x000000

Table 15. Dolby Configuration 3 (Normal - No Sub Out)

Bass Manager Variables	Values	Write Opcode	HEX Value
BASS_MGR_CONTROL	LFE_Only_Sub Disabled LPF Enabled HPF_L Disabled HPF_C Enabled HPF_R Disabled HPF_Ls Disabled HPF_Rs Disabled Bass_Mgr Enabled	0x940000	0x011081
INPUT_L_LEVEL	-4 dB	0x940001	0x50C335
INPUT_C_LEVEL	-4 dB	0x940002	0x50C335
INPUT_R_LEVEL	-4 dB	0x940003	0x50C335
INPUT_LS_LEVEL	-4 dB	0x940004	0x50C335
INPUT_RS_LEVEL	-4 dB	0x940005	0x50C335
INPUT_LFE_LEVEL	-4 dB	0x940006	0x50C335
SUM_OUTPUT_L_LEVEL	-4.5 dB	0x940007	0x4C3EA7
SUM_OUTPUT_C_LEVEL	-∞ dB	0x940008	0x000000
SUM_OUTPUT_R_LEVEL	-4.5 dB	0x940009	0x4C3EA7
SUM_OUTPUT_LS_LEVEL	-∞ dB	0x94000A	0x000000
SUM_OUTPUT_RS_LEVEL	-∞ dB	0x94000B	0x000000
LFE_L_LEVEL	-∞ dB	0x94000C	0x000000
LFE_C_LEVEL	-∞ dB	0x94000D	0x000000
LFE_R_LEVEL	-∞ dB	0x94000E	0x000000
LFE_LS_LEVEL	-∞ dB	0x94000F	0x000000
LFE_RS_LEVEL	-∞ dB	0x940010	0x000000
SUM_INPUT_L_LEVEL	-∞ dB	0x940011	0x000000
SUM_INPUT_C_LEVEL	-4 dB	0x940012	0x50C335
SUM_INPUT_R_LEVEL	-∞ dB	0x940013	0x000000
SUM_INPUT_LS_LEVEL	-∞ dB	0x940014	0x000000
SUM_INPUT_RS_LEVEL	-∞ dB	0x940015	0x000000
SUM_INPUT_LFE_LEVEL	-∞ dB	0x940016	0x000000

Table 16. Dolby Configuration 3 (Sub Out)

Bass Manager Variables	Values	Write Opcode	HEX Value
BASS_MGR_CONTROL	LFE_Only_Sub Enabled LPF Enabled HPF_L Disabled HPF_C Enabled HPF_R Disabled HPF_Ls Enabled HPF_Rs Enabled Bass_Mgr Enabled	0x940000	0x0110B1
INPUT_L_LEVEL	0 dB	0x940001	0x7FFFFF
INPUT_C_LEVEL	0 dB	0x940002	0x7FFFFF
INPUT_R_LEVEL	0 dB	0x940003	0x7FFFFF
INPUT_LS_LEVEL	0 dB	0x940004	0x7FFFFF
INPUT_RS_LEVEL	0 dB	0x940005	0x7FFFFF
INPUT_LFE_LEVEL	-5 dB	0x940006	0x47FACC
SUM_OUTPUT_L_LEVEL	0 dB	0x940007	0x7FFFFF
SUM_OUTPUT_C_LEVEL	$-\infty$ dB	0x940008	0x000000
SUM_OUTPUT_R_LEVEL	0 dB	0x940009	0x7FFFFF
SUM_OUTPUT_LS_LEVEL	$-\infty$ dB	0x94000A	0x000000
SUM_OUTPUT_RS_LEVEL	$-\infty$ dB	0x94000B	0x000000
LFE_L_LEVEL	$-\infty$ dB	0x94000C	0x000000
LFE_C_LEVEL	$-\infty$ dB	0x94000D	0x000000
LFE_R_LEVEL	$-\infty$ dB	0x94000E	0x000000
LFE_LS_LEVEL	$-\infty$ dB	0x94000F	0x000000
LFE_RS_LEVEL	$-\infty$ dB	0x940010	0x000000
SUM_INPUT_L_LEVEL	$-\infty$ dB	0x940011	0x000000
SUM_INPUT_C_LEVEL	-4.5 dB	0x940012	0x4C3EA7
SUM_INPUT_R_LEVEL	$-\infty$ dB	0x940013	0x000000
SUM_INPUT_LS_LEVEL	-4.5 dB	0x940014	0x4C3EA7
SUM_INPUT_RS_LEVEL	-4.5 dB	0x940015	0x4C3EA7
SUM_INPUT_LFE_LEVEL	$-\infty$ dB	0x940016	0x000000

Table 17. Dolby DVD Configuration (Bass to L/R)

Bass Manager Variables	Values	Write Opcode	HEX Value
BASS_MGR_CONTROL	LFE_Only_Sub Disabled LPF Enabled HPF_L Disabled HPF_C Enabled HPF_R Disabled HPF_Ls Enabled HPF_Rs Enabled Bass_Mgr Enabled	0x940000	0x0010B1
INPUT_L_LEVEL	0 dB	0x940001	0x7FFFFF
INPUT_C_LEVEL	0 dB	0x940002	0x7FFFFF
INPUT_R_LEVEL	0 dB	0x940003	0x7FFFFF
INPUT_LS_LEVEL	0 dB	0x940004	0x7FFFFF
INPUT_RS_LEVEL	0 dB	0x940005	0x7FFFFF
INPUT_LFE_LEVEL	0 dB	0x940006	0x7FFFFF
SUM_OUTPUT_L_LEVEL	-∞ dB	0x940007	0x000000
SUM_OUTPUT_C_LEVEL	-∞ dB	0x940008	0x000000
SUM_OUTPUT_R_LEVEL	-∞ dB	0x940009	0x000000
SUM_OUTPUT_LS_LEVEL	-∞ dB	0x94000A	0x000000
SUM_OUTPUT_RS_LEVEL	-∞ dB	0x94000B	0x000000
LFE_L_LEVEL	-∞ dB	0x94000C	0x000000
LFE_C_LEVEL	-∞ dB	0x94000D	0x000000
LFE_R_LEVEL	-∞ dB	0x94000E	0x000000
LFE_LS_LEVEL	-∞ dB	0x94000F	0x000000
LFE_RS_LEVEL	-∞ dB	0x940010	0x000000
SUM_INPUT_L_LEVEL	-∞ dB	0x940011	0x000000
SUM_INPUT_C_LEVEL	-15 dB	0x940012	0x16C310
SUM_INPUT_R_LEVEL	-∞ dB	0x940013	0x000000
SUM_INPUT_LS_LEVEL	-15 dB	0x940014	0x16C310
SUM_INPUT_RS_LEVEL	-15 dB	0x940015	0x16C310
SUM_INPUT_LFE_LEVEL	-5 dB	0x940016	0x47FACC

Table 18. Dolby DVD Configuration (Bass to Subwoofer)

4.3 Unsolicited Messages (Read-Only)

No Write Message. No Read Request.

Unsolicited Read Response = 0x8700HH 0xhhhhhh

0xHH = index, 0xhhhhhh = data value

Index	Variable	Dataword Content

0x00	AUTODETECT_ RESPONSE	**Bit 23** = Decodable_Stream_Flag= 0/1 = This stream is not/is decodable by the application (no need for new download if 1).
		Bits 22:6 = Reserved.
		Bit 5 Non_IEC61937_Stream_Flag= 1/0 = This stream is not/is IEC61937 compressed data.
		If Non_IEC61937_Stream_Flag=1
		Bits 4:0 = Non_IEC61937 Stream Descriptor.
		    0x00 = Silent Input Data (Out of Application Sync).
		    0x01 = DTS Format-16 elementary stream.
		    0x02 = DTS Format-14 elementary stream.
		    0x03 = Linear PCM stream.
		    0x04 = HDCD PCM stream (only available in HDCD application).
		    0x05-0x1F = Reserved.
		If Non_IEC61937_Stream_Flag=0
		Bits 4:0 = IEC61937 Stream Descriptor = Identical to bits [4:0] of the Pc burst data type descriptor in IEC61937 specification. Description of the data-type field of Pc reproduced below from *IEC61937 Specification* (current as of 11/97):
		    0x00 = *Never Reported. (Null data is ignored by CS4932X, unless Bit 13 in Audio_Mgr_Control is set, see Section 4.1).*
		    0x01 = AC-3 data.
		    0x02 = Reserved.
		    0x03 = *Never Reported. (Pause is ignored by CS4932X, unless Bit 13 in Audio_Mgr_Control is set, see Section 4.1).*
		    0x04 = MPEG-1 Layer 1 data.
		    0x05 = MPEG-1 Layer 2 or 3 data or MPEG-2 without extension.
		    0x06 = MPEG-2 data with extension.
		    0x07 = Reserved.
		    0x08 = MPEG-2 Layer 1 Low sampling frequency.
		    0x09 = MPEG-2 Layer 2 or 3 Low sampling frequency.
		    0x0A = Reserved.
		    0x0B = DTS-1 data (512-sample bursts).
		    0x0C = DTS-2 data (1024-sample bursts).
		    0x0D = DTS-3 data (2048-sample bursts).
		    0x0E - 0x1B = Reserved.
		    0x1C = MPEG-2 AAC ADTS data.
		    0x1D - 0x1F = Reserved.

0x03	AUDIO_ CONFIGURATION_ CHANGE	**Bits 23:0** = Reserved. Please see Section 4.1.1 "Audio_Control: (Index 0x00)" on page 37 bit 15 for more information.
0x10	PLL_OUT_OF_LOCK	**Bit 23** = 1. **Bits 22:0** = Reserved.

Table 19. Unsolicited Messages

4.3.1 Autodetect Operation

The sequence of events involving autodetection are described below from the host's perspective:

Note: This example assumes CS49326, CS49328, or CS49329 is being used since only these devices can play DTS. In the case of DTS being detected on CS49325 or AC-3 detected on CS49326, the host should display an appropriate user message to the front panel stating that the detected stream (DTS or AC-3) cannot be played on this receiver.

1) Host downloads CS4932X with a tentative application code, for this example we will use AC3I32X7.LD.

2) Host then configures the CS4932X hardware appropriately and sets up application parameters as desired, including enabling the desired application. For this example we will say the code is configured for AC-3 decode.

3) Host then kickstarts CS4932X with Autodetect enabled (see Section 4.1 "Audio Manager for Dolby Digital, DTS, MP3, MPEG Multichannel, AAC and Crystal Original Surround" on page 33 for details).

4) The Autodetect module of the enabled application of the CS4932X analyzes the input for a maximum of 100 ms of non-silent/non-pause data and determines the content of the input bitstream.

5a) If the enabled application can play the detected input (i.e., if AC-3 was detected in this case), then the CS4932X issues an Unsolicited Message to the host indicating the data type with Decodable_Bitstream_Flag=1. In our example of the IEC61937 AC-3 stream, the message would be 0x870000 0x800001. The CS4932X then goes ahead and processes it according to the application parameters as setup in Step 2 above.

5b) If the enabled application cannot play the detected input (say Non-IEC61937 LD DTS was detected), then the CS4932X soft mutes the

outputs, and issues an Unsolicited Message to the host indicating the data type with Decodable_Bitstream_Flag=0. In our example, the message would be 0x870000 0x000021.

On receiving this message, host repeats Steps 1 onwards but this time downloads the DTS code to the CS4932X (along with correct hardware configuration and application configuration for DTS with Autodetect enabled). Subsequently, DTS will be detected within 100 mS and successfully played by the new DTS code, after sending the corresponding unsolicited message (0x870000 0x800021).

If the host desires, it can reduce the latency of this second autodetection, by enabling Startup_Autodetect_Bypass in the Kickstart. This will allow the DTS decoder to directly start decoding the input data (which is known to be DTS), while still retaining subsequent autodetect functionality.

6) After the steps above have taken place and while CS4932X is successfully playing the input bitstream (still AC-3 in our example), if the host receives external information that the input has been changed (e.g., the user selects a new source using the front panel buttons), then before switching the input data to the CS4932X, the host should send a Soft Reset (or an Application Restart, for AC-3 or C.O.S. code) message. (See Section 4.1 "Audio Manager for Dolby Digital, DTS, MP3, MPEG Multichannel, AAC and Crystal Original Surround" on page 33.)

If Application Restart is used, this effectively puts CS4932X in Step 2, without changing the output hardware configuration (i.e., output clocks are not interrupted). The input configuration would need to be changed if it is different than the default. If Soft Reset is issued, then the entire hardware configuration needs to be resent.

The host should then repeat Steps 2, 3, 4, 5a/b as described above after delivering the new input stream to the CS4932X.

If the new input content is detected as unchanged (still AC-3 in our example), the CS4932X responds and continues processing it as in Step 5a. This situation will happen if the new stream selected by the user is also AC-3.

If the input content is detected as different (non-AC-3 in our example), the CS4932X responds as in Step 5b and continues monitoring the input stream for change in content.

7) During runtime, while successfully playing the input bitstream, the CS4932X also simultaneously monitors the input.

Note: The CS4932X has only one active input. The "input" is defined as the pin receiving data for which the application is configured (e.g. If the application is configured for PCM, the pin receiving PCM data is the input. If the application is configured for compressed data, the pin receiving compressed data is the input). As soon as the CS4932X detects a change in the bitstream (no longer AC-3, in our original example), then the CS4932X automatically reverts to Step 4., i.e., analyzes the input to determine the content. This is an automatic version of Step 6 above, but is intended to cover only the cases where the host is not aware of any possible upstream content changes. Whenever possible, the host should convey information about a possible change in input as in Step 6.

If the input content is detected as different (non-AC-3 in our example), the CS4932X reverts to Step 5b.

For compressed data streams the code will also report the Unsolicited Message 0x870000 0x800020 to indicate an out-of-sync condition when the decoder loses sync with the incoming stream due to silent input data. This is an informative message and no action is needed by the host.

If the input content is detected as unchanged (still AC-3 in our example), the CS4932X continues processing it like in Step 5a, without requiring any further action from the host. This situation could arise due to a pause or track change upstream in the source (such as a DVD player).

In the case of compressed data being played currently (like AC-3 in our example), the host could see an out-of-sync Unsolicited Message followed by an Unsolicited Message indicating MPEG data. The presence of out-of-sync messages due to "special" functions like pause or track change is completely dependent upon the audio source (such as the DVD player).

In the case of a PCM application that is currently active, if the silence is less than PCM_Autodetect_Silence_Threshold (default 48000 samples, i.e. 1 Second at 48 KHz) before transitioning to new PCM, the CS4932X continues to process the input data as if no change had occurred. In case this threshold is set to $8388607 = 0x7fffff$, then silence is completely ignored (threshold is effectively infinite).

However, during PCM processing, if the silence is more than PCM_Autodetect_Silence_Threshold, the CS4932X jumps to a Silent Input Data state, and the output is muted (transparent due to silent input anyway). Transition to this Silent Input Data state is reported via an Unsolicited Message (0x870000 0x800020). This message is informative only, and no action should be taken by the host. CS4932X is effectively in Step 4 above now, waiting to autodetect the input once non-silent data appears. Once input data is detected, a corresponding Unsolicited Message is issued to the host and the process continues as above. Please see Figure 11, for an overview of the previously described flow.
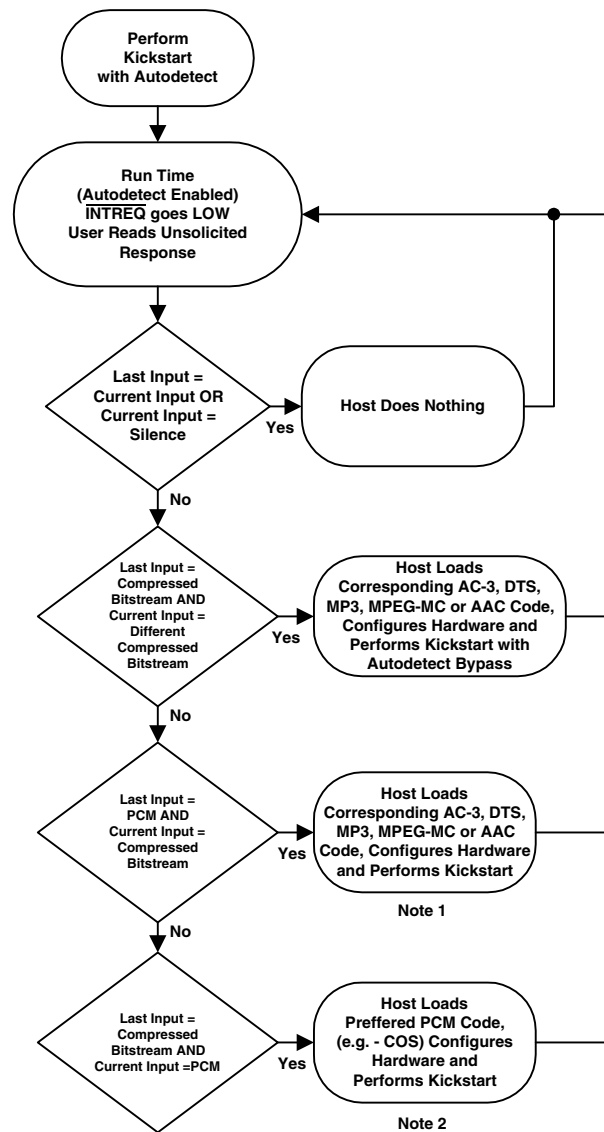
Figure 11. Generic Autodetect Flow Chart

Notes: 1. If AC-3 is detected, the host does not need to load AC-3 code if AC-3 is already loaded and passing-through PCM. The host only needs to: (a) Perform a Soft Reset, Configure the Hardware and perform a Kickstart. (b) Or, perform an Application Restart, Configure the Inputs, and perform a Kickstart.

2. If PCM is detected, host does need to reload AC-3 code if AC-3 is already loaded, as it will pass-through PCM. The host only needs to: (a) Perform a Soft Reset, Configure the Hardware and perform a Kickstart. (b) Or, perform an Application Restart, Configure the Inputs, and perform a Kickstart.

4.3.2 Special Considerations for Autodetection

First, autodetect should only be enabled or disabled when sending the Kickstart command. Changing the state of Autodetect at any other time can produce unpredictable results as it is a function of the input and the application that is currently enabled. If the host needs to change the state of Autodetect at run time for some reason, a soft reset (or Application Restart, if possible) should be sent.

The application must be enabled before sending the autodetect enable with Kickstart. Particularly, for the AC-3 code, either AC-3 or PCM must be enabled *before* the autodetect message is sent with the kickstart. If the Autodetect function senses the data for the enabled application, then the appropriate Unsolicited Message is sent and decode (or PCM processing) will start automatically without any direction from the host. If the host desires complete control of the output, then it should set master volume to 0 before enabling Autodetect. Upon receiving the autodetect message that the appropriate data is present at the input, the host would then set the master volume to the desired level. Alternatively, the host could use Mute instead of Master_Volume.

In summary this is the recommended procedure when using autodetect:

1) Download code (or Soft Reset or Application Restart)

2) Hardware Configuration (for download or Soft Reset) / Input Configuration (for Application Restart)

3) Application Configuration (including enable of desired application)

4) Kickstart with autodetection enabled. Please see Figure 12, for an overview of the previously described summary.
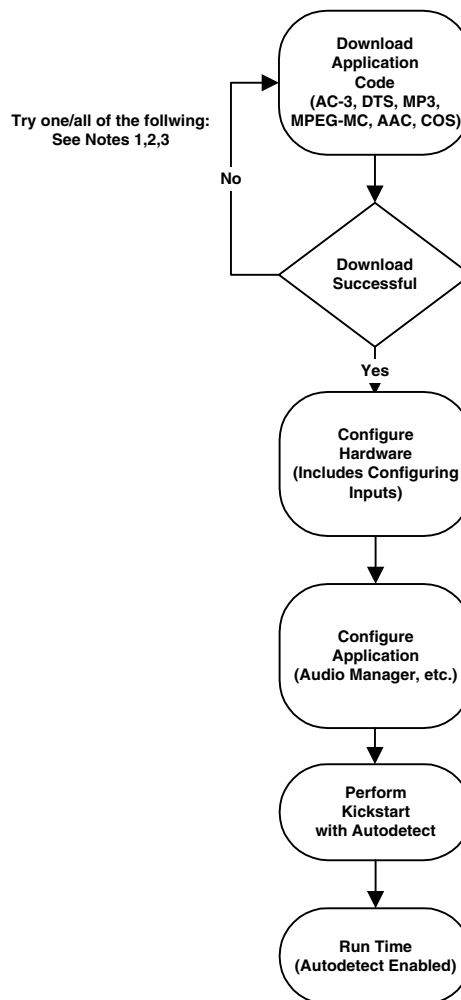
Figure 12. Typical Download Sequence

Notes:
1. Check .LD file version. Contact your FAE for the most recent version.
2. Check to make sure .LD file is correct for specific chip. (E.G. code for CS49329 will NOT load on a CS49326).
3. Check communications protocol technique. Check microcontroller code. Check power-up pin logic. Remove other devices from the communications bus (to rule-out bus contention).

4.4 Generalized Pro Logic Manager for Dolby Digital, DTS, MPEG Multichannel and AAC

Write Opcode = 0x8E; Read Opcode = 0x0F; Read Response Opcode = 0x8F

Write = 0x8E00HH 0xhhhhhh

Read Request = 0x0F00HH;

Read Response = 0x8F00HH 0xhhhhhh

0xHH = index, 0xhhhhhh = data value

Index	Variable	Dataword Content
0x00	PRO_LOGIC_CONTROL	**Bit 8** = Autobalance_Enable = 0/1= Disable/Enable Autobalance. †
		Bit 4 = Btype_NR_Enable = 0/1= Disable/Enable B-Type Noise Reduction. †
		Bit 0 = ProLogic_Enable = 0/1 = Disable/Enable Pro Logic decoding. †
		Default = 0x000000

Table 20. Generalized Pro Logic Manager

For AC-3, the code will *not* automatically enable Pro Logic if the stream is 2/0 Dolby Surround encoded. This task is the responsibility of the host. The end-user *may* choose to override the host controller's default setting when a Dolby Surround encoded 2/0 AC-3 stream is detected. This ability to disable or enable Pro Logic decoding is controlled via commands from this generalized Pro Logic Manager. If the input is NOT Dolby Surround encoded or is more or less than 2 channels, it is not recommended that Pro Logic decoding occur. For PCM input, the AC-3 module is still used, but is configured for PCM Pass-Through and AC-3 decoding is disabled. Please refer to Section 4.4.2 "Pro Logic decode of 48 kHz input PCM (Using AC-3 Application Code)" on page 61 for an example of how to configure the AC-3 code for PCM Pass-Through and Pro Logic decoding.

For DTS, when the input stream is 2/0 Stereo or Dolby Surround encoded material, this module gives the additional flexibility to perform a Pro Logic decode. If the input is NOT Dolby Surround encoded or is more or less than 2 channels, it is not recommended that Pro Logic decoding occur.

For MPEG, when the input stream is 2/0 Stereo or Dolby Surround encoded material, this module gives the additional flexibility to perform a Pro Logic decode. If the input is NOT Dolby Surround encoded, two independent mono channels, or is more or less than 2 channels, it is not recommended that Pro Logic decoding occur.

For AAC, when the input stream is 2/0 Stereo or Dolby Surround encoded material, this module gives the additional flexibility to perform a Pro Logic decode. If the input is NOT Dolby Surround encoded, two independent mono channels, or is more or less than 2 channels, it is not recommended that Pro Logic decoding occur.

4.4.1 Using Pro Logic in Normal, Wide and Phantom Modes

4.4.1.1 Normal Mode

Figure 13 is a example of what commands must be sent to the CS4932X, in order to enable Pro Logic

in Normal mode, when decoding AC-3. This mode is especially helpful for systems that contain a center channel that is not capable of reproducing low frequency information (included in the full-bandwidth range that is directed to the center channel).

```
#-------BASS_MGR_CONTROL for Dolby config 3 subwoofer on------
940000
011081
#-------BASS_MGR_INPUT_LEVELS-----
940001
50c335
940002
50c335
940003
50c335
940004
50c335
940005
7fffff
940006
50c335
#-------BASS_MGR_SUM_OUTPUT_LEVELS-----
940007
4c3ea8
940008
000000
940009
4c3ea8
94000a
000000
94000b
000000
#-------BASS_MGR_LFE_OUTPUT_LEVELS-----
94000c
000000
94000d
000000
94000e
000000
94000f
000000
940010
000000
#-------BASS_MGR_SUM_INPUT_LEVELS-----
940011
000000
940012
50c335
940013
000000
940014
000000
940015
000000
940016
000000
#-------Enables LFE Channel (if present) and AC-3 Decoding------
8a0000
000011
#-------Enable Dolby Pro Logic processing, Dolby B N.R., and Autobalance------
8e0000
000111
```

Figure 13. Normal Mode

4.4.1.2 Wide Mode

When Pro Logic is enabled, without the Bass Manager enabled, the CS4932X is (by default) placed in Wide mode, when decoding AC-3. A summary of the commands enable Wide mode is found in Table 12. This mode is especially helpful for systems that contain a center channel that is capable of reproducing low frequency information (included in the full-bandwidth that is directed to the center channel).

4.4.1.3 Phantom Mode

Figure 15 is a summary of what commands must be sent to the CS4932X, in order to enable Pro Logic in Phantom mode, when decoding AC-3. This mode is especially helpful for systems do not contain a center channel.

4.4.2 Pro Logic decode of 48 kHz input PCM (Using AC-3 Application Code)

In this mode the input should be 48 kHz Pro Logic encoded PCM data. The data will be taken from the PCM input port as designated by the Hardware Configuration message.

```
#-------Enables LFE Channel (if present) and AC-3 Decoding-------
8a0000
000011

#-------Enable Dolby Pro Logic processing, Dolby B N.R., and Autobalance-------
8e0000
000111
```

Figure 14. Wide Mode

```
#-------Enables LFE Channel (if present) and AC-3 Decoding-------
8a0000
000011

#-------Enable Dolby Pro Logic processing, Dolby B N.R., and Autobalance-------
8e0000
000111

# Enable Phantom Mode - Output Mode set to L, R, Ls, Rs (Ls=Rs=S-3dB)-------
8a0001
000006
```

Figure 15. Phantom Mode

Module	Index	Description	Opcode & Index	Value
AC-3 Manager	AC3_CONTROL	AC-3 Decoding Disabled	0x8A0000	0x000000
PCM Manager	PCM_Control	PCM Pass-Through Enabled	0x9c0000	0x000001
Pro Logic Manager	Pro_Logic_Control	Enabled Autobalance Enabled B-Type NR Enabled	0x8e0000	0x000111
Audio Manager	Sampling_Frequency	48 kHz	0x880014	0x000000
Audio Manager	Audio_Manager _Control	Kickstart Application	0x880000	0x001011

Table 21. Enabling Pro Logic decode of 48 kHz PCM

4.5 Generalized Tone Control Manager for Dolby Digital, DTS, MPEG Multichannel and C.O.S.

Write Opcode = 0x9A; Read Opcode = 0x1B; Read Response Opcode = 0x9B

Write = 0x9A00HH 0xhhhhhh

Read Request = 0x1B00HH; Read Response = 0x9B00HH 0xhhhhhh

0xHH = index, 0xhhhhhh = data value

Index	Variable	Dataword Content
0x00	EQ_TONE_CONTROL	**Bit 4** = Reserved. **Bit 0** = Tone_Control_Enable = 0/1 = Disable/Enable Tone Control (Bass/Treble) just before Bass Manager on L, C, and R. *Default = 0x000000*
0x01	PRE_TONE_ATTENUATION	-1.0 to 1.0 = Pre-Tone Control attenuation scale factor to setup desired headroom for Tone control. *Default* = 0.03125 (0x040000)* [†]
0x02	L_PASS_TROUGH_LEVEL	-4.0 to 4.0 in 6.18 format = L Channel pass through level setting.[1] *Default* = 0.0* [†]
0x03	L_BASS_LPF_LEVEL	-4.0 to 4.0 in 6.18 format = L Channel bass LPF level setting.[1] *Default* = 0.0* [†]
0x04	L_BASS_HPF_LEVEL	-4.0 to 4.0 in 6.18 format = L Channel bass HPF level setting.[1] *Default* = 0.0* [†]
0x05	L_TREBLE_HPF_LEVEL	-4.0 to 4.0 in 6.18 format = L Channel treble HPF level setting.[1] *Default* = 0.0* [†]
0x06	L_TREBLE_LPF_LEVEL	-4.0 to 4.0 in 6.18 format = L Channel treble LPF level setting.[1] *Default* = 0.0* [†]
0x07	C_PASS_TROUGH_LEVEL	-4.0 to 4.0 in 6.18 format = C Channel pass through level setting.[1] *Default* = 0.0* [†]
0x08	C_BASS_LPF_LEVEL	-4.0 to 4.0 in 6.18 format = C Channel bass LPF level setting.[1] *Default* = 0.0* [†]
0x09	C_BASS_HPF_LEVEL	-4.0 to 4.0 in 6.18 format = C Channel bass HPF level setting.[1] *Default* = 0.0* [†]
0x0a	C_TREBLE_HPF_LEVEL	-4.0 to 4.0 in 6.18 format = C Channel treble HPF level setting.[1] *Default* = 0.0* [†]
0x0b	C_TREBLE_LPF_LEVEL	-4.0 to 4.0 in 6.18 format = C Channel treble LPF level setting.[1] *Default* = 0.0* [†]
0x0c	R_PASS_TROUGH_LEVEL	-4.0 to 4.0 in 6.18 format = R Channel pass through level setting.[1] *Default* = 0.0* [†]

Table 22. Tone Control

Index	Variable	Dataword Content
0x0d	R_BASS_LPF_LEVEL	-4.0 to 4.0 in 6.18 format = R Channel bass LPF level setting.[1] *Default* = 0.0* [†]
0x0e	R_BASS_HPF_LEVEL	-4.0 to 4.0 in 6.18 format = R Channel bass HPF level setting.[1] *Default* = 0.0* [†]
0x0f	R_TREBLE_HPF_LEVEL	-4.0 to 4.0 in 6.18 format = R Channel treble HPF level setting.[1] *Default* = 0.0* [†]
0x10	R_TREBLE_LPF_LEVEL	-4.0 to 4.0 in 6.18 format = R Channel treble LPF level setting.[1] *Default* = 0.0* [†]
0x1b	BASS_LPF_B0	-1.0 to 1.0 in 1.23 format = I order b0 filter coefficient for LPF (default corner=90 Hz at Fs=48 KHz). *Default* = 0x00bfe4* **Note:** b1 = b0 assumed for LPF.
0x1c	BASS_LPF_A1	-1.0 to 1.0 in 1.23 format = I order a1 filter coefficient for LPF (default corner=90 Hz at Fs=48 KHz). *Default* = 0x7e8037*
0x1d	BASS_HPF_B0	-1.0 to 1.0 in 1.23 format = I order b0 filter coefficient for HPF (default corner=350 Hz at Fs=48 KHz). *Default* = 0x7d220d* **Note:** b1 = - b0 assumed for HPF.
0x1e	BASS_HPF_A1	-1.0 to 1.0 in 1.23 format = I order a1 filter coefficient for HPF (default corner=350 Hz at Fs=48 KHz). *Default* = 0x7a441a*
0x1f	TREBLE_HPF_B0	-1.0 to 1.0 in 1.23 format = I order b0 filter coefficient for HPF (default corner=7500 Hz at Fs=48 KHz). *Default* = 0x536a08* **Note:** b1 = - b0 assumed for HPF.
0x20	TREBLE_HPF_A1	-1.0 to 1.0 in 1.23 format = I order a1 filter coefficient for HPF (default corner=7500 Hz at Fs=48 KHz). *Default* = 0x26d410*
0x21	TREBLE_LPF_B0	-1.0 to 1.0 in 1.23 format = I order b0 filter coefficient for LPF (default corner=1900 Hz at Fs=48 KHz). *Default* = 0x0e38e2* **Note:** b1 = b0 assumed for LPF.
0x22	TREBLE_LPF_A1	-1.0 to 1.0 in 1.23 format = I order a1 filter coefficient for LPF (default corner=1900 Hz at Fs=48 KHz). *Default* = 0x638e3c*

Table 22. Tone Control (Continued)

[1]Please see Section 4.6.1 "Level Settings for +12 dB to -12 dB Tone Control" on page 65 on details of how to set up this register

Notes: "Default" vs. "Default*" See definition page 32.

The tone control application code enhancement allows the host to send commands to the DSP that allows a parametric boost of up to +12 dB to treble and/or bass frequencies in addition to a parametric cut of up to -12 dB to treble and/or bass frequencies on the front three channels (L, C, R). The only application codes covered by AN163 that have Tone Control ability are: ac3_32x8.ld, dts_32x8.ld, mpg_32x8.ld, and eff_32x7.ld (or later releases). Other application codes not covered specifically in AN163, such as SRS Circle Surround (cs__3xx3.ld), also support tone control. Please refer to the appropriate AN163 appendix for more information.

Note: For system certification purposes (such as Dolby Laboratories), tone control should be disabled during certification testing.

4.6 Controlling the Level for Treble and Bass Boost/Cut

When tone control is enabled, the pre-Bass Manager channel PCM is attenuated by the *Pre_Tone_Attenuation* scale factor and passed through tone control shelving filters for bass and treble.

Note: *Pre_Tone_Attenuation* affects only channels which have tone control (L, C, and R) and leaves the other channels (Ls, Rs, LFE) untouched at the full level. Thus, in order to equalize all the levels coming out of the CS4932X, the controller should set the volumes of the other channels (Ls, Rs, LFE) to be the same as *Pre_Tone_Attenuation*.
Pre_Tone_Attenuation and the above matching volume setting will result in reduced level which has to be compensated by analog gain after the digital-to-analog conversion in the analog chain

The controller has the knowledge of this attenuation setting at all times and can thus adjust the post-DAC analog gain accordingly. However, this increased digital headroom will result in smaller dynamic range numbers and thus it is recommended that a passthrough (no Tone Control) mode be implemented for testing purposes.

For each channel (L, C, R), the tone control module accepts 5 independent parameters *Pass_Through_Level, Bass_LPF_Level, Bass_HPF_Level, Treble_LPF_Level,* and *Treble_HPF_Level to* adjust the amount of cut/boost in the low and high frequency bands for that channel. The following section describes the procedure by which appropriate values for these parameters can be arrived at for a desired tone control setting.

4.6.1 Level Settings for +12 dB to -12 dB Tone Control

The desired bass and treble values are mapped to five values which are plugged into a particular channel's X_Pass_Through_Level, X_Bass_LPF_Level, X_Bass_HPF_Level, X_Treble_HPF_Leve,l and X_Treble_LPF_Level registers, where X denotes L, C, R, Ls, or Rs.

The procedure for mapping the desired bass and treble values is given below. These variables can be changed on the fly without audible artifacts, if the step is smaller than 1 dB. Obviously, a one-step change from -12 dB to +12 dB would result in an audible artifact. Please note that the procedure assumes integer values for the tone levels. Also, the abbreviation LUT is used to denote a "Look-Up Table". Table 23, Table 24 and Table 25 are the three Look-Up Tables used in the mapping scheme and are labeled LUT0, LUT1, and LUT2. The mapping is sensitive to the "polarity" of the tone levels. The four possible cases are handled differently as follows:

- Bass Level > = 0 and Treble > = 0:

  X_Pass_Through_Level = 0x03FFFF;
  X_Bass_LPF_Level = LUT1(Bass_Level);
  X_Bass_HPF_Level = 0;
  X_Treble_HPF_Level = LUT1(Treble_Level);
  X_Treble_LPF_Level = 0;

- Bass Level > = 0 and Treble < 0:

  X_Pass_Through_Level = LUT0(abs (Treble_Level));
  X_Bass_LPF_Level = LUT1((Bass_Level));

X_Bass_HPF_Level = 0;

X_Treble_HPF_Level = 0;

X_Treble_LPF_Level = LUT2(abs (Treble_ Level));

- Bass Level < 0 and Treble > = 0:

    X_Pass_Through_Level = LUT0(abs (Bass_ Level));

    X_Bass_LPF_Level = 0;

    X_Bass_HPF_Level = LUT2(abs (Bass_ Level));

    X_Treble_HPF_Level = LUT1(Treble_Level);

    X_Treble_LPF_Level = 0;

- Bass Level < 0 and Treble < 0:

    X_Pass_Through_Level = LUT0(abs (Bass_ Level)) - LUT2(abs (Treble_Level)) + CF;

    X_Bass_LPF_Level = -CF;

X_Bass_HPF_Level = LUT2(abs (Bass_ Level));

X_Treble_HPF_Level = -CF;

X_Treble_LPF_Level = LUT2(abs (Treble_ Level));

4.6.2 Tone Control Look-Up Tables (LUT0, LUT1, LUT2)

The LUT0, LUT1, and LUT2 look-up tables have the following characteristics: The bass and treble setting values range from -12 to +12 dB. In order to accommodate this range in 24-bit signed, fixed-point binary format, the Hex values are all represented in 6.18 format. The tables can be stored in the microcontroller and used with the algorithm described in the previous section to generate the various filter settings for a given user setting.

dB	Real Value	Hex Value in 6.18
0	1	40000
1	0.891240713	390A1
2	0.794310008	32D60
3	0.707921418	2D4E9
4	0.630928389	28612
5	0.562309067	23FCE
6	0.501152734	2012E
7	0.44664772	1C95E
8	0.398070632	197A0
9	0.354776754	16B4B
10	0.316191487	143C8
11	0.281802726	12091
12	0.251154063	1012F

Table 23. Look-Up Table 0; "LUT0"

dB	Real Value	Hex Value in 6.18
0	0	0
1	0.122031327	7CF6
2	0.258954299	1092B
3	0.412586164	1A67D
4	0.584965928	25701
5	0.778381424	31D10
6	0.99539967	3FB4A
7	1.23890094	4F4A2
8	1.512116993	60C68
9	1.818673965	74652
10	2.16264049	8A68B
11	2.548581706	A31BF
12	2.981619842	BED2E

Table 24. Look-Up Table 1; "LUT1"

dB	Real Value	Hex Value in 6.18
0	0	0
1	0.108759287	6F5F
2	0.205689992	D2A0
3	0.292078582	12B17
4	0.369071611	179EE
5	0.437690933	1C032
6	0.498847266	1FED2
7	0.55335228	236A2
8	0.601929368	26860
9	0.645223246	294B5
10	0.683808513	2BC38
11	0.718197274	2DF6F
12	0.748845937	2FED1

Table 25. Look-Up Table 2; "LUT2"

4.7 Controlling the Corner Frequencies of the LPF/HPF for Bass and Treble Control

The tone control module implements shelving filters using two first order LPFs and two first order HPFs. The LPFs default corner frequencies are 90 Hz and 1900 Hz at Fs = 48 KHz. The HPFs default frequencies are 350 Hz and 7500 Hz at Fs = 48 KHz.

Different filter coefficients can be downloaded (before Kickstart only) for any desired corner frequencies or to match any change in Fs.

Only b0 and a1 are required to be downloaded since it is assumed that b1 = b0 for the LPF, and b1 = -b0 for the HPF. These values can be obtained using any standard filter design procedure for first order LPF and HPF with unity gain in the passband. The coefficients are all expected to be downloaded in 1.23 format before Kickstart.

Note: LPF/HPF coefficients should be set up before Kickstart and should not be modified on the fly during runtime.
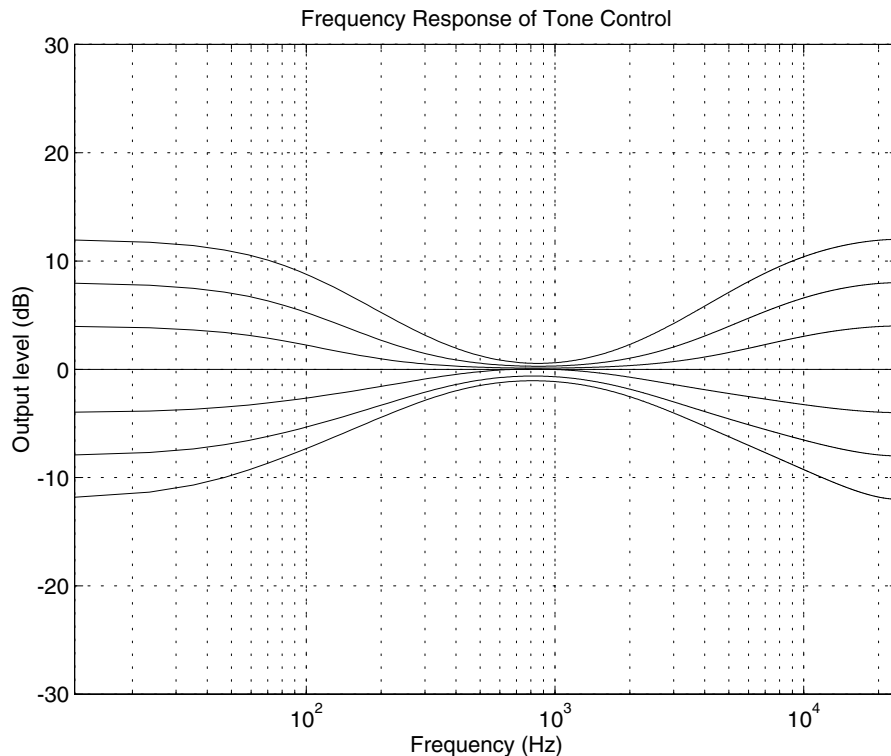
Figure 16. Composite Graph of a Frequency Sweep with Bass and Treble

Note: Boosted by +4 dB, +8 db, and +12 dB along with Bass and Treble Cut by -4 dB, -8 dB, and -12 dB

Frequency Response of Tone Control
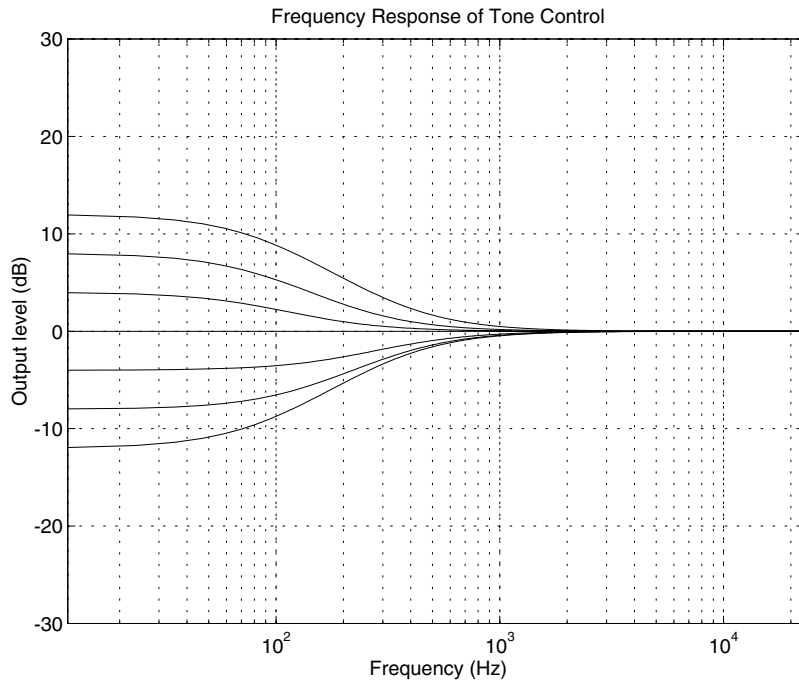
Figure 17. Composite Graph of a Frequency Sweep with Bass

Note: Boosted by +4 dB, +8 db, and +12 dB along with Bass Cut by -4 dB, -8 dB, and -12 dB

Frequency Response of Tone Control
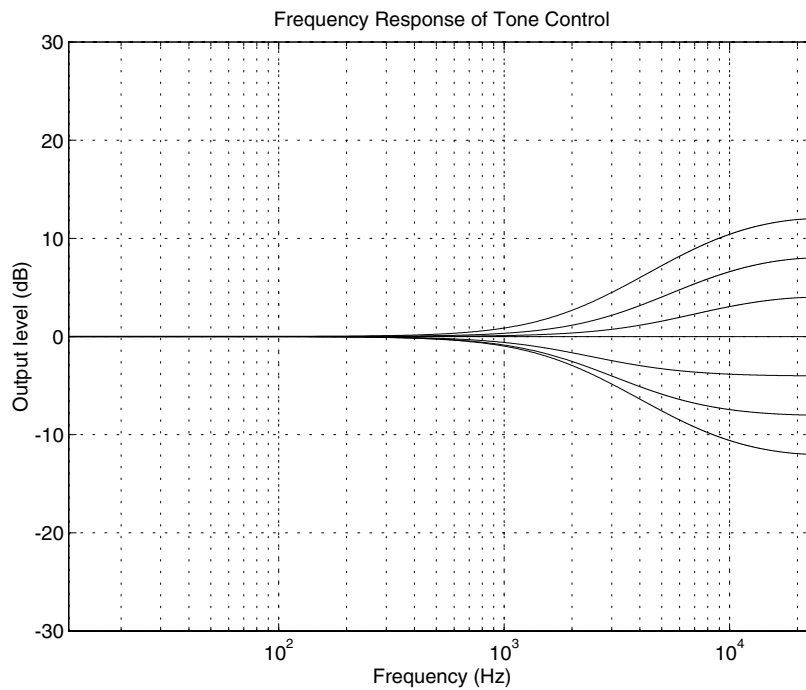
Figure 18. Composite Graph of a Frequency Sweep with Treble

Note: Boosted by +4 dB, +8 db, and +12 dB along with Treble Cut by -4 dB, -8 dB, and -12 dB

5. AC-3 SPECIFIC MODULES AND OPERATIONS

The following two modules are specific to AC-3 code. They are the AC-3 Manager and PCM Manager. In addition to AC-3 specific modules, certain operations such as enabling Autodetection of an AC-3 stream and enabling Autodetection of a Pro Logic encoded AC-3 stream is also included in this section. Please refer to Section 4.2 "Generalized Bass Manager for Dolby Digital, DTS, MP3, MPEG Multichannel, AAC, and Crystal Original Surround" on page 40 section for the listing of the Dolby Bass Manager configurations. AC-3 application code (ac3_32xx.ld) supports the following types of data: AC-3 (Pc = 1). Pass-Through of PCM is also possible using this code.

5.1 AC-3 Manager

Write Opcode = 0x8A; Read Opcode = 0x0B; Read Response Opcode = 0x8B

Write = 0x8A00HH 0xhhhhhh

Read Request = 0x0B00HH;

Read Response = 0x8B00HH 0xhhhhhh

0xHH = index, 0xhhhhhh = data value

Index	Variable	Dataword Content
0x00	AC3_CONTROL	**Bit 4** = LFE_Enable = 0/1 = Disable/Enable LFE channel decoding (if it exists in the stream). † **Bit 0** = AC3_Enable = 0/1 = Disable/Enable AC-3 decoding. † *Default = 0x000000*
0x01	OUTPUT_MODE_CONTROL	**Bit 8** = Karaoke_Capable_Enable = 0/1 = Enable/Disable Karaoke capable downmixing (see Section 5.2 "PCM Manager (for AC-3 Application Code)" on page 78 for Karaoke Capable user settings). † **Bits 5:4** = Dualmode = 0...3 = -u setting of Dolby C decoder. Valid only if acmod=0 (dual mono). † 00b = Stereo. 01b = Left Mono. 10b = Right Mono. 11b = Mixed-Mono. **Bits 3:0** = Output_Mode = 0...7 = -o setting of Dolby C decoder. † 0x0 = 2/0 Lt, Rt Dolby Surround compatible. (Equation in Section 5.1.2 "Downmix Equation for Lt, Rt" on page 75). 0x1 = 1/0 C. 0x2 = 2/0 L, R. 0x3 = 3/0 L, C, R. 0x4 = 2/1 L, R, S (Ls = Rs = S - 3 dB). 0x5 = 3/1 L, C, R, S (Ls = Rs = S - 3 dB). 0x6 = 2/2 L, R, Ls, Rs. 0x7 = 3/2 L, C, R, Ls, Rs. *Default* = 0x000007* **Note:** Undesired Channel outputs must be explicitly muted by setting the corresponding Ch_Vol (see Section 4.1 "Audio Manager for Dolby Digital, DTS, MP3, MPEG Multichannel, AAC and Crystal Original Surround" on page 33) to 0.0

Table 26. AC-3 Manager

Index	Variable	Dataword Content
0x02	COMPRESSION_CONTROL	**Bits 1:0** = 0...3 = -k setting of Dolby C decoder. † 00b = Custom Mode 0 (Analog Dialnorm). 01b = Custom Mode 1 (Digital Dialnorm). 10b = Line Out Mode (This mode is **REQUIRED** for all AVR systems). 11b = RF Remodulation Mode. *Default* = 1*
0x03	CUT_X	0x000000-0x7FFFFF = Compression cut scale factor = -x setting of Dolby C decoder. Cut_x is a scaling factor used on the absolute compression limits. The absolute compression limits are set during the encoding process. † *Default* = 0*
0x04	BOOST_Y	0x000000-0x7FFFFF = Compression boost scale factor = -y setting of Dolby C decoder. Boost_y is a scaling factor used on the absolute compression limits. The absolute compression limits are set during the encoding process. † *Default* = 0*
0x05	FSCOD	**Bits 1:0** = Current Sample Rate Code (fscod) value. 00b = 48kHz. 01b = 44.1kHz. 10b = 32kHz. 11b = Reserved. ***READ-ONLY***
0x06	BSMOD	**Bits 3:0** = Current Bit Stream (bsmod) value. 0x0 = Main audio service: complete main (CM). 0x1 = Main audio service: music and effects (ME). 0x2 = Associated service: visually impaired (VI). 0x3 = Associated service: hearing impaired (HI). 0x4 = Associated service: dialogue (D). 0x5 = Associated service: commentary (C). 0x6 = Associated service: emergency (E). 0x7: if ACMOD == 1: Associated service: voice over (VO). if ACMOD == 010b to 111b: main audio service: karaoke. ***READ-ONLY***

Table 26. AC-3 Manager (Continued)

Index	Variable	Dataword Content
0x07	ACMOD	**Bits 3:0** = Current Audio Coding Mode (acmod) value 0x0 = 1+1 Ch1, Ch2. 0x1 = 1/0 C. 0x2 = 2/0 L, R. 0x3 = 3/0 L, C, R. 0x4 = 2/1 L, R, S. 0x5 = 3/1 L, C, R, S. 0x6 = 2/2 L, R, Ls, Rs. 0x7 = 3/2 L, C, R, Ls, Rs. **Note:** Changes in this stream variable will result in an Unsolicited Message from the DSP if Audio_Config_Change_Notification has been enabled in the Audio Manager. Please refer to Section 4.1.1 "Audio_Control: (Index 0x00)" on page 37 for more information. *READ-ONLY*
0x08	DSURMOD	**Bits 1:0** = Current Dolby Surround Mode (dsurmod value) valid only when (acmod=2). 00b = Not indicated. 01b = NOT Dolby Surround encoded. 10b = Dolby Surround Encoded. 11b = Reserved. **Note:** Changes in this stream variable will result in an Unsolicited Message from the DSP if Audio_Config_Change_Notification has been enabled in the Audio Manager. Please refer to Section 4.1.1 "Audio_Control: (Index 0x00)" on page 37 for more information. *READ-ONLY*
0x09	CLEV	0x000000-0x7FFFFF = Current center mixing level (interpreted stream value as 1.23 fractional integer). Valid only if 3 front channels exist. *READ-ONLY*
0x0a	SLEV	0x000000-0x7FFFFF = Current surround mixing level (interpreted stream value as 1.23 fractional integer). Valid only if a surround channel exists. *READ-ONLY*
0x0b	LFEON	**Bit 0** = Current lfeon value. 0 = LFE channel is off. 1 = LFE channel is on. Notes: Changes in this stream variable will result in an Unsolicited Message from the DSP if Audio_Config_Change_Notification has been enabled in the Audio Manager. Please refer to Section 4.1.1 "Audio_Control: (Index 0x00)" on page 37 for more information. *READ-ONLY*

Table 26. AC-3 Manager (Continued)

Index	Variable	Dataword Content
0x0c	DIALNORM	0...31 = Current dialnorm value. **Note:** Changes in this stream variable will result in an Unsolicited Message from the DSP if Audio_Config_Change_Notification has been enabled in the Audio Manager. Please refer to Section 4.1.1 "Audio_Control: (Index 0x00)" on page 37 for more information. *READ-ONLY*
0x0d	COMPR	0xFFFFFF = Nonexistent in stream. 0x00-0xFF = Current compr value. *READ-ONLY*
0x0e	DIALNORM2	0...31 = Current dialnorm2 value. Valid only in 1+1 dual mono mode (acmod=0). *READ-ONLY*
0x0f	COMPR2	0xFFFFFF = Nonexistent in stream. 0x00-0xFF = Current compr2 value. Valid only in 1+1 dual mono mode (acmod=0). *READ-ONLY*
0x10	BSID	0...31 = Current bsid value. *READ-ONLY*
0x11	LANGCOD	0xFFFFFF = Nonexistent in stream. 0...255 = Current langcod value. *READ-ONLY*
0x12	MIXLEVEL_ROOMTYP	0xFFFFFF = Nonexistent in stream. 0x00...0x7F: **Bits 6:2** = Current mixlevel. **Bits 1:0** = Current roomtyp. *READ-ONLY*
0x13	LANGCOD2	0xFFFFFF = Nonexistent in stream. 0...255 = Current langcod value. Valid only in 1+1 dual mono mode (acmod=0). *READ-ONLY*
0x14	MIXLEVEL2_ROOMTYP2	0xFFFFFF = Nonexistent in stream. 0x00...0x7F: **Bits 6:2** = Current mixlevel2. **Bits 1:0** = Current roomtyp2. Valid only in 1+1 dual mono mode (acmod=0). *READ-ONLY*
0x15	COPYRIGHTB	0/1 = Current copyrightb value. *READ-ONLY*
0x16	ORIGBS	0/1 = Current origbs value. *READ-ONLY*
0x17	FRMSIZCOD	Current frame size code. *READ-ONLY*
0x18	TIMECOD	Current time code. *READ-ONLY*

Table 26. AC-3 Manager (Continued)

Index	Variable	Dataword Content
0x19	TIMECOD2	Current time code 2. This variable can also indicate that the AC-3 stream is Dolby Digital Surround-EX encoded. The host must also take into account the BSID variable setting in order to confirm how a change in TIMECOD2 should be interpreted. Please refer to the *AC-3 Specification* for more details.
		Note: Changes in this stream variable will result in an Unsolicited Message from the DSP if Audio_Config_Change_Notification has been enabled in the Audio Manager. Please refer to Section 4.1.1 "Audio_Control: (Index 0x00)" on page 37 for more information.
		READ-ONLY

Table 26. AC-3 Manager (Continued)

Notes: † See definition on page 32.

"Default" vs. "Default*" See definition page 32.

5.1.1 OUTPUT_MODE_CONTROL

Sets up particular output speaker downmix configuration. When configuring the Output _Mode_Control, first the individual volumes of each undesired channel should be set to 0. Next, the desired output mode should be specified. This should typically be done before kickstart of the application. For example, if output mode 3/0 (L, C, R) is chosen, Ls and Rs volume must be explicitly set to 0 in the Audio Manager module to guarantee no output on those channels.

Bit 8: KARAOKE_CAPABLE_ENABLE - If set, downmixing of Karaoke streams (bsmod=7, acmod = 2...7) will be performed according to Karaoke Capable decoder specification by employing user settings. See Section 5.2 "PCM Manager (for AC-3 Application Code)" on page 78 for more details of the user settings.

If Karaoke_Capable_Enable is not set, then the CS4932X defaults to a Karaoke Aware downmix using levels indicated in the stream.

Bits 5:4: DUALMODE = 0...3 = -u setting of Dolby C decoder. Valid only if ACMOD = 0 (dual mono).

 00b = Stereo.
 01b = Left Mono.
 10b = Right Mono.
 11b = Mixed-Mono.

Bits 3:0: OUTPUT_MODE = 0...7 = -o setting of Dolby C decoder

 0x0 = 2/0 L, R Dolby Surround compatible (a.k.a. Lt, Rt).
 0x1 = 1/0 C.
 0x2 = 2/0 L, R.
 0x3 = 3/0 L, C, R.
 0x4 = 2/1 L, R, S (Ls = Rs = S - 3 dB).
 0x5 = 3/1 L, C, R, S (Ls = Rs = S - 3 dB).
 0x6 = 2/2 L, R, Ls, Rs.
 0x7 = 3/2 L, C, R, Ls, Rs.

5.1.2 Downmix Equation for Lt, Rt

$Lt = (((1/3.1)L) + ((0.7/3.1)C) - ((0.7/3.1)Ls) - ((0.7/3.1)Rs)$

$Rt = (((1/3.1)L) + ((0.7/3.1)C) + ((0.7/3.1)Ls) + ((0.7/3.1)Rs)$

5.1.3 AC-3 Stream Information

Table 26 lists a complete set of READ-ONLY variables that are reported to the host by the AC-3 Decoder module. These values can be read out using the appropriate Read Request, Read Response sessions.

For example, to read the ACMOD value, host should issue the 3-byte Read Request 0x0b0007 and then read the 6-byte Read Response 0x9b0007, 0x00000h, where the dataword (latter 3 bytes)

indicates the reported value of ACMOD. Please refer to the *Dolby AC-3 Specification* for more information on these stream variables.

5.1.4 Special Considerations For AC-3

When using the AC-3 code, note that AC-3 and PCM are completely independent applications, and both are embedded inside the same code download. The host must treat them as such and only enable either AC-3 or PCM at any one time. Transitions from AC-3 to PCM, or vice-versa, should be done with a Soft Reset followed by complete Hardware Configuration and Application Configuration. Conversely, an Application Restart message could be used that would only require that a new input port configuration and application configuration be sent — output clocks would not be affected. For more information on Application Restart please see Section 4.1 "Audio Manager for Dolby Digital, DTS, MP3, MPEG Multichannel, AAC and Crystal Original Surround" on page 33.

5.1.5 AC-3 decode with Autodetect

In this mode, the input should be compressed AC-3. The data will be taken from the compressed input port as designated by the Hardware Configuration message. If the input on this port is not AC-3, the Autodetect function will notify the host as described in Section 4.3.1 "Autodetect Operation" on page 55.

Figure 19 is an example of pseudocode to configure the part for AC-3 decoding with autodetection. This pseudocode can be used as a template for other configurations by swapping out the message array and the size portions of the pseudocode.

Module	Index	Description	Opcode & Index	Value
AC-3	AC3_CONTROL	AC-3 Enabled LFE Enabled	0x8A0000	0x000011
AC-3	COMPRESSION_CONTROL	Set to Line Out Mode	0x8A0002	0x000002
Audio Manager	AUDIO_MANAGER _CONTROL	Autodetect Enabled PLL Enabled Kickstart Application	0x880000	0x001011

Table 27. Enabling AC-3 Decode with Autodetect

```
void AC-3_AD_Config()
{
    unsigned char AC-3_AD_config_message[] =
                                    {0x8A, 0x00, 0x00,
                                     0x00, 0x00, 0x11,

                                     0x8A, 0x00, 0x02
                                     0x00, 0x00, 0x02

                                     0x88, 0x00, 0x00,
                                     0x00, 0x10, 0x11};

    Write_*(AC-3_AD_config_message,12); /* Replace * with I2C or SPI depending on protocol
*/
}
```

Figure 19. AC-3 Decode with Autodetect

5.1.6 AC-3 with Pro Logic Decode and Autodetect

In this mode, the input should be two-channel compressed AC-3 (acmod = 2) with Pro Logic encoded data (dsurmod = 2). The data will be taken from the compressed input port as designated by the Hardware Configuration message. If the input on this port is not AC-3 the Autodetect function will notify the host as described in Section 4.3.1 "Autodetect Operation" on page 55.

Module	Index	Description	Opcode & Index	Value
Pro Logic	ProLogic Control	Enabled Autobalance Enabled B Type NR Enabled	0x8e0000	0x000111
AC-3	AC-3 Control	AC-3 Enabled LFE Enabled	0x8A0000	0x000011
AC-3	Compression_Control	Set to Line Out Mode	0x8A0002	0x000002
Audio Manager	Audio Manager Control	Autodetect Enabled PLL Enabled Kickstart Application	0x880000	0x001011

Table 28. Enabling AC-3 with Pro Logic Decode with Autodetect

5.2 PCM Manager (for AC-3 Application Code)

Write Opcode = 0x9C; Read Opcode = 0x1D; Read Response Opcode = 0x9D

Write = 0x9C00HH 0xhhhhhh

Read Request = 0x1D00HH;

Read Response = 0x9D00HH 0xhhhhhh

0xHH = index, 0xhhhhhh = data value

Index	Variable	Dataword Content
0x00	PCM_CONTROL	**Bit 4** = Deemphasis_Enable = 0/1 = Disable/Enable PCM Deemphasis. † **Note:** Please refer to the deemphasis section, located after this table, for more information about using digital deemphasis for PCM sampling frequencies other than 44.1 kHz when using IBA-based codes. **Bit 0** = PCM_Enable = 0/1 = Disable/Enable PCM. This bit should only be enabled if AC-3 is disabled. *Default = 0x000000*
0x01	K_CPL_MLEV_1_F	M level for Karaoke capable 1 front output channel. 0 <= K_Cpl_mlev_1_F <= 0x7FFFFF. † *Default = 0X1E2A53*
0x02	K_CPL_V1LEV_1_F	V1 level for Karaoke capable 1 front output channel. 0 <= K_Cpl_v1lev_1_F <= 0x7FFFFF. † *Default = 0X1E2A53*
0x03	K_CPL_V2LEV_1_F	V2 level for Karaoke capable 1 front output channel. 0 <= K_Cpl_v2lev_1_F <= 0x7FFFFF. † *Default = 0X1E2A53*
0x04	K_CPL_MLLEV_2_F	m L level for Karaoke capable 2 front output channels. 0 <= K_Cpl_mllev_2_F <= 0x7FFFFF. † *Default = 0X34B4B3*
0x05	K_CPL_V1LLEV_2_F	V1 L level for Karaoke capable 2 front output channels. 0 <= K_Cpl_v1llev_2_F <= 0x7FFFFF. † *Default = 0X4B4B4A*
0x06	K_CPL_V2LLEV_2_F	V2 L level for Karaoke capable 2 front output channels. 0 <= K_Cpl_v2llev_2_F <= 0x7FFFFF. † *Default = 0x0*
0x07	K_CPL_MRLEV_2_F	M R level for Karaoke capable 2 front output channels. 0 <= K_Cpl_mrlev_2_F <= 0x7FFFFF. † *Default = 0X34B4B3*

Table 29. PCM Processor

Index	Variable	Dataword Content
0x08	K_CPL_V1RLEV_2_F	V1 R level for Karaoke capable 2 front output channels. 0 <= K_Cpl_v1rlev_2_F <= 0x7FFFFF. † *Default = 0x0*
0x09	K_CPL_V2RLEV_2_F	V2 R level for Karaoke capable 2 front output channels. 0 <= K_Cpl_v2rlev_2_F <= 0x7FFFFF. † *Default = 0X4B4B4A*
0x0a	K_CPL_MCPAN_3_F	M C level for Karaoke capable 3 front output channels. 0 <= K_Cpl_mcpan_3_F <= 0x7FFFFF. † *Default = 0x7FFFFF.*
0x0b	K_CPL_V1CPAN_3_F	V1 C level for Karaoke capable 3 front output channels. 0 <= K_Cpl_v1cpan_3_F <= 0x7FFFFF. † *Default = 0x0.*
0x0c	K_CPL_V2CPAN_3_F	V2 C level for Karaoke capable 3 front output channels. 0 <= K_Cpl_v2cpan_3_F <= 0x7FFFFF. † *Default = 0x0.*
0x0d	K_CPL_MLPAN_3_F	M L level for Karaoke capable 3 front output channels. 0 <= K_Cpl_mlpan_3_F <= 0x7FFFFF. † *Default = 0x0.*
0x0e	K_CPL_V1LPAN_3_F	V1 L level for Karaoke capable 3 front output channels. 0 <= K_Cpl_v1lpan_3_F <= 0x7FFFFF. † *Default = 0x7FFFFF.*
0x0f	K_CPL_V2LPAN_3_F	V2 L level for Karaoke capable 3 front output channels. 0 <= K_Cpl_v2lpan_3_F <= 0x7FFFFF. † *Default = 0x0.*
0x10	K_CPL_MRPAN_3_F	M R level for Karaoke capable 3 front output channels. 0 <= K_Cpl_mrpan_3_F <= 0x7FFFFF. † *Default = 0x0.*
0x11	K_CPL_V1RPAN_3_F	V1 R level for Karaoke capable 3 front output channels. 0 <= K_Cpl_v1rpan_3_F <= 0x7FFFFF. † *Default = 0x0.*
0x12	K_CPL_V2RPAN_3_F	V2 R level for Karaoke capable 3 front output channels. 0 <= K_Cpl_v2rpan_3_F <= 0x7FFFFF. † *Default = 0x7FFFFF.*

Table 29. PCM Processor

Notes: † See definition on page 32.

"Default" vs. "Default*" See definition page 32.

5.2.1 PCM_CONTROL

Bit 4: *DEEMPHASIS_ENABLE*

Disable/Enable PCM Deemphasis. The filter used is identical to CS4334 (50/15 uS).

    1 - Enables Deemphasis
    0 - Disables Deemphasis

For systems that plan on using a "Non-IBA-Based" PCM Processing application code such as ac3_3265.ld (that has been configured for PCM Pass-Through) and want to support digital deemphasis for the sampling frequencies of 32 kHz and 48 kHz, as well as 44.1 kHz, PCM, the host only needs to inform the DSP the setting for the current PCM sampling frequency, which is set via the Audio Manager (0x14) before Kickstart.

Note: For systems that plan on using an "IBA-Based" PCM Processing application code such as ac3i326x.ld (that has been configured for PCM Pass-Through) and want to support digital de-emphasis for the sampling frequencies of 32 kHz and 48 kHz, as well as 44.1 kHz, PCM, the host, in addition to communicating the setting for the current PCM sampling frequency, which is set via the Audio Manager (0x14), must also download the following commands before Kickstart:

For 32 kHz support:

848ff9
bb82fa
848ffe
42aef3
848fff
f8d406

For 44.1 kHz support:

848ff8
aeacda
848ffc
3b6543
848ffd
f34797

For 48 kHz support:

848ff7
ab9ed7
848ffa
39be36
848ffb
f1e0a1

For 64 kHz support:

848ff9
a25a7b
848ffe
34f310
848fff
ed676b

For 88.2 kHz support:

848ff8
99f80b
848ffc
30eb8a
848ffd
e90c82

For 96 kHz support:

848ff7
981257
848ffa
300d08
848ffb
e8054f

Digital de-emphasis support does not exist for sampling frequencies other than 96 kHz, 88.2 kHz, 64 kHz, 48 kHz, 44.1 kHz, or 32 kHz regardless of what code is being downloaded.

Bit 1: *PCM_ENABLE*

Disable/Enable PCM. This bit should only be enabled if AC-3 is disabled.

    1 - Enables PCM Pass-Through
    0 - Disables PCM Pass-Through

5.2.2 Karaoke Capable Function:

Karaoke Capable functionality is enabled when Bit 8 of Output_Mode_Control in AC-3_Manager is set to 1.

If Karaoke_Capable_Enable is not set, then CS4932X defaults to a Karaoke Aware downmix, using levels indicated in the stream.

Karaoke Capable functionality is enabled when Bit 8 of Output_Mode_Control in AC-3_Manager is set to 1.

In a karaoke bit stream (bsmod=7, acmod=2...7), 3 special channels have to be downmixed according to user specification. They are M (guide melody), and V1, V2 (vocal tracks one and two).

In Karaoke Aware mode (default mode where Karaoke_Capable_Enable=0), the CS4932X

downmixes these channels according to the clev/slev values indicated in the bitstream. This Karaoke Aware downmix reproduces all the program content in the front speakers according to the programmed levels and audio configuration of the bitstream.

In Karaoke Capable mode, the downmix coefficients are set up by the user according to the number of output front channels: 1, 2, or 3. Independent settings are available for each of these 3 cases as indicated in Table 29 on page 78. The following equations define the downmixing performed for each of the 3 cases:

- 1 front channel:

  $Ck = mlev * M + v1lev * V1 + v2lev * V2.$

  $Co = 0.707 * L + Ck + 0.707 * R.$

- 2 front channels:

  $Lk = mllev * M + v1llev * V1 + v2llev * V2.$

  $Rk = mrlev * M + v1rlev * V1 + v2rlev * V2.$

  $Lo = L + Lk$

  $Ro = R + Rk$

- 3 front channels:

  $Ck = mcpan * M + v1cpan * V1 + v2cpan * V2.$

  $Lk = mlpan * M + v1lpan * V1 + v2lpan * V2.$

  $Rk = mrpan * M + v1rpan * V1 + v2rpan * V2.$

  $Lo = L + Lk$

  $Co = Ck$

  $Ro = R + Rk$

5.2.3 Special Considerations For PCM

When using the AC-3 code, it should be noted that AC-3 and PCM are completely independent applications. The host must treat them as such and only enable either AC-3 or PCM at any one time. Transitions from AC-3 to PCM or vice-versa should be done with a soft reset followed by complete Hardware Configuration and Application Configuration. Conversely, an Application Restart message could be used, which would only require that a new input port configuration and application configuration be sent — output clocks would not be affected.

5.2.4 Special Considerations For Pink/White Noise

Pink/White Noise is controlled by three parameters. These parameters are the currently configured application (i.e. PCM), the channel selected by the Noise_Control variable:, and the state of the Autodetect Enable. In order to use Pink/White Noise, the code should be configured strictly for PCM before Kickstart. If Pro Logic is enabled, the host loses independent control of Pink/White Noise on the Ls and Rs channels — the surround channels operate in unison.

White noise can be generated by additionally setting Bit 4 of Noise_Control in the messages above. In the same message, if Bit 5 is set, a 10.5 dB boosted noise is received.

5.2.5 PCM Pass-through (Using AC-3 Application Code)

In this mode, the input should be 44.1 kHz PCM data. The data will be taken from the PCM input port, as designated by the Hardware Configuration message. (See Table 30.)

Module	Index	Description	Opcode and Index	Value
PCM	PCM Control	Enabled	0x9c0000	0x000001
Audio Manager	Sampling Frequency	44.1 kHz	0x880014	0x000001
Audio Manager	Audio Manager Control	Kickstart Application	0x880000	0x001011

Table 30. Enabling PCM Pass-through of 44.1 kHz PCM

5.2.6 Pink/White Noise Generation

Module	Index	Description	Opcode and Index	Value
PCM	PCM Control	Enabled	0x9c0000	0x000001
Audio Manager	Noise Control	Pink Noise On Left Channel	0x880001	0x000000
Audio Manager	Audio Manager Control	Kickstart Application Autodetect Disabled	0x880000	0x000011

Table 31. Enabling Pink/White Noise Generation

To cycle through the channels (after sending the above messages), the host would send the following sequence of commands, pausing between each command to hold the pink noise on a particular channel (See Table 32).

Module	Index	Description	Opcode and Index	Value
Audio Manager	Noise Control	Pink Noise On Left Channel	0x880001	0x000000
Audio Manager	Noise Control	Pink Noise On Center Channel	0x880001	0x000001
Audio Manager	Noise Control	Pink Noise On Right Channel	0x880001	0x000002
Audio Manager	Noise Control	Pink Noise On Left Surround Channel	0x880001	0x000003
Audio Manager	Noise Control	Pink Noise On Right Surround Channel	0x880001	0x000004
Audio Manager	Noise Control	Pink Noise On LFE Channel	0x880001	0x000005

Table 32. Pink/White Noise Generation for All 5.1 Channel

6. DTS MANAGER AND OPERATIONS

The module described in this section is specific to DTS code. In addition to the DTS Manager, certain operations such as enabling Autodetection of a DTS stream is also included in this section. DTS application code (dts_32xx.ld) supports the following types of DTS data: DTS-1 data (512-sample bursts) (Pc = B); DTS-2 data (1024-sample burst) (Pc = C); DTS-3 data (2048-sample bursts) (Pc = D). This application code also supports DTS

CDs (DTS Format-16 elementary stream) and DTS LDs (DTS Format-14 elementary stream). Other DTS data formats (not currently available or defined) *may* be able to be decoded, but will not be able to be autodetected.

Note: For DTS certification, your system should be able to enable DTS decoding with Autodetection disabled in order to detect and decode certain streams which are only available on the DTS certification test disc.

6.1 DTS Manager

Write Opcode = 0x90; Read Opcode = 0x11; Read Response Opcode = 0x91

Write Message = 0x9000HH 0xhhhhhh

Read Request Message = 0x1100HH;

Read Response Message = 0x9100HH 0xhhhhhh

0xHH = index, 0xhhhhhh = data value

Index	Variable	Dataword Content
0x00	DTS_CONTROL	**Bit 8** = DRC_Enable = 0/1 = Disable / Enable Dynamic Range Compression (DRC) **Note:** DRC coefficients must be present in the data stream for DRC to take effect. **Bits 7:4** = Output_Mode = 0...7 † 0 = 2/0 Lt, Rt Surround Encoded. (Refer to Section 6.1.2 "Downmix Equation for Lt, Rt" on page 86 for the Equation). 1 = 1/0 C. 2 = 2/0 L, R. 3 = 3/0 L, C, R. 4 = 2/1 L, R, S (Ls = Rs = S - 3dB). 5 = 3/1 L, C, R, S (Ls = Rs = S - 3dB). 6 = 2/2 L, R, Ls, Rs. 7 = 3/2 L, C, R, Ls, Rs. **Bit 0 =** DTS_Enable = 0/1 = Disable/Enable DTS decoding. **Note:** Undesired Channel outputs should be explicitly muted by setting the corresponding Ch_Vol (see Section 4.1 "Audio Manager for Dolby Digital, DTS, MP3, MPEG Multichannel, AAC and Crystal Original Surround" on page 33) to 0.0 *Default* = 0x000070*
0x01	FTYPE	0..1 = Frame Type Identifier. ***READ-ONLY***
0x02	SHORT	0..31 = Deficit Sample Count. ***READ-ONLY***

Table 33. DTS Manager

Index	Variable	Dataword Content
0x03	CPF	0..1 = CRC Present Flag. ***READ-ONLY***
0x04	NBLKS	5..127 = Number of PCM Sample Blocks. ***READ-ONLY***
0x05	FSIZE	96..8192 = Primary Frame Byte Size. ***READ-ONLY***
0x06	AMODE	0..63 = Audio Channel Arrangement (AMODE). 0x0 = A 0x1 = A + B (dual mono) 0x2 = L + R (stereo) 0x3 = (L+R) + (L-R) (sum-difference) 0x4 = LT + RT (left and right total) 0x5 = C + L + R 0x6 = L + R + S 0x7 = C + L + R + S 0x8 = L + R + SL + SR 0x9 = C + L + R + SL + SR 0xa - 0xf = Reported, but not supported by this code. 0x10 - 0x3f = User defined. **Note:** Changes in this stream variable will result in an Unsolicited Message from the DSP if Audio_Config_Change_Notification has been enabled in the Audio Manager. Please refer to Section 4.1.1 "Audio_Control: (Index 0x00)" on page 37 for more information. ***READ-ONLY***
0x07	SFREQ	0..15 = Source Sampling Frequency. ***READ-ONLY***
0x08	RATE	0..31 = Transmission Bit Rate. ***READ-ONLY***
0x09	MIX	0..1 = Embedded Down Mix Enabled. ***READ-ONLY***
0x0a	DYNF	0..1 = Embedded Dynamic Range Flag. ***READ-ONLY***
0x0b	TIMEF	0..1 = Embedded Time Stamp Flag. ***READ-ONLY***
0x0c	AUXF	0..1 = Auxiliary Data Flag. ***READ-ONLY***
0x0d	EXT_AUDIO	0..1 = Extended Coding Flag. ***READ-ONLY***
0x0e	ASPF	0..1 = Audio Sync Word Insertion Flag. ***READ-ONLY***

Table 33. DTS Manager (Continued)

Index	Variable	Dataword Content
0x0f	LFF	0..3 = LFF Channel Interpolation factor: 0x0 = Not Present 0x1 = Present, 128 0x2 = Present, 64 0x3 = Invalid **Note:** Changes in this stream variable will result in an Unsolicited Message from the DSP if Audio_Config_Change_Notification has been enabled in the Audio Manager. Please refer to Section 4.1.1 for more information. *READ-ONLY*
0x10	HFLAG	0..1 = Predictor History Flag Switch.
0x11	FILTS	0..1 = Multirate Interpolator Switch. *READ-ONLY*
0x12	PCMR	0..7 = Source PCM coding Resolution. This variable has also been chosen as the indicator of a DTS stream being DTS-ES encoded. The DTS-ES present flag has been placed into the LSB of the PCMR field. If the PCMR variable is reported as being set to 0x3 or 0x5, then the material is DTS-ES encoded. 0x0 = 16-bits 0x1 = 16-bits (or material is DTS-ES encoded or DTS 96/24 encoded) 0x2 = 20-bits 0x3 = 20-bits (or material is DTS-ES encoded or DTS 96/24 encoded) 0x4 = Invalid 0x5 = 24-bits (or material is DTS-ES encoded or DTS 96/24 encoded) 0x6 = 24-bits 0x7 = Invalid **Note:** Changes in this stream variable will result in an Unsolicited Message from the DSP if Audio_Config_Change_Notification has been enabled in the Audio Manager. Please refer to Section 4.1.1 for more information. *READ-ONLY*
0x13	SUMF	0..1 = Front Sum / Difference Flag. *READ-ONLY*
0x14	SUMS	0..1 = Surround Sum / Difference Flag. *READ-ONLY*
0x15	SUBFS	0..15 = Number of Subframes. *READ-ONLY*
0x16	PCHS	0..7 = Number of Primary Audio Channels. *READ-ONLY*
0x17	SSC	0..3 = Subsubframe Count. *READ-ONLY*
0x18	PSC	0..7 = Partial Subsubframe Sample Count. *READ-ONLY*
0x19	RANGE	Dynamic Range Coefficient. *READ-ONLY*

Table 33. DTS Manager (Continued)

Index	Variable	Dataword Content
0x1a..0x1e	SUBS[0...4]	0..31 = Subband Activity Count. **READ-ONLY**
0x1f..0x23	VQSUB[0...4]	0..31 = High Frequency VQ Start Subband. **READ-ONLY**
0x24..0x28	JOINX[0...4]	0..7 = Joint Intensity Coding Index. **READ-ONLY**
0x29..0x32	DOWN[0...4][0...1]	0..127 = Scale Factors for downmixing. **READ-ONLY**

Table 33. DTS Manager (Continued)

Notes: "Default" vs. "Default*"

See definition page 32.

6.1.1 DTS_CONTROL

Bits 8 *DRC_ENABLE*

By enabling this bit the DSP will recognize Dynamic Range Compression *(*DRC) coefficients placed in the data stream by a DTS-certified encoder.

Bits 7:4: *OUTPUT_MODE*

These bits set up the output speaker downmix configuration. When configuring the Output _Mode, first the individual volumes of each undesired channel should be set to 0. Next, the desired output mode should be specified. This should be done before kickstart of the application. For example, if output mode 3/0 (L, C, R) is chosen, Ls and Rs volume must explicitly be set to 0 in the Audio Manager module to guarantee no output on those channels. Here is a summary of Output_Mode configuration settings.

> 0 = 2/0 Lt, Rt Surround Encoded.
> 1 = 1/0 C.
> 2 = 2/0 L, R.
> 3 = 3/0 L, C, R.
> 4 = 2/1 L, R, S (Ls = Rs = S - 3dB).
> 5 = 3/1 L, C, R, S (Ls = Rs = S - 3dB).
> 6 = 2/2 L, R, Ls, Rs.
> 7 = 3/2 L, C, R, Ls, Rs.

6.1.2 Downmix Equation for Lt, Rt

The following equation is used to produce a Lt, Rt Surround Encoded downmix.

$$Lt = (((1/3.1)L) + ((0.7/3.1)C) - ((0.7/3.1)Ls) - ((0.7/3.1)Rs)$$

$$Rt = (((1/3.1)L) + ((0.7/3.1)C) + ((0.7/3.1)Ls) + ((0.7/3.1)Rs)$$

6.1.3 DTS Stream Information

Table 33 lists out a complete set of **READ-ONLY** variables that are reported to the host by the DTS Decoder module. These can be read using the appropriate Read Request, and Read Response sessions.

For example, to read the AMODE value, host should issue the 3-byte Read Request 0x110006 and then read the 6-byte Read Response 0x910006, 0x00009h, where the dataword (latter 3 bytes) indicates the reported value of AMODE. Please refer to the *DTS Specification* for more information on these stream variables.

6.1.4 DTS Decode with Autodetect

In this mode, the input should be compressed DTS. The data will be taken from the compressed input port as designated by the Hardware Configuration message. If the input on this port is *not* DTS, the Autodetect function will notify the host as described in Section 4.3.1 "Autodetect Operation" on page 55.

Module	Index	Description	Opcode & Index	Value
DTS Manager	DTS Control	DTS Enabled (Assumes all 5.1 channels are available.)	0x900000	0x000071
Audio Manager	Audio Manager Control	Autodetect Enabled PLL Enabled Kickstart Application	0x880000	0x001011

Table 34. Enabling DTS Decode with Autodetect

Figure 20 is an example of pseudocode to configure the part DTS decode with autodetect. This pseudocode can be used as a template for other configurations by swapping out the values in the message array and the size.

```
void DTS_AD_Config()
{
   unsigned char DTS_AD_config_message[] =
                                       {0x90, 0x00, 0x00,
                                        0x00, 0x00, 0x71,

                                        0x88, 0x00, 0x00,
                                        0x00, 0x10, 0x11};

   Write_*(DTS_AD_config_message,12); /* Replace * with I2C or SPI depending on protocol */
}
```

Figure 20. DTS decode with Autodetect

6.1.5 Special Considerations For DTS CDs and Autodetection

A DTS CD is treated like a linear PCM CD by all CD/DVD players. During FF/REW ("trick" modes), the CD/DVD player drops audio samples to create an effect audible to the listener. Dropping DTS data, however, causes corruption of the bitstream which makes the data indistinguishable from linear PCM. As a result, the DTS decoder will generate an autodetect message indicating PCM was detected.

However, if a user actually changed the CD (either by hand or by CD changer), this would cause the DSP to report silence; also known as "Silent Input Data (Out of Application Sync)." If this is the case, the host must then respond to whatever the DSP reports after this detection of silence. If the disc is switched to PCM, then the host must load the corresponding PCM code. If the disc is switched to DTS, then the host must load the DTS code.

An abrupt transition from DTS to PCM should indicate to the system controller that a "trick" mode is being used, hence the PCM code should not be loaded. If the host does nothing, the DTS code will then automute during this FF/REW period and there will be no problems. Please refer to Figure 21 to see a flow chart depicting the DTS autodetection process.

However, the recommendation described above does not take into account a sudden change from DTS to PCM due to a source change. It is therefore suggested that if the user selects a source change (e.g. could change input from a DTS CD playing to a DAT playing 44.1 kHz PCM), the host will be aware of this input change and should subsequently perform a Soft Reset, Configure the Hardware, and
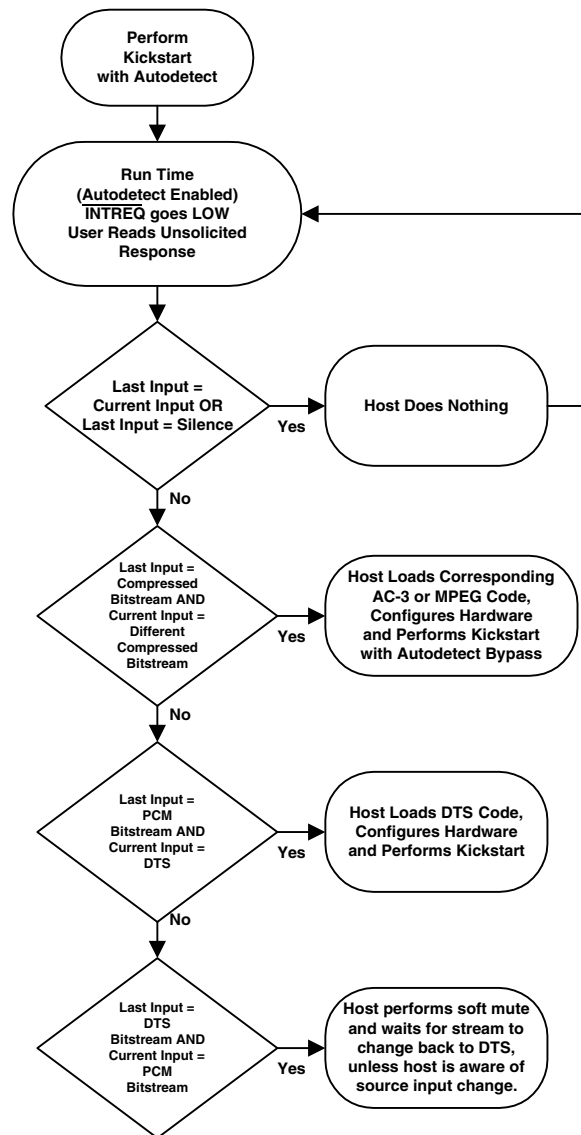
Figure 21. DTS Specific Autodetection Flowchart

perform a kickstart. This will ensure protection from such a scenario.

7. MP3 MANAGER AND OPERATIONS

The module described in this section is specific to MP3 code. In addition to the MP3 Manager, certain operations such as enabling Autodetection of a MP3 stream is also included in this section. MP3 application code (mp3_32xx.ld) only supports the following types of MPEG data: (1) MPEG-1, Layer 3 (Pc = 5), and (2) MPEG-2, Layer 3 CBR and VBR streams. It also supports the half-sampling frequencies outlined in the MPEG-2, Layer 3 specification. It does not support the "free" format where the bit rate is unspecified (bit_index = "0000"=free). The MP3 code supports IEC60958 Stereo PCM output or IEC61937 packed MP3 in an I2S, left-justified, or S/PDIF format from the AUDATA3 pin.

7.1 MP3 Manager

Write Opcode = 0x8C; Read Opcode = 0x0D; Read Response Opcode = 0x8D

Write Message = 0x8C00HH 0xhhhhhh

Read Request Message = 0x0D00HH;

Read Response Message = 0x8D00HH 0xhhhhhh

0xHH = index, 0xhhhhhh = data value

Index	Variable	Dataword Content
0x00	MP3_CONTROL	**Bits 7:4** = Output_Mode = 0...7 †
		0 = Reserved.
		1 = 1/0 C.
		2 = 2/0 L, R.
		3 = 2/0 L, R.
		4 = 2/0 L, R.
		5 = 2/0 L, R.
		6 = 2/0 L, R.
		7 = 2/0 L, R.
		Note: Undesired Channel outputs should be explicitly muted by setting the corresponding Ch_Vol (see Audio Manager). It should also be noted that unlike the other application code managers in AN163, the MP3 Manager does not have a "Decode Enable Bit." This is intentional.
		Default* = 000070
0x01	ID	0 - Extension to lower frequencies, 1 - ISO/IEC 11172-3 or ISO/IEC 13818-3. ***READ-ONLY***
0x02	LAYER	2 bits to indicate which layer is used. ***READ-ONLY***
0x03	PROTECTION_BIT	0 - stream is CRC protected, 1 - not CRC protected. ***READ-ONLY***
0x04	BITRATE_INDEX	4-bit index to a bitrate table. ***READ-ONLY***

Table 35. MP3 Manager

Index	Variable	Dataword Content
0x05	SAMPLING_FREQUENCY	2-bit index to a sampling frequency table. ***READ-ONLY***
0x06	PADDING_BIT	0 - no padding, 1 - padding exists. ***READ-ONLY***
0x07	PRIVATE_BIT	reserved for non-ISO purposes. ***READ-ONLY***
0x08	MODE	2 bits to indicate the audio configuration mode (for 2 channels only) 00 - stereo 01 - joint stereo (intensity stereo and/or ms_stereo) 10 - dual channel 11 - single channel **Note:** Changes in this stream variable will result in an Unsolicited Message from the DSP if Audio_Config_Change_Notification has been enabled in the Audio Manager. Please refer to Section 4.1.1 "Audio_Control: (Index 0x00)" on page 37 for more information. ***READ-ONLY***
0x09	MODE_EXTENSION	2 bits to indicate which sub-bands are in intensity stereo. ***READ-ONLY***
0x0a	COPYRIGHT	0 - no copyright, 1 - copyright protected bitstream. ***READ-ONLY***
0x0b	ORIGHOME	0 - bitstream is a copy, 1 - bitstream is an original. ***READ-ONLY***
0x0c	EMPHASIS	2 bits to indicate which type of de-emphasis is used. ***READ-ONLY***

Table 35. MP3 Manager (Continued)

Notes: † See definition on page 32.

"Default" vs. "Default*" See definition page 32.

7.1.1 MP3_CONTROL

Bits 7:4: *OUTPUT MODE*

These bits set up the output speaker downmix configuration. When configuring the Output_Mode, first set the individual volumes of each undesired channel to 0. Next, the desired output mode should be specified. This should be done before kickstart of the application. For example, if output mode 1/0 C is chosen, R, L volume must explicitly be set to 0 in the Audio Manager module to guarantee no output on those channels.

MPEG Stream Information: Table 35 lists a complete set of ***READ-ONLY*** variables that are reported to the host by the MP3 Decoder module. These can be read using the appropriate Read Request, and Read Response sessions.

For example, to read the MODE value, host should issue the 3-byte Read Request 0x0D0008 and then read the 6-byte Read Response 0x8D0008, 0x00000h, where the dataword (latter 3 bytes) indicates the reported value of MODE. Please refer to the *MPEG Specification ISO/IEC 11172-3* for more information on these stream variables.

7.2 MP3 Decode with Autodetect

In this mode, the input should be compressed MP3. The data will be taken from the compressed input port as designated by the Hardware Configuration

message. If the input on this port is not MP3, the Autodetect function will notify the host as described in Section 4.3.1 "Autodetect Operation" on page 55.

Figure 22 is an example of pseudocode to configure the part MP3 decode with autodetect. This pseudocode can be used as a template for other configurations by swapping out the values for the message array and the size in the pseudocode.

7.2.1 Special Considerations MP3 Autodetect

In general, if the Autodetect function senses AC-3, DTS, PCM, or MPEG data, the appropriate Unsolicited Message is sent indicating that the corresponding stream was detected. The host would then need to download the appropriate code. However, if the autodetect function senses a valid IEC61937-packed MP3 stream, it will send the host an Unsolicited Message of 0x000005 (provided the current application code was not MP3). The host must then download the MP3 application code, in order to determine if the stream is MPEG-1, Audio Layer 3 or is another type of MPEG with a reported Pc value of 5.

Note: This same Unsolicited Message of 0x000005 could also indicate that the stream was IEC61937-packed MPEG-1, Audio Layer 2 or IEC61937-packed MPEG-2 without extension. However, the only way to determine that the stream is MP3 is to download the MP3 application code.

After downloading the MP3 application code, configuring the hardware and performing a kickstart, the DSP will then generate an Unsolicited Message indicating that the stream has a Pc value of 5 (0x000005). This first Unsolicited Message does not indicate the MPEG stream type. If the input stream is MPEG-1, Audio Layer 3, the DSP will send the host an Unsolicited Message of 0x800005 and begin decoding the audio stream.

Note: This second Unsolicited Message may not occur for up to 26ms after the first Unsolicited Message.

However if the host has not seen a second Unsolicited Message of 0x80005 after 26ms from the first Unsolicited Message of 0x000005, the host then knows that the input stream is MPEG-1, Layer 2 or MPEG-2 without extension. The host must then download the MPEG-2 application code (mpg_2xxx.ld), Configure the Hardware and perform a kickstart.

Module	Index	Description	Opcode & Index	Value
MP3 Manager	MP3 Control	MP3 Enabled (Assumes 2.0 channels are available.)	0x8C0000	0x000020
Audio Manager	Audio Manager Control	Autodetect Enabled PLL Enabled Kickstart Application	0x880000	0x001011

Table 36. Enabling MP3 Decode with Autodetect

```
void MP3_AD_Config()
{
    unsigned char MP3_AD_config_message[] =
                                    {0x8C, 0x00, 0x00,
                                     0x00, 0x00, 0x20,

                                     0x88, 0x00, 0x00,
                                     0x00, 0x10, 0x11};

    Write_*(MP3_AD_config_message,12); /* Replace * with I2 C or SPI depending on protocol
*/
}
```

Figure 22. MP3 Decode with Autodetect

7.2.2 For System Designers: When to Download MP3 Application code...

The worst-case autodetection scenario is if MPEG Multichannel application code is already loaded into the DSP and the input data stream changes seemlessly (on the frame, and with no frames of silence) from an MPEG-1, Layer 2 IEC61937-packed data stream or an MPEG-2 without extension IEC61937-packed data stream to an MPEG-1, Layer 3 IEC61937-packed data stream. Detection of both stream types by the MPEG code will always report 0x000005, however if the stream is actually MPEG-1, Layer 3, the DSP will not be able to decode the stream, hence no audio will be heard.

However, we will assume that an MPEG-1, Layer 2 or MPEG-2 without extension stream will be separated by some detectable amount of silence (approximate 500ms) before being followed by an MPEG-1, Audio Layer 3 stream. With this assumption in mind, the two data streams will be separated with an entirely different type of stream, silence. If the current application code is MPEG and it detects silence, it will send an Unsolicited Message of 0x000010. The host must then download the MP3 application code, configure the hardware and perform a kickstart. Again, if the DSP detects anything other than silence, it will generate the appropriate unsolicited message which the host will interpret and then download the appropriate application code.

The inverse of the above scenario however does not present a problem though if the MP3 application code was already loaded. The host would have received an Unsolicited Message of 0x800005 when MPEG-1, Layer 3 was detected. Even if the input data stream changed seemlessly (on the frame, and with no frames of silence) from an MPEG-1, Layer 3, IEC61937-packed data stream to an MPEG-1, Layer 2, IEC61937-packed data stream or an MPEG-2 without extension IEC61937-packed data stream, the host would still report an Unsolicited Message of 0x00005. The host need only wait 26 ms and then load the MPEG-2 application code, and audio could then be heard.

Again, if another type of application code is loaded, like AC-3, and the DSP sends the host an Unsolicited Message of 0x000005, the host must download the MP3 to determine if the reported Pc value of 5 indicates that the stream is either MPEG-1, Layer 2 or MPEG-1, Layer 3 or MPEG-2 without extension. Please see Section 23 "When to Download MP3 Code..." on page 94, for an overview of the previously described summary of when to download MP3 versus MPEG Multichannel application code.
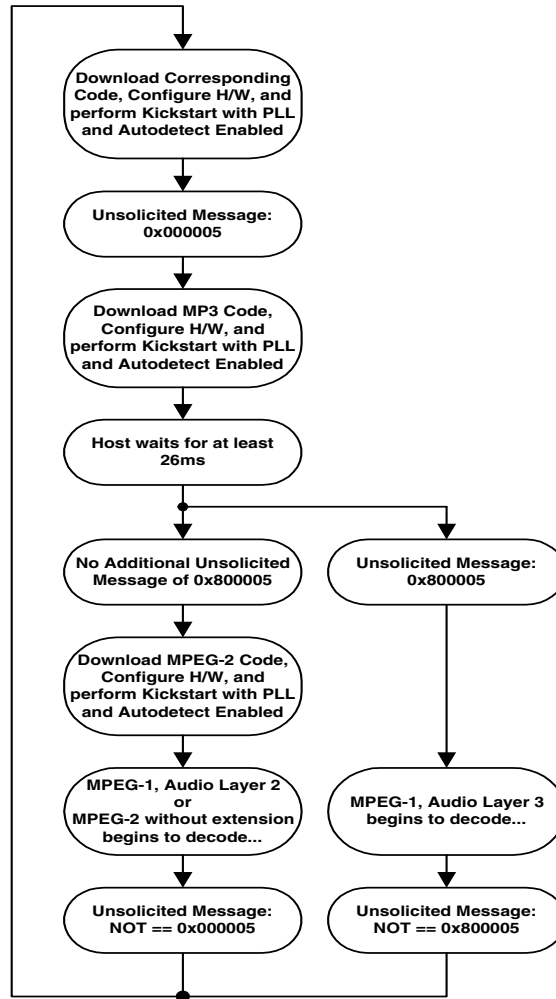
Figure 23. When to Download MP3 Code...

AN163

8. MPEG MULTICHANNEL MANAGER AND OPERATIONS

The module described in this section is specific to MPEG code. In addition to the MPEG Manager, certain operations such as enabling Autodetection of an MPEG stream are also included in this section. MPEG-2 Multichannel application code (mpg_32xx.ld) supports the following types of

MPEG data: MPEG-1, Layer 2 (Pc = 5); MPEG-2 without extension (Pc = 5); MPEG-2 with extension (Pc = 6). Please refer to MP3 application code for support of MPEG-1, Audio Layer 3.

Note: MPEG-2 Multichannel application code (mpg_32xx.ld) does NOT support the following types of MPEG data: MPEG-1, Audio Layer 1 (Pc = 4); or MPEG-2, Audio Layer 1, Low Sampling Frequency (Pc = 8); MPEG-2, Audio Layer 2 or 3, Low Sampling Frequency (Pc = 9).

8.1 MPEG Multichannel Manager

Write Opcode = 0x8C; Read Opcode = 0x0D; Read Response Opcode = 0x8D

Write Message = 0x8C00HH 0xhhhhhh

Read Request Message = 0x0D00HH;

Read Response Message = 0x8D00HH 0xhhhhhh

0xHH = index, 0xhhhhhh = data value

Index	Variable	Dataword Content
0x00	MPEG_CONTROL	**Bits 7:4** = Output_Mode = 0...7 † 0 = 2/0 Lt, Rt Surround Encoded. (Refer to Section 8.1.2 "Down-mix Equation for Lt, Rt" on page 98 for Equation). 1 = 1/0 C. 2 = 2/0 L, R. 3 = 3/0 L, C, R. 4 = 2/1 L, R, S (Ls = Rs = S - 3dB). 5 = 3/1 L, C, R, S (Ls = Rs = S - 3dB). 6 = 2/2 L, R, Ls, Rs. 7 = 3/2 L, C, R, Ls, Rs. **Bit 0 =** MPEG_Enable = 0/1 = Disable/Enable MPEG decoding. *Default = 0x000070* **Note:** Undesired Channel outputs should be explicitly muted by setting the corresponding Ch_Vol (see Section 4.1 "Audio Manager for Dolby Digital, DTS, MP3, MPEG Multichannel, AAC and Crystal Original Surround" on page 33) to 0.0
0x01	ID	0 - Extension to lower frequencies, 1 - ISO/IEC 11172-3 or ISO/IEC 13818-3. ***READ-ONLY***
0x02	LAYER	2 bits to indicate which layer is used. ***READ-ONLY***
0x03	PROTECTION_BIT	0 - stream is CRC protected, 1 - not CRC protected. ***READ-ONLY***

Table 37. MPEG Manager

Index	Variable	Dataword Content
0x04	BITRATE_INDEX	4-bit index to a bitrate table. ***READ-ONLY***
0x05	SAMPLING_FREQUENCY	2-bit index to a sampling frequency table. ***READ-ONLY***
0x06	PADDING_BIT	0 - no padding, 1 - padding exists. ***READ-ONLY***
0x07	PRIVATE_BIT	reserved for non-ISO purposes. ***READ-ONLY***
0x08	MODE	2 bits to indicate the audio configuration mode (for 2 channels only) 00 - Stereo 01 - Joint stereo (intensity stereo and/or ms_stereo) 10 - Dual channel 11 - Single channel **Note:** Changes in this stream variable will result in an Unsolicited Message from the DSP if Audio_Config_Change_Notification has been enabled in the Audio Manager. Please refer to Section 4.1.1 "Audio_Control: (Index 0x00)" on page 37 for more information. ***READ-ONLY***
0x09	MODE_EXTENSION	2 bits to indicate which sub-bands are in intensity stereo. ***READ-ONLY***
0x0a	COPYRIGHT	0 - no copyright, 1 - copyright protected bitstream. ***READ-ONLY***
0x0b	ORIGHOME	0 - bitstream is a copy, 1 - bitstream is an original. ***READ-ONLY***
0x0c	EMPHASIS	2 bits to indicate which type of de-emphasis is used. ***READ-ONLY***
0x0d	EXT_BIT_STREAM_PRESENT	0 - extension bit stream not present, 1 - ext. present. ***READ-ONLY***
0x0e	N_AD_BYTES	8 bits: number of bytes used in MPEG-1 base frame for ancillary data field if an extension bit stream exists. ***READ-ONLY***
0x0f	CENTER	2 bits to indicate presence and bandwidth of center channel. 00 - No center channel present 01 - Center channel present 10 - Not defined 11 - Center bandwidth limited (Phantom coding) **Note:** Changes in this stream variable will result in an Unsolicited Message from the DSP if Audio_Config_Change_Notification has been enabled in the Audio Manager. Please refer to Section 4.1.1 "Audio_Control: (Index 0x00)" on page 37 for more information. ***READ-ONLY***

Table 37. MPEG Manager (Continued)

Index	Variable	Dataword Content
0x10	SURROUND	2 bits to indicate whether surround channels are present or if secondary stereo exists. 00 - No surround 01 - Mono surround 10 - Stereo surround 11 - No surround, but second stereo program present **Note:** Changes in this stream variable will result in an Unsolicited Message from the DSP if Audio_Config_Change_Notification has been enabled in the Audio Manager. Please refer to Section 4.1.1 "Audio_Control: (Index 0x00)" on page 37 for more information. *READ-ONLY*
0x11	LFE	1 bit to indicate whether LFE channel is present. 0 - LFE not present 1 - LFE present **Note:** Changes in this stream variable will result in an Unsolicited Message from the DSP if Audio_Config_Change_Notification has been enabled in the Audio Manager. Please refer to Section 4.1.1 "Audio_Control: (Index 0x00)" on page 37 for more information. *READ-ONLY*
0x12	AUDIO_MIX	1 bit to indicate whether the signal is mixed for large listening rooms. *READ-ONLY*
0x13	DEMATRIX_PROCEDURE	2 bits to indicate which dematrix procedure is used. *READ-ONLY*
0x14	NO_OF_MULTI_LINGUAL_CH	3 bits: number of multi-lingual channels present in the Multichannel extension. *READ-ONLY*
0x15	MULTI_LINGUAL_FS	1 bit to indicate if fs is the same as in the main audio. *READ-ONLY*
0x16	MULTI_LINGUAL_LAYER	1 bit to indicate which layer is used. *READ-ONLY*
0x17	COPYRIGHT_ID_BIT	1 bit: part of a 72-bit copyright identification number. *READ-ONLY*
0x18	COPYRIGHT_ID_START	0 - no start of copyright identification in this frame, 1 - start of copyright identification in this audio frame. *READ-ONLY*

Table 37. MPEG Manager (Continued)

Notes: † See definition on page 32.

8.1.1 MPEG_CONTROL

Bits 7:4: *OUTPUT_MODE*

These bits set up the output speaker downmix configuration. When configuring the OUTPUT _MODE, first the individual volumes of each undesired channel should be set to 0. Next, the desired output mode should be specified. This should be done before kickstart of the application. For example, if output mode 3/0 (L, C, R) is chosen, Ls and Rs volume must explicitly be set to 0 in the Audio Manager module to guarantee no output on those channels.

> 0 = 2/0 Lt, Rt Surround Encoded.
> 1 = 1/0 C.
> 2 = 2/0 L, R.
> 3 = 3/0 L, C, R.
> 4 = 2/1 L, R, S (Ls = Rs = S - 3dB).
> 5 = 3/1 L, C, R, S (Ls = Rs = S - 3dB).
> 6 = 2/2 L, R, Ls, Rs.
> 7 = 3/2 L, C, R, Ls, Rs.

8.1.2 Downmix Equation for Lt, Rt

The following equation is used to produce a Lt, Rt Surround Encoded downmix.

> $Lt = (((1/3.1)L) + ((0.7/3.1)C) - ((0.7/3.1)Ls) - ((0.7/3.1)Rs)$

> $Rt = (((1/3.1)L) + ((0.7/3.1)C) + ((0.7/3.1)Ls) + ((0.7/3.1)Rs)$

8.1.3 MPEG Stream Information

Table 37 on page 95 lists a complete set of *READ-ONLY* variables that are reported to the host by the MPEG Decoder module. These can be read using the appropriate Read Request, and Read Response sessions.

For example, to read the MODE value, host should issue the 3-byte Read Request 0x0D0008 and then read the 6-byte Read Response 0x8D0008, 0x00000h, where the dataword (latter 3 bytes) indicates the reported value of MODE. Please refer to the *MPEG Specification* (ISO/IEC 11172-3 and ISO/IEC 13818-3) for more information on these stream variables.

8.2 MPEG Multichannel Decode with Autodetect

In this mode, the input should be compressed MPEG (of a supported stream type). The data will be taken from the compressed input port as designated by the Hardware Configuration message. If the input on this port is not MPEG, the Autodetect function will notify the host as described in Section 4.3.1 "Autodetect Operation" on page 55.

Figure 24 is an example of pseudocode to configure the part AC-3 decode with autodetect. This pseudocode can be used as a template for other configurations by swapping out the message array and the size.

Module	Index	Description	Opcode & Index	Value
MPEG Manager	MPEG Control	MPEG Enabled (Assumes all 5.1 channels are available.)	0x8C0000	0x000071
Audio Manager	Audio Manager Control	Autodetect Enabled PLL Enabled Kickstart Application	0x880000	0x001011

Table 38. Enabling MPEG Decode with Autodetect

```
void MPEG_AD_Config()
{
    unsigned char MPEG_AD_config_message[] =
                                    {0x8C, 0x00, 0x00,
                                     0x00, 0x00, 0x71,

                                     0x88, 0x00, 0x00,
                                     0x00, 0x10, 0x11};


    Write_*(MPEG_AD_config_message,12); /* Replace * with I2C or SPI depending on protocol
*/
}
```

Figure 24. MPEG Multichannel decode with Autodetect

9. AAC MANAGER AND OPERATIONS

The module described in this section is specific to AAC code. The only AAC-specific module is the AAC Manager. In addition to the AAC-specific module mentioned before, certain operations such as enabling Autodetection of a AAC stream is also included in this section. AAC application code (aac_329x.ld) only supports the following type of AAC data: AAC, Low Complexity, and ADTS Format (Pc = 1C).

9.1 AAC Manager

Write Opcode = 0x9E; Read Opcode = 0x1F; Read Response Opcode = 0x9F

Write = 0x9E00HH 0xhhhhhh

Read Request = 0x1F00HH;

Read Response = 0x9F00HH 0xhhhhhh

0xHH = index, 0xhhhhhh = data value

Index	Variable	Dataword Content
0x00	AAC_CONTROL	**Bits 7:4** = Output_Mode = 0...7 †
		0 = 2/0 Lt, Rt Surround Encoded. (Equation in Section 9.1.2 "Downmix Equation for Lt, Rt" on page 102).
		1 = 1/0 C.
		2 = 2/0 L, R.
		3 = 3/0 L, C, R.
		4 = 2/1 L, R, S (Ls = Rs = S - 3dB).
		5 = 3/1 L, C, R, S (Ls = Rs = S - 3dB).
		6 = 2/2 L, R, Ls, Rs.
		7 = 3/2 L, C, R, Ls, Rs.
		Bit 0 = AAC_Enable = 0/1 = Disable/Enable AAC decoding. †
		Default* = 0x000070
0x01	PROFILE	0x0...0x3 = profile used. See *ISO/IEC 13818-7*, subclause 2.1.
		Note: Only Low Complexity (LC) profile == 0x1 is supported by this application code.
		READ-ONLY
0x02	FS_INDEX	0x0...0xF: Index to AAC sampling frequency table. Only the following frequencies are supported: 0x3 = 48 kHz 0x4 =44.1 kHz 0x5 =32 kHz ***READ-ONLY***
0x03	PCE_PRESENT	0 - The num_?_channel elements variables (see 0x04 - 0x06 below) are not valid 1 - The num_?_channel_elements variables (see 0x04 - 0x06 below) are valid ***READ-ONLY***

Table 39. AAC Manager

Index	Variable	Dataword Content
0x04	NUM_FRONT_CHANNEL_ELEMENTS	Number of audio syntactic elements in the front channels, front center to back center, symmetrically by left and right, or alternating by left and right in the case of single channel elements. *READ-ONLY*
0x05	NUM_FRONT_CHANNELS	Number of front channels. This field is valid only when channel_configuration is zero. *READ-ONLY*
0x06	NUM_BACK_CHANNEL_ELEMENTS	Number of side and front channel elements, for back channels. *READ-ONLY*
0x07	NUM_BACK_CHANNELS	Number of back channels. This field is valid only when channel_configuration is zero. *READ-ONLY*
0x08	NUM_LFE_CHANNEL_ELEMENTS	Number of LFE channel elements associated with the program. *READ-ONLY*
0x09	HOME	**Bit 0 =** Home bit from the stream. *READ-ONLY*
0x0a	ORIGINAL_COPY	**Bit 0 =** Original/Copy bit from the input stream. *READ-ONLY*
0x0b	COPYRIGHT_ID_PRESENT	**Bit 0 =** When 1, Index 0x0c to 0x0e carries the last received copyright identification field. *READ-ONLY*
0x0c	COPYRIGHT_ID_WORD_1	**Bits 0:23** = Map to the first 24 bits of the 72-bit copyright identification field. **Bit 23** is the first bit of the field. *READ-ONLY*
0x0d	COPYRIGHT_ID_WORD_2	**Bits 0:23** = Map to the second 24 bits of the 72-bit copyright identification field. **Bit 23** is the 25th bit of the field, while **Bit 0** is the 48th bit of the field. *READ-ONLY*
0x0e	COPYRIGHT_ID_WORD_3	**Bits 0:23** = Map to the second 24 bits of the 72-bit copyright identification field. **Bit 0** is the 72nd bit of the field. *READ-ONLY*
0x0f	FRAME_LENGTH	13 bit field: length of the frame in bytes, including headers and error_check. *READ-ONLY*
0x10	NUMBER_OF_RAW_DATA_BLOCKS_IN_FRAME	Number of raw_data_blocks in the ADTS frame. *READ-ONLY*

Table 39. AAC Manager (Continued)

Index	Variable	Dataword Content
0x11	CHANNEL_CONFIGURATION	0x0...0x7 = channel configuration used. If *channel_configuration* is greater than 0, the channel configuration is given by the default bitstream index number in Table 41, Implicit Speaker Mapping. If channel_configuration equals 0, the channel configuration is not specified in the header and must be given by a program_config_element following as first bitstream element in the first raw_data_block after the header, or by the implicit configuration (See *ISO/IEC 13818-7,* subclause 3.5) or must be known in the application.
		Note: Changes in this stream variable will result in an Unsolicited Message from the DSP if Audio_Config_Change_Notification has been enabled in the Audio Manager. Please refer to Section 4.1.1 "Audio_Control: (Index 0x00)" on page 37 for more information.
		READ-ONLY
0x12	PROTECTION_ABSENT	0 - stream is CRC protected, 1 - stream is not CRC protected.
		READ-ONLY

Table 39. AAC Manager (Continued)

Notes: † See definition on page 32.

"Default" vs. "Default*" See definition page 32.

9.1.1 AAC_CONTROL

Bits 7:4: *OUTPUT_MODE*

These bits set up the output speaker downmix configuration. When configuring the OUTPUT _MODE, first set the individual volumes of each undesired channel to 0. Next, the desired output mode should be specified. This should be done before kickstart of the application. For example, if output mode 3/0 (L, C, R) is chosen, Ls and Rs volume must explicitly be set to 0 in the Audio Manager module to guarantee no output on those channels.

> 0 = 2/0 Lt, Rt Surround Encoded.
> 1 = 1/0 C.
> 2 = 2/0 L, R.
> 3 = 3/0 L, C, R.
> 4 = 2/1 L, R, S (Ls = Rs = S - 3dB).
> 5 = 3/1 L, C, R, S (Ls = Rs = S - 3dB).
> 6 = 2/2 L, R, Ls, Rs.
> 7 = 3/2 L, C, R, Ls, Rs.

9.1.2 Downmix Equation for Lt, Rt

The following equation is used to produce a Lt, Rt Surround Encoded downmix.

$Lt = (((1/3.1)L) + ((0.7/3.1)C) - ((0.7/3.1)Ls) - ((0.7/3.1)Rs)$

$Rt = (((1/3.1)L) + ((0.7/3.1)C) + ((0.7/3.1)Ls) + ((0.7/3.1)Rs)$

9.1.3 AAC Stream Information

Table 39 lists a complete set of READ-ONLY variables that are reported to the host by the AAC Decoder module. These can be read using the appropriate Read Request, Read Response sessions.

For example, to read the PROFILE value, host should issue the 3-byte Read Request 0x1f0001 and then read the 6-byte Read Response 0x9f0001, 0x00001h, where the dataword (latter 3 bytes) indicates the reported value of PROFILE. Please refer to the *MPEG-2 AAC Specification* (ISO/IEC 13818-7) for more information on these stream variables.

9.2 AAC Decode with Autodetect

In this mode, the input should be compressed ADTS formatted AAC. The data will be taken from the compressed input port as designated by the Hardware Configuration message. If the input on this port is not AAC, the Autodetect function will notify the host as described in Section 4.3.1 "Autodetect Operation" on page 55.

Figure 25 is an example of pseudocode to configure the part AC-3 decode with autodetect. This pseudocode can be used as a template for other configurations by swapping out the values for the message array and the size in the pseudocode.

Module	Index	Description	Opcode & Index	Value
AAC Manager	AAC Control	AAC Enabled (Assumes all 5.1 channels are available.)	0x9E0000	0x000071
Audio Manager	Audio Manager Control	Autodetect Enabled PLL Enabled Kickstart Application	0x880000	0x001011

Table 40. Enabling AAC Decode with Autodetect

```
void AAC_AD_Config()
{
    unsigned char AAC_AD_config_message[] =
                                    {0x9E, 0x00, 0x00,
                                     0x00, 0x00, 0x71,

                                     0x88, 0x00, 0x00,
                                     0x00, 0x10, 0x11};

    Write_*(AAC_AD_config_message,12); /* Replace * with I2C or SPI depending on protocol
*/
}
```

Figure 25. AAC decode with Autodetect

9.2.1 Implicit Speaker Mapping

Default Bitstream Index Number	Number of Speakers	Audio Syntactic Elements, Listed in Order Received	Default Element to Speaker Mapping
0	See Note.	See Note.	**Note:** If channel configuration is equal to zero, the speaker mapping is given by the PCE.
1	1	single_channel_element	center front speaker
2	2	channel_pair_element	left, right front speakers
3	3	single_channel_element, channel_pair_element	center front speaker left, right front speakers
4	4	single_channel_element, channel_pair_element, single_channel_element	center front speaker left, right front speakers rear surround
5	5	single_channel_element, channel_pair_element, channel_pair_element	center front speaker left, right front speakers left surround, right surround rear speakers
6	5+1	single_channel_element, channel_pair_element, channel_pair_element, lfe_element	center front speaker left, right front speakers left surround, right surround rear speakers front low frequency effects speaker
7	7+1	single_channel_element, channel_pair_element, channel_pair_element, channel_pair_element, lfe_element	center front speaker left, right center front speakers left, right outside front speakers left surround, right surround rear speakers front low frequency effects speaker

Table 41. Implicit Speaker Mapping

10. CRYSTAL ORIGINAL SURROUND (C.O.S.) SPECIFIC MODULES AND OPERATIONS

The two modules described in this section are specific to C.O.S. code. They are: the C.O.S. Effects Manager and the C.O.S. PCM Processor. In addition to the C.O.S. specific-modules, certain operations such as enabling Autodetection of a PCM stream is also included in this section, along with a discussion of: reverb, echo, filter techniques,

as well as using COEFFMKR.EXE and EFFECTS.EXE in order to help programmers generate their own processing effects configurations. Some sample configurations have been listed to give you an idea of possible techniques you might utilize in your development efforts. The samples given here do not exhaust all the different ways that you might create your own processing effects.

10.1 C.O.S. Effects Manager

Write Opcode = 0x98; Read Opcode = 0x19; Read Response Opcode = 0x99

Write Message = 0x9800HH 0xhhhhhh

Read Request Message = 0x1900HH;

Read Response Message = 0x9900HH 0xhhhhhh

0xHH = index, 0xhhhhhh = data value

Index	Variable	Dataword Content
0x00	EFFECTS_MGR_CONTROL	**Bit 8** = Test Impulse Enable = 0/1 = Disable/Enable replacement of L/R input with periodic impulse signal (repeated every 24000 samples). †
		Bit 4 = Deemphasis_Enable = 0/1 = Disable/Enable PCM Deemphasis. †
		Note: Please refer to the deemphasis section, located after this table, for more information about using digital deemphasis for PCM sampling frequencies other than 44.1 kHz when using IBA-based codes.
		Bit 0 = Effects Enable = 0/1 = Disable/Enable Surround Effects.
		Default = 0x000000
0x01	L_MONO_MIXLEVEL	-1.0 to 1.0. † *Default = 0.5*
0x02	R_MONO_MIXLEVEL	-1.0 to 1.0. † *Default = 0.5*
0x03	DIRECT_FIR_MIXLEVEL	-1.0 to 1.0. † *Default = 0.5*
0x04	LPF_FIR_MIXLEVEL	-1.0 to 1.0. † *Default = 0.5*
0x05	RESERVED	For future use.
0x06	RESERVED	For future use.
0x07	RESERVED	For future use.
0x08-0x27	T0, T1, ... T30, T31	0...5615 = Delay in samples for each of the 32 taps.
		Default = 0.
		Note: Setting is in samples, and will thus produce different delays for different sampling frequencies.

Table 42. C.O.S. Effects Manager

Index	Variable	Dataword Content
0x28-0x47	C0, C1,... C31	-1.0 to 1.0 = Coefficients for each of the 32 taps. *Default = 0.0*
0x48	DIRECT_REVERB_MIXLEVEL	-1.0 to 1.0. † *Default = 0.875*
0x49	FIR_REVERB_MIXLEVEL	-1.0 to 1.0. † *Default = 0.125*
0x4a	REVERB_TIME (COMB_G)	0.0 to 1.0 = Reverberation time: unitless constant (minimum 0.0 to maximum 1.0). *Default = 0.5*
0x4b	REVERB_LIVENESS (COMB_H)	0.0 to 1.0 = High-frequency attenuation in reverberator: unitless constant. *Default = 0.2*
0x4c	L_L_OUT_MIXLEVEL	-1.0 to 1.0. † *Default = 1.0*
0x4d	R_L_OUT_MIXLEVEL	-1.0 to 1.0. † *Default = 0.0*
0x4e	L_C_OUT_MIXLEVEL	-1.0 to 1.0. † *Default = 0.5*
0x4f	R_C_OUT_MIXLEVEL	-1.0 to 1.0. † *Default = 0.5*
0x50	L_R_OUT_MIXLEVEL	-1.0 to 1.0. † *Default = 0.0*
0x51	R_R_OUT_MIXLEVEL	-1.0 to 1.0. † *Default = 1.0*
0x52	L_LS_OUT_MIXLEVEL	-1.0 to 1.0. † *Default = 0.5*
0x53	R_LS_OUT_MIXLEVEL	-1.0 to 1.0. † *Default = 0.5*
0x54	L_RS_OUT_MIXLEVEL	-1.0 to 1.0. † *Default = 0.5*
0x55	R_RS_OUT_MIXLEVEL	-1.0 to 1.0. † *Default = 0.5*
0x56	L_LFE_OUT_MIXLEVEL	-1.0 to 1.0. † *Default = 0.5*
0x57	R_LFE_OUT_MIXLEVEL	-1.0 to 1.0. † *Default = 0.5*
0x58	FIR_L_OUT_MIXLEVEL	-1.0 to 1.0. † *Default = 0.0*
0x59	REVERB_L_OUT_MIXLEVEL	-1.0 to 1.0. † *Default = 0.0*
0x5a	FIR_C_OUT_MIXLEVEL	-1.0 to 1.0. † *Default = 0.0*
0x5b	REVERB_C_OUT_MIXLEVEL	-1.0 to 1.0. † *Default = 0.0*
0x5c	FIR_R_OUT_MIXLEVEL	-1.0 to 1.0. † *Default = 0.0*
0x5d	REVERB_R_OUT_MIXLEVEL	-1.0 to 1.0. † *Default = 0.0*
0x5e	FIR_LS_OUT_MIXLEVEL	-1.0 to 1.0. † *Default = 0.0*
0x5f	REVERB_LS_OUT_MIXLEVEL	-1.0 to 1.0. † *Default = 0.0*
0x60	FIR_RS_OUT_MIXLEVEL	-1.0 to 1.0. † *Default = 0.0*
0x61	REVERB_RS_OUT_MIXLEVEL	-1.0 to 1.0. † *Default = 0.0*
ADVANCED SETTINGS		
0x62	COMB_0_DELAY_SIZE	1...1103 samples. *Default=887*
0x63	COMB_1_DELAY_SIZE	1...1213 samples. *Default=947*
0x64	COMB_2_DELAY_SIZE	1...1321 samples. *Default=1069*
0x65	COMB_3_DELAY_SIZE	1...1433 samples. *Default=1297*
0x66	COMB_4_DELAY_SIZE	1...1543 samples. *Default=1439*
0x67	COMB_5_DELAY_SIZE	1...1657 samples. *Default=1571*
0x68	COMB_0_G1	0.0 to 1.0. *Default=0.46*
0x69	COMB_1_G1	0.0 to 1.0. *Default=0.48*
0x6a	COMB_2_G1	0.0 to 1.0. *Default=0.50*
0x6b	COMB_3_G1	0.0 to 1.0. *Default=0.52*
0x6c	COMB_4_G1	0.0 to 1.0. *Default=0.53*
0x6d	COMB_5_G1	0.0 to 1.0. *Default=0.55*

Table 42. C.O.S. Effects Manager (Continued)

Index	Variable	Dataword Content
0x6e	COMB_0_MIXLEVEL	0.0 to 1.0. *Default=0.9*
0x6f	COMB_1_MIXLEVEL	0.0 to 1.0. *Default=0.9*
0x70	COMB_2_MIXLEVEL	0.0 to 1.0. *Default=0.9*
0x71	Comb_3_Mixlevel	0.0 to 1.0. *Default=0.9*
0x72	Comb_4_Mixlevel	0.0 to 1.0. *Default=0.9*
0x73	Comb_5_Mixlevel	0.0 to 1.0. *Default=0.9*
0x74	Allpass_g1	0.0 to 1.0. *Default=0.7*
0x75	Allpass_g2	0.0 to 1.0. *Default=0.51*

Table 42. C.O.S. Effects Manager (Continued)

Notes: †   See definition on

10.1.1 EFFECTS_MGR_CONTROL

Bit 8: *Test Impulse Enable*

Disable/Enable replaces L/R input with periodic impulse signal (every 24K samples). In this mode, disabling of autodetection causes pulse generation to be independent of the input. If autodetect is left enabled, a pulse will only be generated if PCM is presented at the active PCM input.

> 1 - Enables Test Impulse
> 0 - Disables Test Impulse

Bit 4: *Deemphasis_Enable*

Disable/Enable PCM Deemphasis. The filter used is identical to CS4334 (50/15 uS).

> 1 - Enables Deemphasis
> 0 - Disables Deemphasis

For systems that plan on using a "Non-IBA-Based" PCM Processing application code such as eff_32x6.ld and want to support digital deemphasis for the sampling frequencies of 32 kHz and 48 kHz, as well as 44.1 kHz, PCM, the host only needs to inform the DSP the setting for the current PCM sampling frequency, which comes via the Audio Manager (0x14) before kickstart.

Note: For systems that plan on using an "IBA-Based" PCM Processing application code such as effi32xx.ld and want to support digital deemphasis for the sampling frequencies of 32 kHz and 48 kHz, as well as 44.1 kHz, PCM, the host must, in addition to communicating the current PCM sampling frequency setting via the Audio Manager (0x14), also download the following commands before kickstart:

For 32 kHz support:

> 848fff
> bb82fa
> 848fe8
> 42aef3
> 848fe9
> f8d406

For 44.1 kHz support:

> 848ffe
> aeacda
> 848fe6
> 3b6543
> 848fe7
> f34797

For 48 kHz support:

> 848ffd
> ab9ed7
> 848fe4
> 39be36
> 848fe5
> f1e0a1

For 64 kHz support:

> 848fff
> a25a7b
> 848fe8
> 34f310
> 848fe9
> ed676b

For 88.2 kHz support:

> 848ffe
> 99f80b
> 848fe6
> 30eb8a
> 848fe7
> e90c82

For 96 kHz support:

> 848ffd
> 981257
> 848fe4
> 300d08
> 848fe5
> e8054f

Digital deemphasis support does not exist for sampling frequencies other than 96 kHz, 88.2 kHz, 64 kHz, 48 kHz, 44.1 kHz, or 32 kHz, regardless of what code is being downloaded.

Bit 0: *Effects Enable*

Disable/Enable Surround Effects.

> 1 - Enables Effects
> 0 - Disables Effects

Note: Surround Effects parameters (listed without the † symbol) are sampled by the CS4932X only once just after kickstart (after a download/Soft Reset or Application Restart). These parameters cannot be varied at run-time. Therefore, to change modes, one has to issue a Soft Reset or Application Restart, reconfigure hardware settings, issue the new Effects messages and then kickstart.

10.1.2 Basic Crystal Surround Effects Explanation

Crystal Original Surround Effects is based on public-domain research on acoustic reverberation. For a more detailed discussion of the properties of the Cirrus Logic implementation, see "About This Reverberation Business", James A. Moorer, *Computer Music Journal* 3(2): 13-28, 1979.

Figure 26 is an illustration of basic reverb theory. In this figure, E-R should be interpreted as Early Reflections, while A-R should be interpreted as After Reflections. Figure 27 illustrates the various modules of the Crystal Original Surround Effects and the corresponding control parameters available to the user. This forms the Surround Effects Processor whose outputs (square nodes with bold legend) are available to the Output Channel Mixer. The Output Channel Mixer also receives the LR stereo pair input.

If Deemphasis is enabled, the LR stereo input is first de-emphasized (50/15 uS). This stage is not shown in Figure 27 and is required only if the input LR PCM input has been pre-emphasized at the source.

The next stage allows for Equalization of the input LR stereo pair. When EQ is enabled, stereo PCM input to the Effects Module (after de-emphasis if enabled) is first attenuated by the Pre_EQ_Attenuation scale factor (specified in usual 1.23 format), and then processed by the cascade of three EQ biquads specified by the {EQ0...2: b2, b1, b0, a2, a1} coefficients. The output of the last EQ biquad is then scaled by the Post_EQ_Gain scale factor (specified in 6.18 format to allow for up to 30 dB of gain, i.e., scale factor of 32). The EQ-ed stereo pair output then forms the LR input for all further effects processing.

The Equalization module affects the signal at the input, while Tone Control (refer to Section 4.5 "Generalized Tone Control Manager for Dolby Digital, DTS, MPEG Multichannel and C.O.S." on page 63) is designed to only boost the bass or treble just before the Bass Manager.

The mono input for the surround processing is derived by mixing the input LR stereo pair with user parameters *L_Mono_Mixlevel* and *R_Mono_Mixlevel*. No internal protection or checking is applied to prevent overflow. This is also true for all other summing stages in the application. Therefore, the user settings should ensure that the sum of the magnitudes of summing mix levels does not exceed unity.

In addition, an impulse generator (not shown in Figure 27) is also provided for testing purposes. In this mode, the input L/R stereo pair is zeroed out, and a unit amplitude impulse is inserted in both input channels every 24000 samples (half a second at Fs = 48 KHz).

The LPF at the input of the FIR delay structure is a first-order filter with a corner frequency of Fs/6, i.e., 8 KHz for Fs = 48 KHz and 7.35 KHz for Fs = 44.1 KHz. The variable mix levels, *Direct_FIR_Mixlevel* and *LPF_FIR_Mixlevel* allow one to select the amount of high frequencies to be passed into the FIR stage of the surround processor. This can be used to vary the amount of "brightness" of the surround effect.

The Delay Buffer can hold up to 5616 samples which corresponds to 117 mS at Fs = 48 KHz, and 127.35 mS at Fs = 44 KHz. Thirty two user-specifiable taps *T0-T31* are available from this delay buffer. The settings *T0-T31* specify the absolute number of samples of delay (0... 5615). The tap delays need not be in monotonic or any specific ordering, and can be arbitrarily sequenced. For example, *T0*=4000 samples, *T1* = 300 samples, *T31* = 0 samples are valid settings.

An FIR filter can be constructed using these delay buffer taps with coefficients *C0-C31* corresponding to the taps *T0-T31*. This FIR filter can be used to model the Early Reflections of a particular room. The FIR output is available to the Output Channel Mixer directly. Tap 31 is available as an input to the reverberator for further surround effects processing.

The mono input to the Reverberator Module is derived by mixing the output of Tap 31 of the delay buffer and the FIR output using user-specified parameters, *Direct_Reverb_Mixlevel* and *FIR_Reverb_Mixlevel* respectively. Note that Tap 31 defaults to a delay of 0, i.e., the direct output of Variable LPF structure. Tap 31 can be adjusted to align the impulse response of the Reverberator Module (After Reflections) with the end of the Early Reflections produced by the FIR filter.

The Reverberator Module accepts user settings for *Reverb_Time* and *Reverb_Liveness*, which are both unitless constants. A larger value of *Reverb_Time* increases the duration of the reverberation effect, i.e., the period of sustained echo. A larger value of *Reverb_Liveness* decreases the high frequency content (increased high-frequency attenuation) in the reverberation. The Reverberator Module produces 5 outputs to the Output Channel Mixer, one for each output full-bandwidth channel. A more detailed treatment of the advanced settings possible in the reverberator, as well as information on creating custom effects modes is presented in Section 10.2.2. Figure 28 illustrates the Output Channel Mixer

Note: *"Mix level"* abbreviated as *"Mlv."* due to lack of space.

Each of the five full-bandwidth output channels (L, C, R, Ls, and Rs) is derived as a user-specifiable mix of the input L/R stereo pair, the FIR output, and the corresponding output channel of the Reverberator Module. The Low Frequency Effects (LFE) channel output carries no surround processing, and is only a user-specifiable mix of the input L/R stereo pair.

Disabling the surround effects can be used for simple PCM pass through.
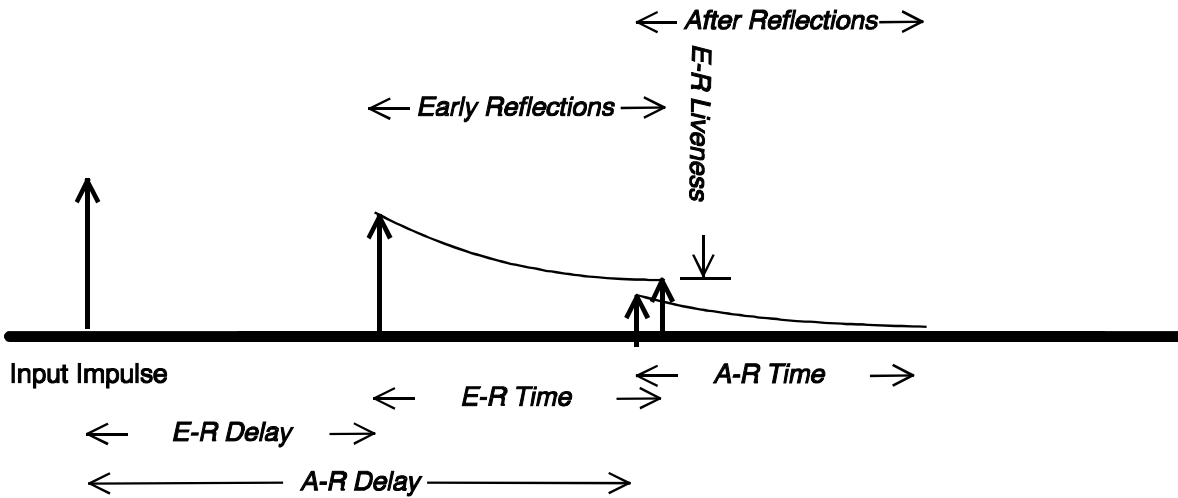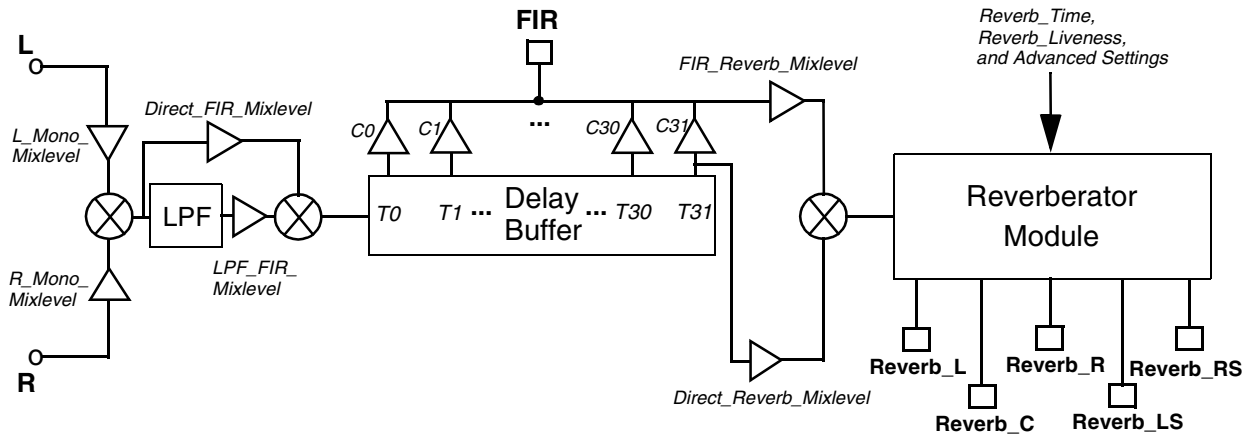
Figure 26. Illustrated Basic Reverb Theory

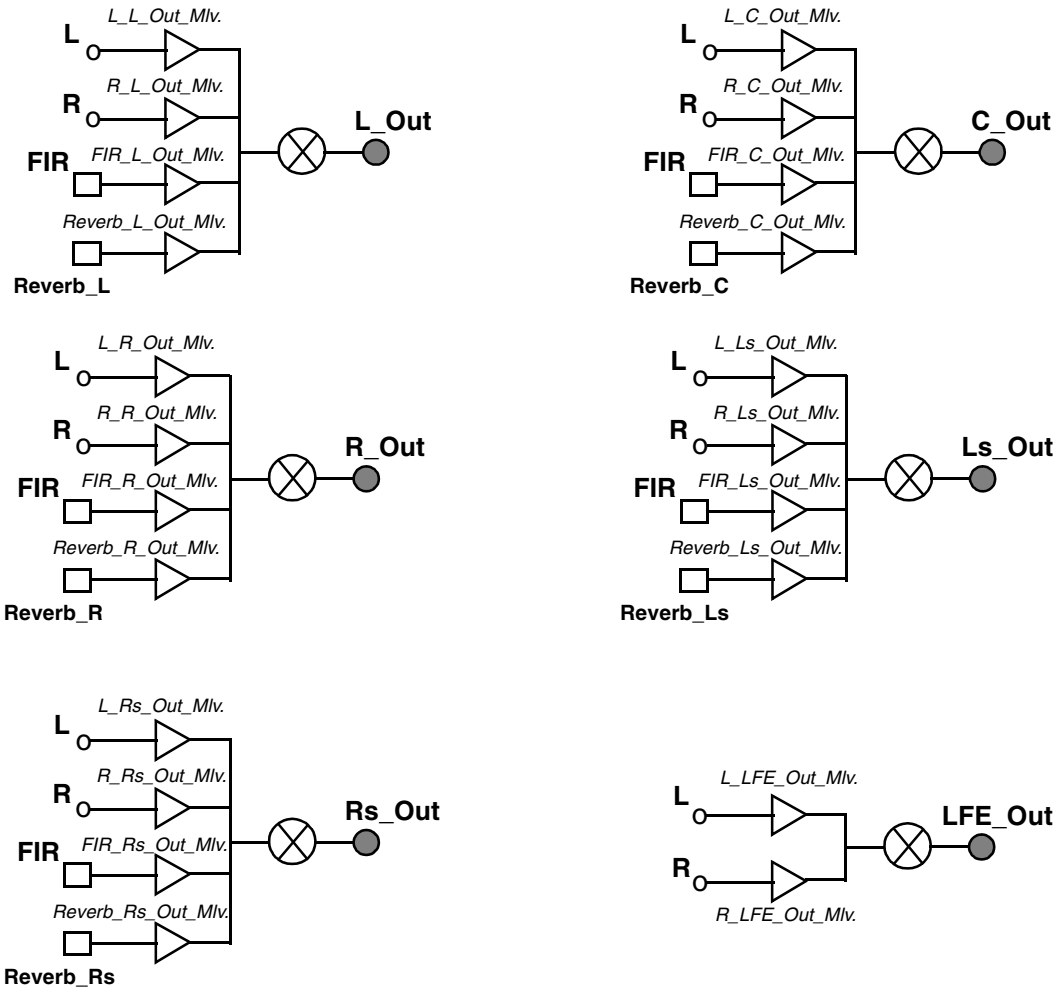**Figure 27. Block Diagram of Effects Code**

Figure 28. Block Diagram of Effects Channel Mixer

10.2 C.O.S. PCM Processor

Write Opcode = 0x9C; Read Opcode = 0x1D; Read Response Opcode = 0x9D

Write = 0x9C00HH 0xhhhhhh

Read Request = 0x1D00HH;

Read Response = 0x9D00HHh   0xhhhhhh

0xHH = index

0xhhhhhh = data value

Index	Variable	Dataword Content
0x00	PCM_CONTROL	**Bit 4** = PCM_Input_EQ_Enable = 0/1 = Disable/Enable 3-band EQ filters on input PCM. **Bit 0** = PCM_Divider_FIR_Enable = 0/1 = Disable/Enable FIR filter on PCM output of Ls/Rs channels. *Default = 0x000000*
0x01-0x10	DIV_C0, DIV_C1, DIV_C15	-1.0 to 1.0 = Coefficients for each of the 16 taps of the FIR filter. *Default* DIV_C0 = 1.0* *Default* DIV_C1-15 = 0.0*
0x11	PRE_EQ_ATTENUATION	-1.0 to 1.0 = Pre-EQ attenuation scale factor to setup desired headroom for EQ filtering. † *Default* = 1.0*
0x12	EQ0_B2	-4.0 to 4.0 in 3.21 format = Biquad coefficient for EQ Filter 0. *Default* = 1.0 (0x200000)*
0x13	EQ0_B1	-4.0 to 4.0 in 3.21 format = Biquad coefficient for EQ Filter 0. *Default* = 0.0*
0x14	EQ0_B0	-4.0 to 4.0 in 3.21 format = Biquad coefficient for EQ Filter 0. *Default* = 0.0*
0x15	EQ0_A2	-4.0 to 4.0 in 3.21 format = Biquad coefficient for EQ Filter 0. *Default* = 0.0*
0x16	EQ0_A1	-4.0 to 4.0 in 3.21 format = Biquad coefficient for EQ Filter 0. *Default* = 0.0*
0x17	EQ1_B2	-4.0 to 4.0 in 3.21 format = Biquad coefficient for EQ Filter 1. *Default* = 1.0 (0x200000)*
0x18	EQ1_B1	-4.0 to 4.0 in 3.21 format = Biquad coefficient for EQ Filter 1. *Default* = 0.0*
0x19	EQ1_B0	-4.0 to 4.0 in 3.21 format = Biquad coefficient for EQ Filter 1. *Default* = 0.0*
0x1a	EQ1_A2	-4.0 to 4.0 in 3.21 format = Biquad coefficient for EQ Filter 1. *Default* = 0.0*
0x1b	EQ1_A1	-4.0 to 4.0 in 3.21 format = Biquad coefficient for EQ Filter 1. *Default* = 0.0*

Table 43. C.O.S. PCM Processor

Index	Variable	Dataword Content
0x1c	EQ2_B2	-4.0 to 4.0 in 3.21 format = Biquad coefficient for EQ Filter 2. *Default* = 1.0 (0x200000)*
0x1d	EQ2_B1	-4.0 to 4.0 in 3.21 format = Biquad coefficient for EQ Filter 2. *Default* = 0.0*
0x1e	EQ2_B1	-4.0 to 4.0 in 3.21 format = Biquad coefficient for EQ Filter 2. *Default* = 0.0*
0x1f	EQ2_A2	-4.0 to 4.0 in 3.21 format = Biquad coefficient for EQ Filter 2. *Default* = 0.0*
0x20	EQ2_A1	-4.0 to 4.0 in 3.21 format = Biquad coefficient for EQ Filter 2. *Default* = 0.0*
0x21	POST_EQ_GAIN	-32.0 to 32.0 in 6.18 format = Post-EQ gain scale factor to compensate for Pre-EQ attenuation and EQ filter gains and maximize digital dynamic range usage. † *Default* = 1.0 (0x040000)*

Table 43. C.O.S. PCM Processor (Continued)

Notes: † See definition on .

"Default" vs. "Default*" See definition .

10.2.1 PCM_CONTROL

Bit 4: *INPUT_PCM_EQ_ENABLE*

Switches ON/OFF EQ of input PCM.

    1 - Enables Input PCM EQ
    0 - Disables Input PCM EQ

When EQ is enabled, stereo PCM input to the Effects Module (after de-emphasis if enabled) is first attenuated by the Pre_EQ_Attenuation scale factor (specified in usual 1.23 format), and then processed by the cascade of three EQ biquads specified by the {EQ0...2: b2, b1, b0, a2, a1} coefficients. The output of the last EQ biquad is then scaled by the Post_EQ_Gain scale factor (specified in 6.18 format to allow for up to 30 dB of gain, i.e., scale factor of 32). The EQ-ed stereo pair output then forms the LR input for all further effects processing.

Note: All EQ parameters should be downloaded by the host controller before kickstart.

For an input x(n) the output y(n) of each EQ biquad filter is given by:

$$y(n) = x(n)*b0 + x(n-1)*b1 + x(n-2)*b2 + y(n-1)*a1 + y(n-2)*a2$$

Note: The coefficients b2, b1, b0, a2, a1 are all specified in 3.21 format to allow for coefficient magnitudes of up to 8.0.

Bit 0: *PCM_Divider_FIR_Enable*

Switches ON/OFF the PCM Divider FIR.

    1 - Enables FIR Filter on Ls and Rs PCM output
    0 - Disables FIR Filter

PCM produced for output at the Ls and Rs channels according to the Effects setup is passed individually through 2 identical 16-tap FIR filters (assuming Control above has been turned ON). This forms the input to the Bass Manager (if it is turned ON).

For an input x(n), the output y(n) of the FIR filter is defined by:

$$y(n) = x(n)*DIV\_C0 + x(n-1)*DIV\_C1 + x(n-2)*DIV\_C2 ... + x(n-15)*DIV\_C15$$

The coefficients *DIV_C0...15* are set by the host, using messaging into the indices as specified above.

10.2.2 Creating Custom Effects Using Advanced Settings

This section describes the Reverberator module. Figure 29 shows a block diagram of the internal implementation of the reverberator module.

10.2.2.1 FIR Section: Early Reflections

The FIR filter shown in Figure 29 allows one to simulate Early Reflections and model specific room acoustics. Given a particular set of FIR filter taps and coefficients (generated by actual room measurements and/or listening experiments), various sizes of rooms can be derived by simply scaling the tap locations (not the coefficients). Other effects can be generated by moving back all of the taps to simulate a delay in the Early Reflections. The amplitude of the coefficients can be scaled according to the desired liveness of the room. If the amplitude of the coefficients does not fall off quickly, then the room will appear more live (that is, later Early Reflections will be more present in the output).

10.2.2.2 Comb Filter Section: After-reflections

Following the Early Reflections simulated by the FIR, comb filters are used to simulate the dense After Reflections that are caused by walls and transitions thereof. Each comb filter reproduces the input signal with periodic delay and decaying amplitude. Furthermore, a first order low pass filter is included in the feedback to simulate the absorption of high frequencies by air. This produces a more realistic or 'live' reverberation that sounds less 'tinny'.

Six combs are used in parallel to produce dense After Reflections. The delay sizes of the combs are chosen to be mutually prime to avoid coinciding echoes from different combs. It is recommended to set the delays of each comb as roughly 10% more than the previous one. However comb sizes with arrangements that are too regular can cause a buzzy output.

The coefficients G1 and G2 for a comb should satisfy G1+G2<1 for stability. The scheme for controlling the parameters in the CS493XX guarantees this. The individual default Comb_[0-5]_g1 can be changed. G1 and G2 are derived for each comb automatically as shown in Figure 29 using Comb_g and Comb_h as well as Comb_[0-5]_g1.

Increasing (or decreasing) the Reverb_Time (Comb_g) parameter increases (or decreases) the length of the after-reflection tail since it scales up (or down) the feedback in the comb.

Increasing (or decreasing) the Reverb_Liveness (Comb_h) parameter increases (or decreases) the amount of LPF action in the feedback, thus making the sound more (or less) dampened. This is particularly important to reduce the clicking effect when reverberating short impulsive sounds.

While experimenting, it may be found that a particular comb's reverberation (among the six) is audibly predominant (buzzing sound). To overcome this, the corresponding Comb_[0-5]_Mixlevel for that comb can be reduced to equalize its audible effect with the other combs. These can also be used to overcome overflow issues that may happen with different choices of g1, g, and h.

10.2.2.3 All-Pass Filter Section: More After-reflections and Decorrelation

All-Pass filters are another means of simulating After Reflections. In addition, they are used here to decorrelate the different channel outputs from each other.

In the CS493XX implementation, the delay sizes are fixed (as specified in Figure 29) and all of the All-Pass filters share the same values of G1 and G2. For stability, $G2 <= 1 - G1 ^ 2$ should be guaranteed by the host if these values are changed.

Increasing (or decreasing) the G2 parameter increases (or decreases) the reverberation time (length of after-reflection tail) by scaling up (or down) the reverberated contribution to the output.
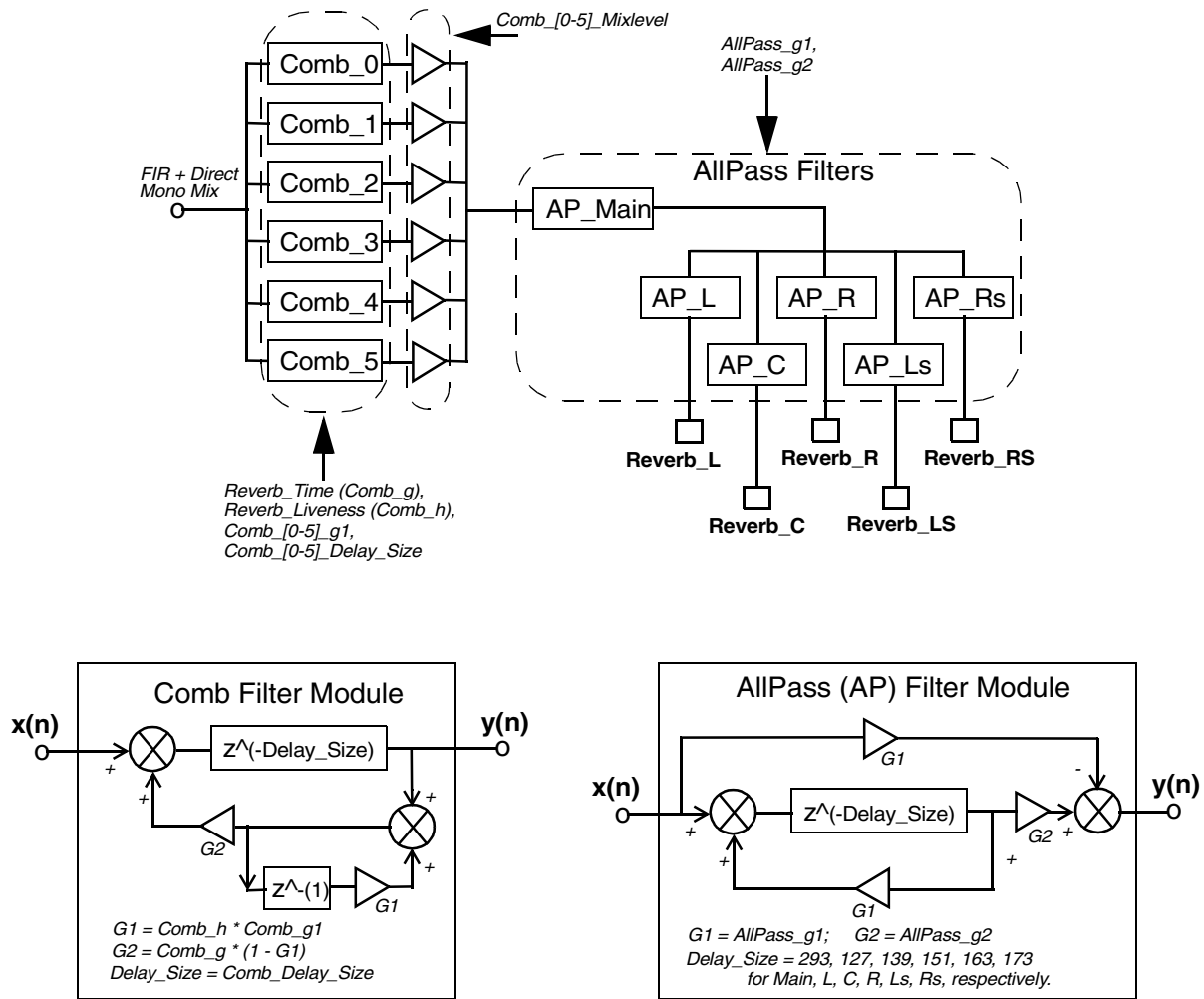
Figure 29. Block Diagram of Effects Reverberator Module

10.2.3 Crystal Original Surround Effects Modes

This section describes some standard effects modes that are provided along with the application microcode. Each mode is setup by issuing an Application Restart message or Soft Reset (assuming the microcode is already downloaded), sending any required hardware configuration messages, and finally sending the corresponding configuration message set for each surround effects mode. For each mode described, the configuration settings are given. Although one can assume the default values as listed in Table 42, the provided message sets explicitly set up all the parameters in most cases. Thus they can serve as templates for any future custom development by the user.

Note: Surround Effects parameters are sampled by the CS4932X only once just after kickstart (after a download/soft reset or Application Restart). These parameters cannot be varied at run-time. Therefore, to change modes, one has to issue a Soft Reset or Application Restart, reconfigure hardware settings, issue the new Effects messages and then kickstart.

The included effects modes were all derived from one of two basic filter structures:

A basic 7-point FIR filter with taps at: T0-6 = 0, 955, 1699, 1867, 1987, 3355, 3821 samples. The corresponding coefficients are C0-6 = 0.9804, 1.0, 0.802, 0.6225, 0.7049, 0.2618, and 0.2373.

A 19-point FIR filter with taps at: T0-18 = 0, 206, 1032, 1080, 1286, 1296, 1430, 2198, 2328, 2746, 2818, 2856, 2938, 3394, 3398, 3485, 3557, 3614, and 3826 samples. The corresponding coefficients are C0-18 = 1.0, 0.841, 0.504, 0.491, 0.379, 0.380, 0.346, 0.289, 0.272, 0.192, 0.193, 0.217, 0.181, 0.180, 0.181, 0.176, 0.142, 0.167, and 0.134.

By scaling these taps and coefficients, different effects can be created. By decreasing (or increasing) the coefficients, the liveness of the room can be increased (or decreased). Scaling the tap locations will increase/decrease the length of Early Reflections. Delaying the taps (offsetting all taps by the same amount of time) causes a delay in the Early Reflections.

A brief description of each surround effect mode is provided below. Note that the modes have been designed for Fs=48 KHz. When used for Fs=44.1 KHz, the change is largely inaudible and acceptable in most cases. If necessary, the user may scale the delay and reverb time parameters (if used) appropriately.

The effects can be broken into two categories, reverberative and non-reverberative. Reverberative effects have both late and early reflections, while non-reverberative effects are composed entirely of early reflections. The next two sections list the different configuration examples sent with the code. The examples are categorized by whether the effect is reverberative or not. Non-reverberative effects are much more subtle, while reverberative effects are more extreme and can sometimes interfere with the actual source material. The naming of each mode is very subjective and is only intended as a suggestion.

Section 10.3.1 "COEFMAKR.EXE and EFFECTS.EXE" on page 120 describes a program that can create effects configuration files for the user.

10.2.4 Non - Reverberative Effects

Non-reverberative effects provide a subtle effect through remapping and/or redirecting the stereo input. In this way, the end user can fill all of the speakers in a surround system without significantly altering the content of the source. Delays are used in some modes to simulate a larger listening area.

10.2.4.1 Passthru

This mode does not engage the Effects Manager and thus has no reverberation or Early Reflections. The default setup of Output Channel Mixer is used, i.e.,

L_Out = L

R_Out = R

C_Out = Ls_Out = Rs_Out = LFE_Out = 0.5 L + 0.5 R

10.2.4.2 Mono

This mode has no reverberation or Early Reflections and only uses the Output Channel Mixer. The Output Channel Mixer is setup for:

L_Out = R_Out = C_Out = Ls_Out = Rs_Out =0.5 L + 0.5 R

LFE_Out = 0

10.2.4.3 Classical

This mode has no reverberation and only uses FIR and the Output Channel Mixer. The mono mix is set up as (L+R)/2 with no LPF contribution and feeds FIR, a 5-tap filter.

The Output Channel Mixer is setup for:

L_Out = L

R_Out = R

C_Out = Ls_Out = Rs_Out = FIR

LFE_Out = 0

10.2.4.4 Panorama

This mode has no reverberation and only uses FIR and the Output Channel Mixer. The mono mix is set up as (L-R)/2 with no LPF contribution and feeds FIR, the same 5-tap filter as in Classical mode.

The Output Channel Mixer is setup for:

L_Out = L - 0.31623 R

R_Out = R - 0.31623 L

C_Out = 0.5 L + 0.5 R

Ls_Out = Rs_Out = FIR

LFE_Out = 0

10.2.4.5 Movie 1-4

These modes have no reverberation and only use FIR and the Output Channel Mixer. Each mode has a different FIR delay tap/coefficient setting, but all have the following generic setup. The mono mix is set up as (L+R)/2 with no LPF contribution and feeds FIR, a 4 or 5-tap filter depending on the mode.

The Output Channel Mixer is setup for:

L_Out = L

R_Out = R

C_Out = Ls_Out = Rs_Out = FIR

LFE_Out = 0

10.2.4.6 Music 1-4

These modes are identical to Movie1-4 modes above, except that the Output Channel Mixer is set up for C_Out = 0.

10.2.5 Reverberative Effects

Reverberative effects combine the different filters discussed in sections Section 10.1.2 "Basic Crystal Surround Effects Explanation" on page 108, and in Section 10.2.2 "Creating Custom Effects Using Advanced Settings" on page 114 to create a more obvious effect. These effects can be used to fill out the incoming source if so desired. Care should be taken as some reverberative effects can be so strong that they can interfere with the original stereo source signal. The effects listed below start with the subtle effects (Room) and graduate to the very obvious (Stadium).

To simplify the use of the following configuration files, the following naming convention has been used:

t**Er**G**a**K**i**N.cfg

E - This number represents the FIR filter type used for Early Reflections. Type 1 is a 7-Tap, and Type 2 is a 19-Tap, per the discussion in the previous

section. The 19-Tap filter will make the Early Reflections sound much thicker. This is useful when trying to create an echo pattern that sounds thick.

G - This represents the amount of Initial Delay and Early Reflection time. A shorter Early Reflection time will make the room size sound smaller, and a longer time will make the room sound larger. Initial Delay until Early Reflections will cause the room to sound as if the source is further from the walls.

G == 1: Early Reflections last for 40 ms, and the Initial Delay until Early Reflections is 20 ms.

G == 2: Early Reflections last for 80 ms (default), and the Initial Delay until Early Reflections is 10 ms.

G == 3: Early Reflections last for 115 ms, and there is no Initial Delay until Early Reflections (0 ms).

K - This represents the amount of time for After Reflections. The After Reflections produced by Reverb Module will produce a dense tail to the effect, and this parameter will influence the length of that tail.

K == 1: Reverb time is set to 0.25 ms.

K == 2: Reverb time is set to 0.5 ms(default).

K == 3: Reverb time is set to 0.8 ms.

K == 4: Reverb time is set to 0.98 ms. This will produce a reverb time that is very large.

Note: Only one example with K=4 is presented here (Stadium).

N - This represents the After Reflection (Reverb) Liveness. To absorb more of the high frequency content of the After Reflections (to sound less bright), the number should be higher. To increase the high frequency content (to sound more bright), the number should be lower.

For N == 1:Reverb Liveness (After Reflection Liveness) is set to 0.2

For N == 2:Reverb Liveness (After Reflection Liveness) is set to 0.8

To simplify the use of the filters, a description of some of the filters follows. Note that configuration files are supplied for many more variations of the parameters specified above. The supplied configuration files are intended to span a range of effects that can be created using the coefmakr.exe program. To create intermediate sounding effects, one could simply interpolate the parameters specified to the coefmakr.exe program and use the corresponding configuration files. Section 10.3.1 "COEFMAKR.EXE and EFFECTS.EXE" on page 120 has more details.

For all Reverberative Modes, the following mixlevels (default in coefmakr.exe) are used:

L_Mono_Mixlevel = 0.5

R_Mono_Mixlevel = 0.5

Direct_FIR_Mixlevel = 0.875

LPF_FIR_Mixlevel = 0.125

Reverb_Direct_Mixlevel = 1

Reverb_FIR_Mixlevel = 0

L_to_L_Out = 0.875

R_to_L_Out = 0

L_to_C_Out = 0.4375

R_to_C_Out = 0.4375

L_to_R_Out = 0

R_to_R_Out = 0.875

L_to_Ls_Out = 0.125

R_to_Ls_Out = 0.125

L_to_Rs_Out = 0.125

R_to_Rs_Out = 0.125

L_to_LFE_Out = 0.5

R_to_LFE_Out = 0.5

FIR_to_L_Out = 0.0833

Reverb_to_L_Out = 0.0417

FIR_to_C_Out = 0.0417

Reverb_to_C_Out = 0.0213

FIR_to_R_Out = 0.0833

Reverb_to_R_Out = 0.0417

FIR_to_Ls_Out = 0.5

Reverb_to_Ls_Out = 0.25

FIR_to_Rs_Out = 0.5

Reverb_to_Rs_Out = 0.25

10.2.5.1 t1r1a1i1.cfg - Room 1

This effect mode simulates the reflections from a small room. The room is not live, and has small Early and Late Reflection Times. The type 1 filter is used, providing a very mild effect.

10.2.5.2 t2r3a1i2.cfg - Room 2

This effect mode simulates the reflections from a small room, but uses a larger FIR filter spread out over a larger time to thicken the sound of the Early Reflections.

10.2.5.3 t1r1a2i1.cfg - Theater 1

This effect mode simulates the reflections as in a medium size theater. The Early Reflections will be short, but very present. The Late Reflections are bright, but are present for medium amount of time.

10.2.5.4 t2r3a2i2.cfg -- Theater 2

The Early Reflections in this effect are dominant and long, allowing for a very thick sounding Early Reflection. This effect mode has a short Reverb Time but large Reverb Liveness, making the sound less bright.

10.2.5.5 t1r1a3i1.cfg -- Hall 1

This effect has a lot of Early and Late Reflections, with a large Reverb Time, but a small Reverb Liveness. This effect will have a very large sound,

and the high frequencies will be present in the After Reflections.

10.2.5.6 t2r3a3i2.cfg -- Hall 2

This effect, similar to Hall1, has the same amount of Early and Late Reflections. In this effect mode, the higher frequencies are less present, resulting in a less bright sound in the reflections.

10.2.5.7 t2r3a4i2.cfg -- Stadium

This effect, utilizing a maximum of Early and Late Reflections, has a long decay time. The sound is highly decorrelate, making the echoes sound muddy. The long decay time provides for an effect of the sound bouncing off many hard walls, and a very bright tone is used. This effect is presented merely to demonstrate the extreme capability of the Crystal scheme, and may require modification before use in a real system that you develop.

10.3 Effects Configuration Files

COEFMAKR.EXE is a program written to assist development of unique effects configurations. It is documented in Section 10.3.1 "COEFMAKR.EXE and EFFECTS.EXE" on page 120. This program can be used to create configuration files for the effects modes presented in the previous sections. The configuration files have been omitted from this document to conserve space. The COEFMAKR.EXE program along with the configuration files are in a zip file available at www.cirrus.com under the CS49300 page. The configuration file contains paired 3-byte words which are the actual commands to be sent to the CS49300. These messages correlate with the parameters in Table 42. In addition comment lines marked by a # symbol are included to help explain the configuration messages. When porting the messages to code for a host microcontroller, the comment lines should be omitted and care should be taken to keep the paired 3- byte words together in the correct order. The configuration files also contain a "Kickstart" as the last command (see Section 4.1 "Audio Manager for Dolby Digital, DTS, MP3, MPEG Multichannel, AAC and Crystal

for more
information).

10.3.1 COEFMAKR.EXE and EFFECTS.EXE

COEFMAKR.EXE (Coefficient Maker) is a simple, easy-to-use program for generating configuration files used in conjunction with Crystal Effects code for the CS49300 series of Surround Sound DSP's. The program takes all of the configuration options on the command line, and all of these options can be displayed with the -? command line switch. Note that since there are 56 command line options (a minimum of 1 must be specified), to display Mix Level options, the -x? switch must be used. For all Advanced Options, the -o? switch must be used.

EFFECTS.EXE is a GUI version of COEFMAKR.EXE which allows the user to edit effects configuration files and try them on one of the CRD49300/CDB49300 application boards in real time. The program mirrors the block diagrams shown in and allows the user control over the parameters discussed in section.

Here is a short description of the files and their contents:

- *COEFMAKR.EXE*: This file is the main program for the coefficient maker.

- *FIR.CFG*: This file contains the templates for 2 basic FIR (Early Reflection) filters, which are scaled by COEFMAKR.EXE into the FIR section of the effects code. Additional filters can be added to this file, or the file can be replaced by one of the same structure. However, if a different file is used, it must be specified in the command line (-c option). For COEFMAKR.EXE to run correctly, this file must reside in the same directory.

- *MAKETEST.BAT*: This batch file will generate all of the default configuration files shown in section (with reverb) using COEFMAKR.EXE.

- *EFFECTS.EXE*: GUI version of COEFMAKR.EXE

More files may need to be present. Please read the README.TXT file in the zip file from which the programs were extracted for more information on these programs.

10.3.2 Command Line Options for COEFMAKR.EXE

Basic Options Settings:

Usage: coefmakr [-qfile.out -cfilter.cfg -fx -dy -tz -rw -lb -mu -v -xelevel]

    -qfile.out : file.out = Config file to be generated(default=file.out)

    -cfilter.cfg : filter.cfg = Filter parameter file(default=fir.cfg)

    -fb : b = Sampling Rate (32, 44, 48* KHz)

    -du : u = Initial Delay of Early Reflections (0*,1,...,115ms)

    -te : e = FIR Filter Type (1* or 2)

    -rg : g = Early Reflection Time (0,1,...,80*,... ms)

    -lh : h = Liveness of Early Reflection (0,...,1*)

    -mj : j = Delay Until After Reflections (0,1,...,100*,...ms)

    -ak : k = After Reflection Time (0,...,0.5*,...,1)

    -in : n = After Reflection Liveness (0,...,1*)

    -xp : p = Mix Level(s) to Change (? for options)

    -os : s = Advanced Options (? for options)

    -v = Run in non-verbose mode

-x Options:

Settings for Mix level:

a = L_Mono_Mixlevel(-1..0.5*..1)

b = R_Mono_Mixlevel(-1..0.5*..1)

c = Direct_FIR_Mixlevel(-1..0.875*..1)

d = LPF_FIR_Mixlevel(-1..0.125*..1)

e = Reverb_Direct_Mixlevel(-1..0.875*..1)

f = Reverb_FIR_Mixlevel(-1..0*..1)

g = L_to_L_Out(-1..0.875*..1)

h = R_to_L_Out(-1..0*..1)

i = L_to_C_Out(-1..0.4375*..1)

j = R_to_C_Out(-1..0.4375*..1)

k = L_to_R_Out(-1..0*..1)

l = R_to_R_Out(-1..0.875*..1)

m = L_to_Ls_Out(-1..0.125*..1)

n = R_to_Ls_Out(-1..0.125*..1)

o = L_to_Rs_Out(-1..0.125*..1)

p = R_to_Rs_Out(-1..0.125*..1)

q = L_to_LFE_Out(-1..0.5*..1)

r = R_to_LFE_Out(-1..0.5*..1)

s = FIR_to_L_Out(-1..0.0625*..1)

t = Reverb_to_L_Out(-1..0.0625*..1)

u = FIR_to_C_Out(-1..0.03125*..1)

v = Reverb_to_C_Out(-1..0.03125*..1)

w = FIR_to_R_Out(-1..0.0625*..1)

x = Reverb_to_R_Out(-1..0.0625*..1)

y = FIR_to_Ls_Out(-1..0.375*..1)

z = Reverb_to_Ls_Out(-1..0.375*..1)

1 = FIR_to_Rs_Out(-1..0.375*..1)

2 = Reverb_to_Rs_Out(-1..0.375*..1)

-o Options:

Usage of Advanced Options Settings:

 a = Comb_0_Delay_Size(1......1103*)

 b = Comb_1_Delay_Size(1......1213*)

 c = Comb_2_Delay_Size(1......1321*)

 d = Comb_3_Delay_Size(1......1433*)

 e = Comb_4_Delay_Size(1......1543*)

 f = Comb_5_Delay_Size(1......1657*)

 g = Comb_0_g1(0...0.0895*...1)

 h = Comb_1_g1(0...0.0950*...1)

 i = Comb_2_g1(0...0.1002*...1)

j = Comb_3_g1(0...0.1055*...1)

k = Comb_4_g1(0...0.1106*...1)

l = Comb_5_g1(0...0.1160*...1)

m = Comb_0_Mixlevel(0...0.9*...1)

n = Comb_1_Mixlevel(0...0.9*...1)

o = Comb_2_Mixlevel(0...0.9*...1)

p = Comb_3_Mixlevel(0...0.9*...1)

q = Comb_4_Mixlevel(0...0.9*...1)

r = Comb_5_Mixlevel(0...0.9*...1)

s = Allpass_g1(0...0.7*...1)

t = Allpass_g2(0...0.51*...1)

The following are the command line parameters to use with the COEFMAKR.EXE program to create the configuration files for the effects modes presented in through .

coefmakr -qt1r1a1i1.cfg -r40 -d20 -a0.25 -v

coefmakr -qt2r3a1i2.cfg -t2 -r115 -i0.6 -v

coefmakr -qt1r1a2i1.cfg -r40 -d20 -v

coefmakr -qt2r3a2i2.cfg -t2 -r115 -i0.6 -v

coefmakr -qt1r1a3i1.cfg -r40 -d20 -a0.8 -v

coefmakr -qt2r3a3i2.cfg -t2 -r115 -a0.8 -i0.6 -v

coefmakr -qt2r3a4i2.cfg -r115 -l0.5 -m120 -a0.97 -i0.8 -v

SMART

Analog ™