Chipsmall Limited consists of a professional team with an average of over 10 year of expertise in the distribution of electronic components. Based in Hongkong, we have already established firm and mutual-benefit business relationships with customers from,Europe,America and south Asia,supplying obsolete and hard-to-find components to meet their specific needs.

With the principle of “Quality Parts,Customers Priority,Honest Operation,and Considerate Service”,our business mainly focus on the distribution of electronic components. Line cards we deal with include Microchip,ALPS,ROHM,Xilinx,Pulse,ON,Everlight and Freescale. Main products comprise IC,Modules,Potentiometer,IC Socket,Relay,Connector.Our parts cover such applications as commercial,industrial, and automotives areas.

We are looking forward to setting up business relationship with you and hope to provide you with the best service and solution. Let us make a better world for our industry!

Contact us

Tel: +86-755-8981 8866 Fax: +86-755-8427 6832
Email & Skype: info@chipsmall.com Web: www.chipsmall.com
Address: A1208, Overseas Decoration Building, #122 Zhenhua RD., Futian, Shenzhen, China

6+2-Ch. multistandard audio decoder

Technical Literature

CUSTOM ATTRIBUTES

Alternate Identifier(s)	9622
Key process	Product Development
ISO Definition	Specification
Confidentiality Level	Public
Document Type	Technical Literature
Document Category	Datasheet
Document Family	
Original ID	
Original Repository	
Status	IN APPROVAL
Responsible	
Keywords	Technical Literature, 9622, Product Development, Specification, Datasheet, STA310,

DOCUMENT HISTORY

Version	Release Date	Change Qualifier
Rev 1.3	11-Jun-2013	Properties Changes
AUTOMATIC OBSOLETE RPN WORKFLOW STARTED		

DOCUMENT APPROVAL

User	Function	Date
Camilleri Evelina	Document Controller	12-Jun-2013

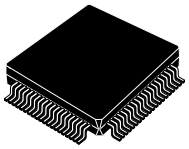
STA310

6+2-CH. MULTISTANDARD AUDIO DECODER

PRELYMINARY DATA

1 FEATURES

- ***DVD Audio*** **decoder:**
  - ***Meridian Lossless Packing*** (***MLP***), with up to 6 channels,
  - Uncompressed LPCM with 1-8 channels,
  - Precision of up to 24 bits and sample rates of between 44.1 kHz and 192 kHz.
- ***Dolby Digital*** (*) **decoder:**
  - Decodes 5.1 *Dolby Digital Surround.*
  - Output up to 6 channels. downmix modes: 1, 2, 3 or 4 channels.
- **MPEG** -1 2- channel audio decoder, layers I and II.
- **MPEG-2 6-channel audio decoder, layer II.**
  - 24 bits decoding precision.
- **MP3 (MPEG layer III) decoder.**
- Accepts MPEG-2 PES stream format for: MPEG-2, MPEG-1, Dolby Digital and linear PCM.
- **Karaoke System.**
- **Prologic decoder.**
- **Downmix for Dolby Prologic compatible.**
  - A separate (2-ch) PCM output available for simultaneous playing and recording.
- Bitstream input interface: serial, parallel or SPDIF.
- SPDIF and IEC-61937 input interface.
- SPDIF and IEC-61937 output interface.
- PLL for internal PCM clock generation. frequencies supported: 44.1KHz family (22.05, 88.2, 176.4) and 48KHz family (24, 48, 96, 192).
- PCM: transparent, downsampling 192 to 96 Khz and 96 to 48kHz.
- PTS handling control on-chip.
- **No external DRAM required**
- I^2C or parallel control bus
- Embedded *Development RAM* for customizable software capability.
- Configurable internal PLLs for system and audio clocks, from an externally provided clock.
- 80-PIN TQFP package



TQFP80
ORDERING NUMBER: STA310

- 2.5V (for core) and 3V (for I/O) power supply.
  - 3V Capable I/O Pads .
- True-SPDIF input receiver supporting AES/ EBU, IEC958, S/PDIF.
  - No external chip required.
  - Differential or single ended inputs can be decoded.

APPLICATIONS

- High-end audio equipment.
- DVD consumer players.
- Set top box.
- HDTV .
- Multimedia PC.

(*) **"Dolby ", "AC-3"** and **"ProLogic"** are trademarks of **Dolby Laboratories.**

DESCRIPTION

The STA310 is a fully integrated Audio Decoder capable of decoding all the above listed formats.

Encoded input data can be entered either by a serial (I2S or SPDIF) or a parallel interface. A second input data stream (I2S) is available for micro input.

The control interface can be either **I^2C** or a parallel 8-bit interface. No external DRAM is necessary for a total of 35ms surround delays.

2 STA310 AUDIO DECODER PIN DESCRIPTION

Pin Number	Name	Type	Function
CONTROL INTERFACES			
48	IRQB	O (1)	Interrupt Signal (level), active low
47	SELI2C	I (2)	Selects the Control Interface (when high: serial interface; when low: parallel interface)
I ^{2}C Control Interface			
43	SDAI2C	I/O (1)	I ^{2}C Serial Data
46	SCLKI2C	I	I ^{2}C Clock
53	MAINI2CADR	I (2)	Determines the slave address
Parallel Control Interface			
78 - 79 - 80 - 1 2 - 3 - 6 - 7	D0 - D1 - D2 - D3 D4 - D5 - D6 - D7	I/O	Host Data
12 - 13 - 14 - 15 16 - 18 - 19 - 20	A0 - A1 - A2 - A3 A4 - A5 - A6 - A7	I	Host Address
21	DCSB	I	Chip Select, active low
22	R/$\overline{W}$	I	Read/Write Selection: read access when high, write access when low
35	WAITB	O (3)	Data Acknowledge, active low
DATA INPUT INTERFACE			
First Serial Data Interface (I^{2}S)			
37	DSTRB	I	Clock Input Data, active low
41	SIN	I	Serial Input Data
40	LRCLKIN	I	Word Clock for the Input
42	REQ	O	Handshake for the Data Transfer, aconfigurable by the SIN_SETUP register
Second Serial Data Interface (I^{2}S)			
62	DSTRB2	I	Clock Input Data, active low
60	SIN2	I	Serial Input Data
61	LRCLKIN2	I	Word Clock for the Input
63	REQ2	O	Handshake for the Data Transfer, active low
DATA OUTPUT INTERFACES			
69	PCMCLK	I/O	Oversampling Clock input for STA310 when generated externally
DAC Interface			
67	SCLK	O	Bit Clock for the DAC

2 STA310 AUDIO DECODER PIN DESCRIPTION (continued)

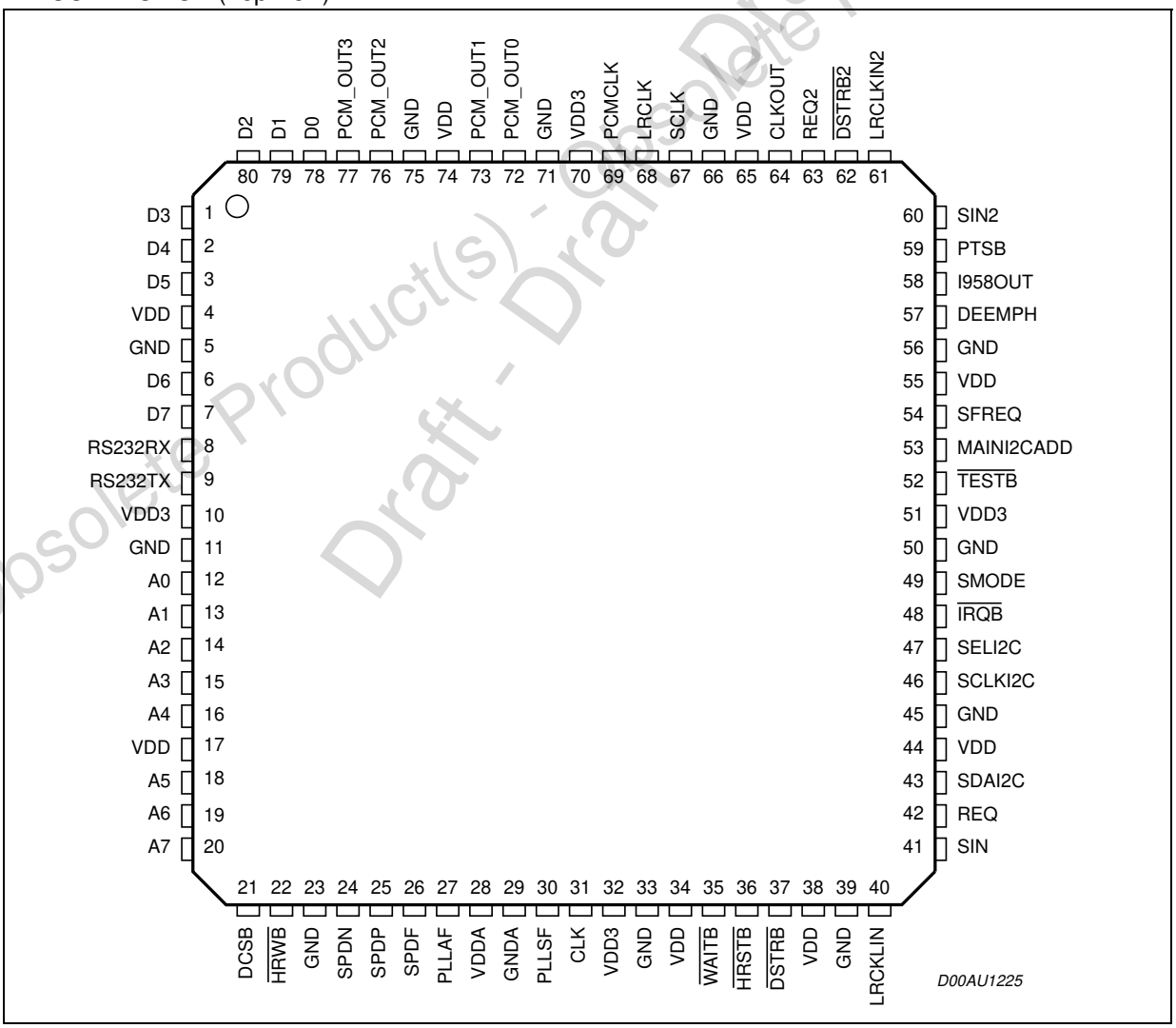
Pin Number	Name	Type	Function
68	LRCLK	O	Word Clock for the DAC
72	PCM_OUT0	O	Data from a Prologic downmix (VCR_L/VCR_R)
73	PCM_OUT1	O	Data for the first DAC (Left/Right)
76	PCM_OUT2	O	Data for the second DAC (Centre/Sub)
77	PCM_OUT3	O	Data for the third DAC (LeftSur/RightSur)
IEC958 Interface (S/PDIF) - One Output Port., One Input Ports.			
58	I958OUT	O	S/PDIF Signal
25	SPDP	I	First differential input of S/P DIF port
24	SPDN	I	Second differential input of S/P DIF port
26	SPDF	I	External Filter
28	VDDA	I	Analog VDD for S/P DIF Input port
29	GNDA	I	Analog GND for S/P DIF Input port
STATUS INFORMATION			
PCM Related Information			
54	SFREQ	O	Then high, indicates that the sampling freq. is either 44.1Khz or 22.05Khz. When low, indicates that the sampling frequency is either 32 Khz, 48 Khz, 24 Khz or 16Khz.
57	DEEMPH	O	Indicates if de-emphasis is performed.
Audio Video Synchronization			
59	PTSB	O	Indicates that a PTS has been detected, active low.
Other Signals			
31	CLK	I	Master Clock Input Signal.
36	RESET	I(2)	Reset signal input, active low.
52	TESTB	I(2)	Reserved pin: to be connected to VDD
49	SMODE	I	Reserved pin : to be connected to GND
RS232 Interface			
8	RS232RX	I	
9	RS232TX	O	
PLLs INTERFACES			
64	CLKOUT	O	System clock output with programmable division ratio
27	PLLAF	I	External Filter For Audio PLL.

2 STA310 AUDIO DECODER PIN DESCRIPTION (continued)

Pin Number	Name	Type	Function
30	PLLSF	I	External Filter For System PLL.
Power and Ground			
5 - 11 - 23 - 33 - 39 - 45 - 50 - 56 - 66 - 71 - 75	GND	GND	Ground
4 - 17 - 34 - 38 44 - 55 - 65 - 74	VDD	VDD	2.5V Power Supply
10 - 32 - 51 - 70	VDD3	VDD3	3.3V Power Supply

Notes (1) Open Drain
(2) Internal Pull-up
(3) Tri-State

PIN CONNECTION (Top view)



ABSOLUTE MAXIMUM RATINGS

Symbol	Parameters	Value	Unit
Vdd	2.5V Power Supply Voltage	-0.5 to 3.3	V
	2.5V Input or Output Voltage	-0.5 to (Vdd+0.5)	V
Vdd3	3.3V Power Supply Voltage	-0.5 to 4	V
	3.3V Input or Output Voltage	-0.5 to (Vdd+0.5)	V

ELECTRICAL CHARACTERISTICS (V_{DD} = 3.3V +/-0.3V; T_{amb} = 0 to 70°C; Rg = 50 Ω unless otherwise specified

DC OPERATING CONDITIONS			
Symbol	**Parameters**	**Value**	**Unit**
Vcc	Power Supply Voltage	2.5	V
Tj	Operating Junction Temperature	-20 to 125	°C

GENERAL INTERFACE

Symbol	Parameters	Conditions	Min	Typ	Max	Unit	Note
I_{il}	Low level input current without pull-up device	Vi = 0V			1	µA	1
I_{ih}	High level input current without pull-down device	Vi = Vdd			1	µA	1
I_{oz}	Tri-state output leakage without pull-up/down device	Vi = 0V or Vdd			1	µA	1
$I_{latchup}$	I/O Latch-up current	V<0V, V>Vdd	200			mA	2
Vesd	Electrostatic protection	Leakage <1µA	2000			V	3

Note: 1. The leakage currents are generally very small, <1nA. The value given here, 1µA, is a maximum that can occur after an Electrostatic Stress on the pin.
2. V> Vdd3 for 3.3V buffers.
3. Human Body Model

LVTTL & LVCMOS DC Input Specification 2.7V <Vdd3 <3.6V

Symbol	Parameters	Conditions	Min	Typ	Max	Unit	Note
V_{il}	Low level input voltage				0.8	V	1
V_{ih}	High level input voltage		2.0			V	1
V_{ilhyst}	Low level threshold input falling		0.8		1.35	V	1
V_{ihhyst}	High level threshold input rising		1.3		2.0	V	1
V_{hyst}	Schmitt trigger hysteresis		0.3		0.8	V	1

Note: 1. Takes into account 200mV voltage drop in both supply lines.
2. X in the source/sink current under worst case conditions and is reflected in the name of the I/O cell according to the drive capability

ELECTRICAL CHARACTERISTICS (continued)

Symbol	Parameters	Conditions	Min	Typ	Max	Unit	Note
Ipu	Pull-up current	Vi = 0V		-66	0.8	µA	1
Rpu	Equivalent Pull-up resistance	Vi = 0V		50		KΩ	

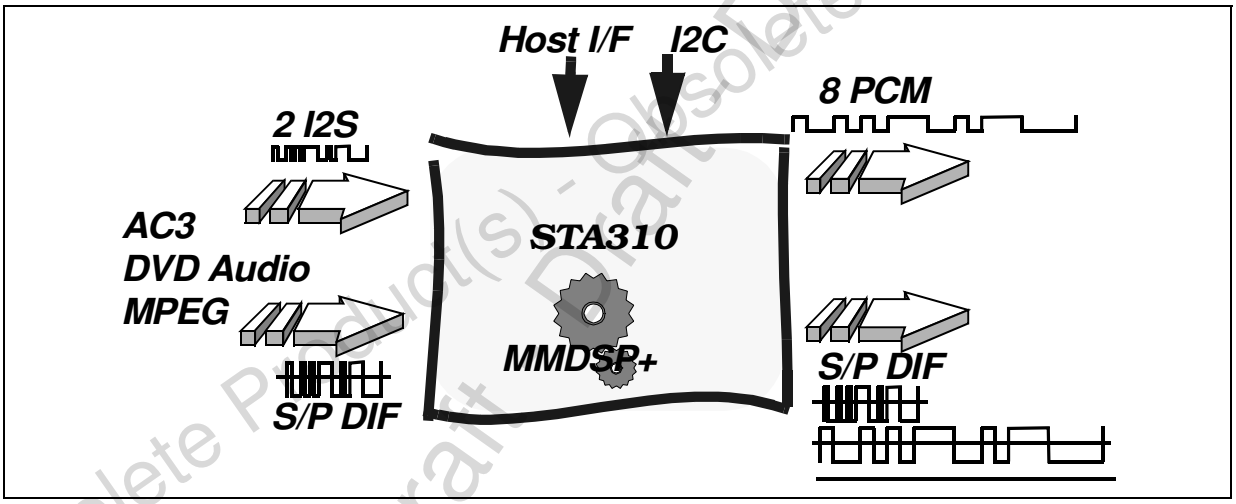
Note: 1. **Min condition**: V_{DD} = 2.7V, 125°C Min precess **Max condition:** V_{DD} = 3.6V, -20°C Max

POWER DISSIPATION

Symbol	Parameters	Conditions	Min	Typ	Max	Unit	Note
P_D	Power Dissipation @V_{DD} = 2.4V	Sampling frequecy ≤ 24KHz		t.b.d.		mW	1
		Sampling frequecy ≤ 32KHz		t.b.d.		mW	1
		Sampling frequecy ≤ 48KHz		t.b.d.		mW	1

INTRODUCTION

The STA310 is a fully integrated multi-format audio decoder. It accepts as input, audio data streams coded with all the formats listed above.



2.1 Inputs and Outputs

2.1.1 Data Inputs

- Through a parallel interface (shared with the control interface)
- Through a serial interface (for all the **I^2S** formats)
- Through a S/P DIF (**SPDIF** or **IEC-61937** standards).

- Trough a second, independent,**I^2S** (for application like i..e. Karaoke mixing).

2.1.2 Data outputs

- The PCM audio ooutput interface, which provide:
  PCM data on 4 outputs:
  - Left/Right,
  - Centre/Subwoofer
  - Left Surround/Right Surround.

• Data From a Prologic downmix (encoder)
"Lrclk" "Sclk" "PcmClk"

- S/P DIF Output

2.1.3 Control I/F

I2C slave or parallel interface:

The device configuration and the command issuing is done via this interface. To facilitate the contact with the MCU, 2 interrupt lines (IRQB and INTLINE) are available.

3 ARCHITECTURE OVERVIEW

3.1 Data flow

The STA310 is based on a programmable MMDSP+ core optimized for audio decoding algorithms.

Dedicated hardware has been added to perform specific operations such as bitstream depacking or IEC data formatting.

The arrows in Figure 3 indicate the data flow within the chip.

The compressed bitstream is input via the data input interface.

Data are transferred on a byte basis to the FIFO. This FIFO allows burst input data at up to 33Mbit/s.

The input processor, which is composed of a packet parser and an audio parser, unpacks the bitstream (Packet parser) and verifies the syntax of the incoming stream (audio parser).

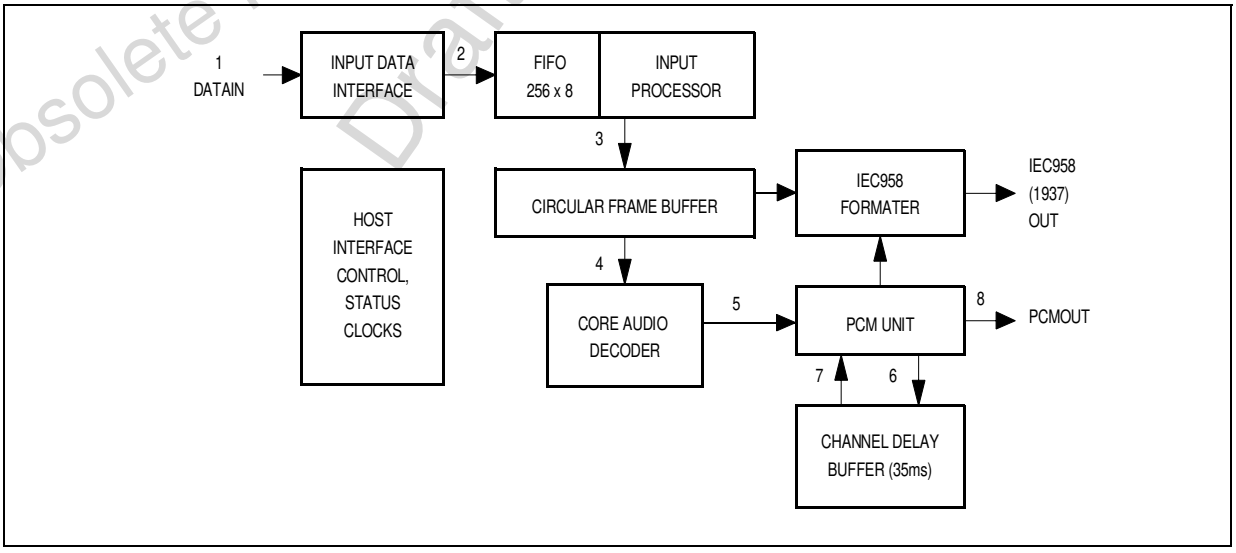
The compressed audio frames with their associated information (PTS) are stored into the circular frame buffer.

While a second frame is stored in the circular frame buffer, the first frame is extracted by the audio core decoder which decodes it to produce audio samples.

The PCM unit converts the samples to the PCM format. The PCM unit controls also the channel delay buffer in order to delay each channel independently.

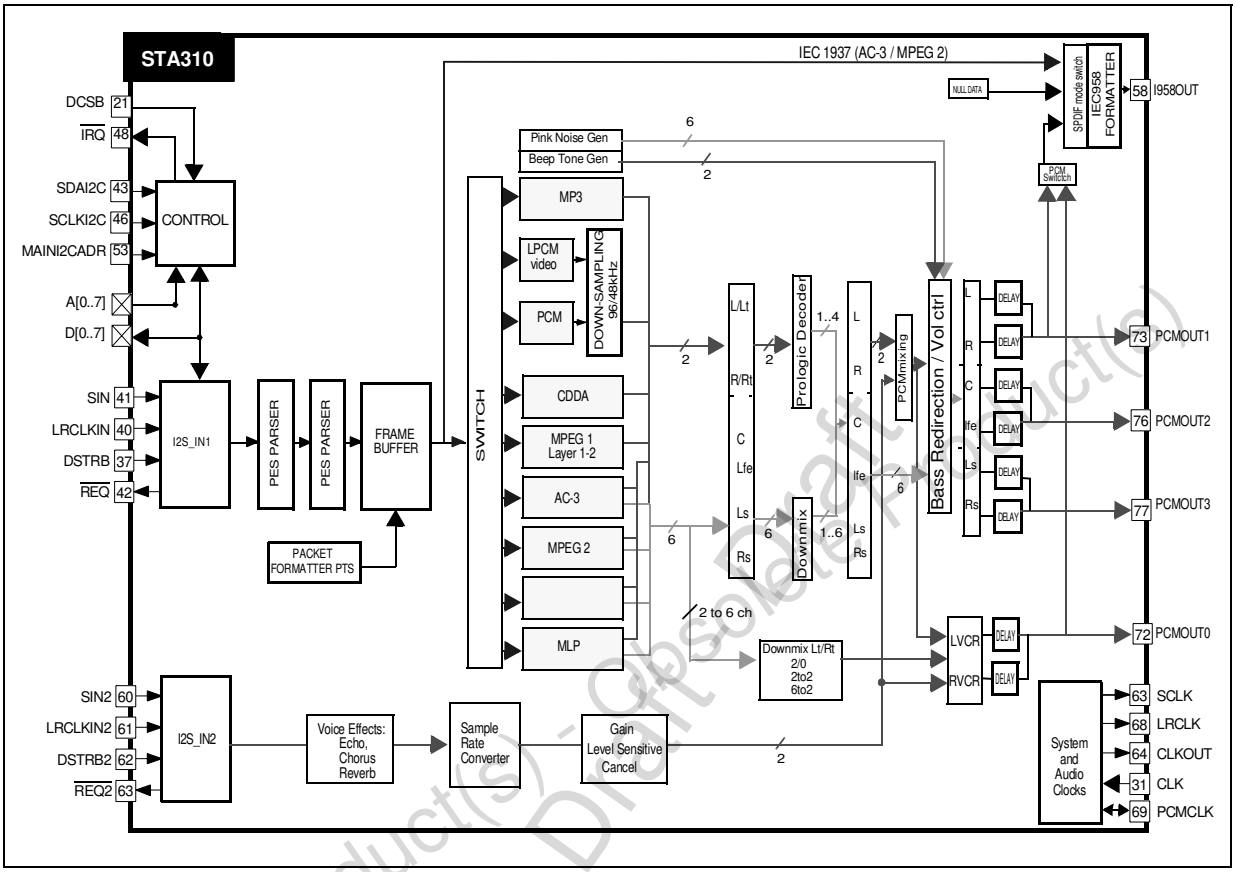
In parallel, the IEC unit transmits non compressed data or compressed data according to the selected mode. In the compressed mode, the data are extracted directly from the circular buffer and formatted according to the IEC-61937 standard. In non compressed mode, the left and right PCM channels formatted by the PCM unit are output by the IEC unit, according to the SPDIF standard

Figure 1. Architecture and data flows



3.2 Functional diagram

Figure 2. Audio decoder top level functional diagram



3.3 Control interface description

The IC can be controlled either by a host using an I²C interface, or by a general purpose host interface.

These interfaces provide the same functions and are described in the following sections. The selection is performed by the means of the pin SELI2C: when high, this pin indicates that the I²C interface is used. When low, the parallel interface is used.

3.3.1 Parallel control interface

When the pin SELI2C is low, the control of the chip is performed through the parallel interface. When accessing the device through the parallel interface, the following signals are used:

- The address bus A[7..0]. It is used to select one of the 256 register locations.
- The data bus DATA[7..0]. If a read cycle is requested, the data lines D[7:0] will be driven by the IC. For a write cycle, the STA310 will latch the data placed on the data lines when the $\overline{WAIT}$ signal is driven high.
- The signal R/$\overline{W}$. It defines the type of register access: either read (when high), or write (when low). Some registers can be either written or read, some are read only, some are write only.
- The signal $\overline{DCSB}$. A cycle is defined by the assertion of the signal $\overline{DCSB}$.

Note: 1. The address bus A[7..0], and read/write signal R/$\overline{W}$ must be setup before the $\overline{DCSB}$ line is activated.

- The signal $\overline{\text{WAIT}}$. This signal is always driven low in response to the $\overline{\text{DCSB}}$ assertion.

The timing diagrams for the parallel control interface are given in *Electrical specifications* on page 5.

3.4 I²C control interface

When the pin SELI2C is high, the chip is controlled through the I²C interface. The I²C unit works at up to 400kHz in slave mode with 7-bit addressing.

- The Pin MAINI2CADR selects the device address. When MAINI2CADR is high the slave address is 0x5C, when low the device address is equal to the value on the address bus (A0...A6).
- The pin SDAI2C is the serial data line.
- The pin SCLKI2C is the serial clock.

The I²C Bus standard does not specify sub-addressing. There are thus potentially multiple ways to implement it. Any implementation that respects the standard is of course legal but a particular implementation is used by many companies. The following paragraphs describe this implementation.

3.4.1 Protocol description

For write accesses only, the first data which follows the slave address is always the sub-address.

This is the one and only way to declare the sub-address. It should be noticed that the sub-address is implemented as a standard data on the I²C Bus protocol point of view. It is a sub-address because the slave knows that it must load its address pointer with the first data sent by the master.

See in the Appendix X.x for I^2C message format examples.

3.5 Decoding process

The decoding process in the STA310 is done in several stages:

- Parsing,
- Main decoding,
- Post decoding,
- Bass redirection,
- Volume and Balance control.

Each of the stages can be activated or bypassed according to the configuration registers.

Parsing

The bitstream parsing (performed by the input processor) is in charge of discarding all the non audio information in order to transmit to the next stage (the circular frame buffer) only the audio elementary stream (AC3, MPEG1/2, LPCM, PCM, DVD Audio).

The parsing stage operates in two phases: the packet parser unpacks the stream, the audio parser checks the syntax of the bitstream.

Main Decoding

The input of this stage is an elementary stream, the outputs are decoded samples. The number of output channels is defined by the downmix register (1 channel up to 6 channels). For details, please refer to the description of the register.

The decoding formats currently supported are AC3, MPEG1 layers I and II, MPEG2 layer II, LPCM. It is necessary to select the appropriate stream format by configuring the registers STREAMSEL and DECODESEL before running the decoder.

Post Decoding

The post decoding includes specific PCM processing: DC filter, de-emphasis filter, downsampling filter. These filters can be independently enabled or disabled through the register DWSMODE.

It provides also a Pro Logic decoder, which is described in detail in a next section.

Bass Redirection

This stage redirects the low frequency signals to the subwoofer.

The subwoofer is extracted from the other channels (L, R, C, Ls, Rs, LFe). There are six possible configurations to extract the subwoofer channel, which can be selected thanks to the OCFG register.

Volume and Balance Control

The volume is a master volume (no independent control for each channel). It is controlled by the PCMSCALE register, which enables to attenuate the signals by steps of 2dB.

Two balance controls are available: one for Left/Right channels, one for Left Surround/Right Surround channels. They are configurable by means of registers BAL_LR (Left-Right Balance) and BAL_SUR (Left Surround-Right Surround Balance), which provide attenuation of signals by steps of 0.5dB.

4 OPERATION

4.1 Reset

The STA310 can be reset either by a hardware reset or by a software reset:

- The hardware reset is sent when the pin $\overline{\text{RESET}}$ is activated low during at least 60ns. This is equivalent to a power-on reset.
  This resets all the configuration registers, i.e. PLL registers (PLLSYS, PLLPCM), Interrupt registers (INTE, INT, ERROR), interface registers (SIN_SETUP, CAN_SETUP) and command registers (SOFTRESET, RUN, PLAY, MUTE, SKIP_FRAME, REPEAT_FRAME).
- The software reset is sent when the register SOFTRESET is written to 1 (the register is automatically reset once the software reset is performed). It resets only the interrupt related registers (INTE, INT, ERROR) and the command registers (SOFTRESET, RUN, PLAY, MUTE, SKIP_FRAME, REPEAT_FRAME). All other decoding configurations are not changed by softreset.

Some information concerning the post-processing are anywayt of date after a soft-reset

Note: 1. The chip must be soft reset before changing any configuration register.

4.2 Clocks

There are two embedded PLLs in the STA310: the system PLL and the PCM PLL.

The following is the block diagram of the system and audio clocks used in the STA310

Figure 3. PLL Block Diagram

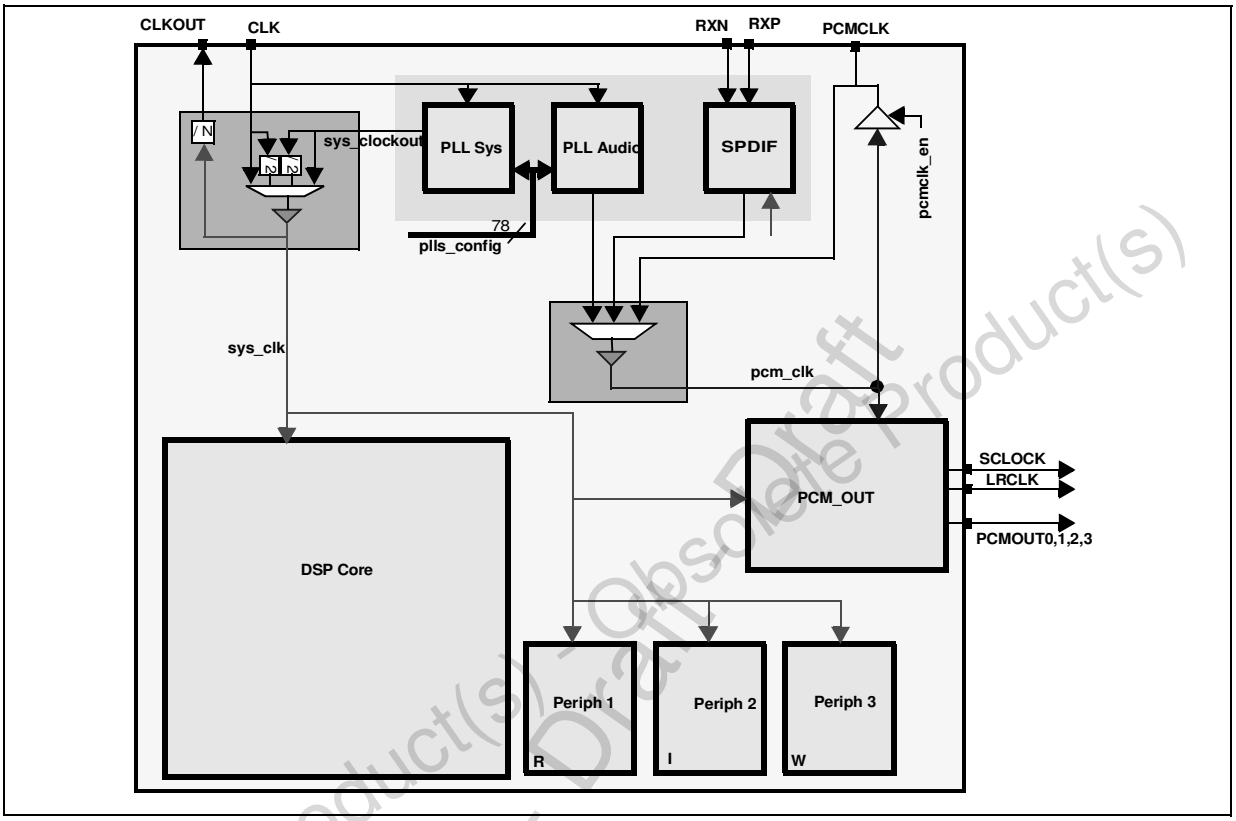
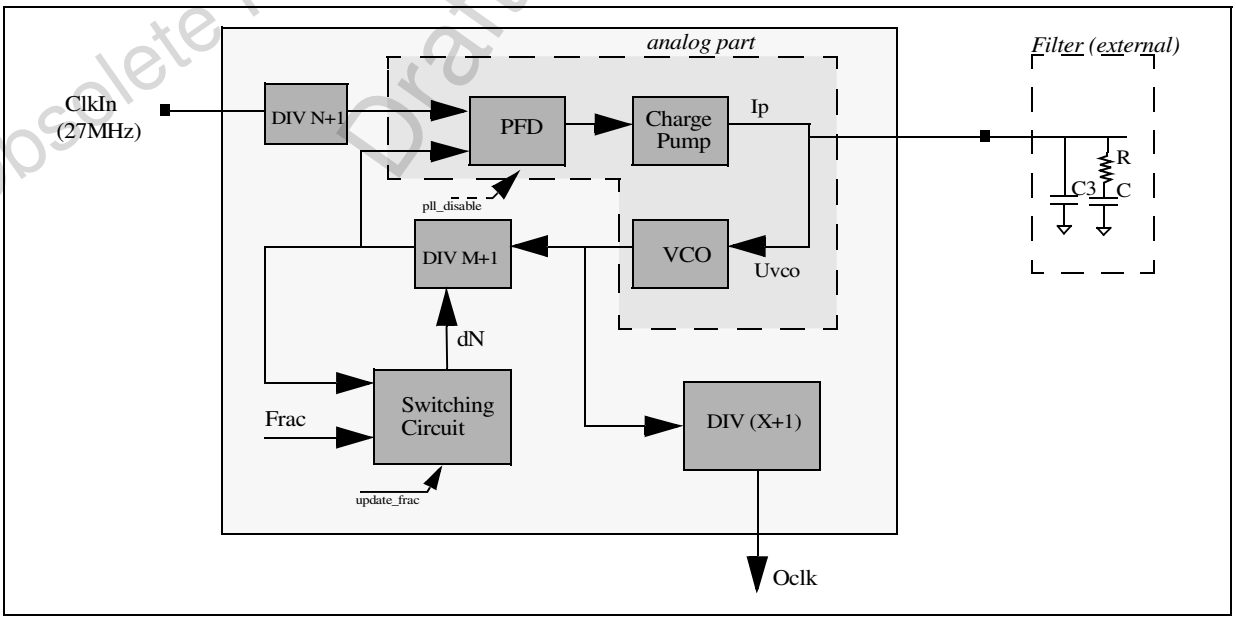


Figure 4. Block Diagram of Functional PLL



4.2.1 System clock

The system clock sent to the DSP core and the peripherals can be derived from 4 sources and the selection is performed through an Host Register; external clock, external clock divided by 2, internal system PLL and internal system PLL divided by 2.

The system PLL is used to create the system clock from the input clock. This PLL is software programmable through the Host Registers mechanism. The system PLL is used to set the any frequency up to the maximum allowed device speed. After hard reset the system clock is running at 47.25MHz. An RC network must be connected to the filter Pin PLLSF.

The system clock is output on the pin CLKOUT after a programmable divider ranging from 1 to 16.

4.2.2 DAC clocks

4.2.2.1 PCM clock

The PCM clock can be either input to the device or generated by the internal PLL or recovered by the embedded SPDIF receiver. The selection is done via the Host Registers.

After a hardware reset, the internal PLL is disabled and the PCMCLK pad is an input. PCMCLK may be equal to the PCM output bit rate, or it may be an integer multiple of this, allowing the use of oversampling D-A converters.

The internal fractional PLL is able to generate PCMCLK at any “FsX Oversampling Factor” frequencies, where Fs is any multiple or sub-multiple of the two 44.1kHz and 48kHz sampling frequencies. An RC network must be connected to the filter pin PLLAF; refer to External circuitry on page 9 for recommended values.

If the PCMCLK is recovered from the embedded SPDIF receiver, the only supported overampling frquency is 128 Fs.

4.2.2.2 Bit clock SCLK

The PCM serial clock SCLK is the bit clock. It provides clocks for each time slot (16 cycles for each channel in 16-bit mode, 32 cycles for each channel in 18-, 20-, 24-bit modes). The frequency of SCLK is therefore fixed to 2 x Nb time slots x Fs, where Fs is the sample frequency.

The clock is derived from the clock PCMCLK. The register PCMDIVIDER must be configured according to the selected output precision and the frequency of PCMCLK, so that the device can construct SCLK:

Fsclk = Fpcmclk / (2 x (PCMDIVIDER+1)) gives

Table 1.

PCM Divider Value	Mode Description
5	PCMCLK = 384 Fs, DAC is 16-bit mode
3	PCMLK = 256 Fs, DAC is 16-bit mode
2	PCMLK = 384 Fs, DAC is 32-bit mode
1	PCMLK = 256 Fs, DAC is 32-bit mode

The value of PCMDIVIDER = 0 is reserved. If this number is loaded, the divider is bypassed and the frequency of SCLK equals the frequency of PCMCLK. The PCMDIVIDER register must be setup before the output of SCLK starts.

This can be done by first disabling PCM outputs, by de-asserting the MUTE and PLAY commands and then writing into the PCMDIVIDER register. Once the register is setup, the MUTE and/or PLAY commands can be asserted. PCMDIVIDER can not be changed “on the fly”.

4.2.2.3 Word clock LRCLK

The frequency of LRCLK is given by:

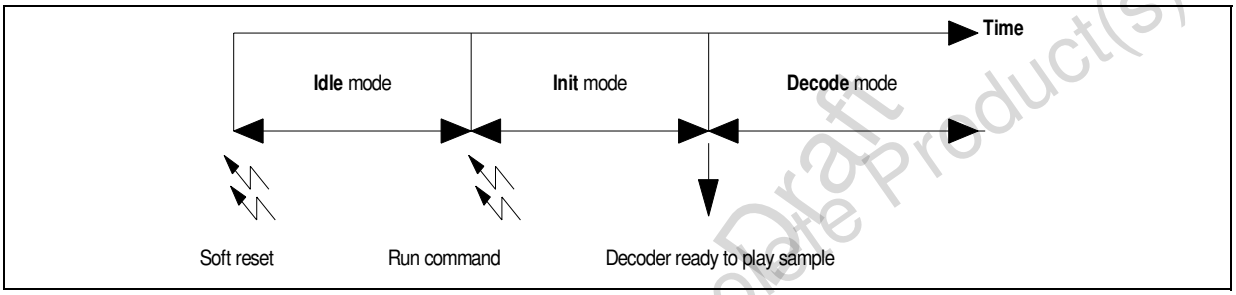
- Flrclk = Fsclk/32; for 16 bit PCM output,
- Flrclk = Fsclk/64; for 18, 20 or 24 bits PCM output.

No special configuration is required. The polarity can be changed in the register PCMCONF, by setting up the field INV as needed.

4.3 Decoding states

There are two different decoder states: Idle state and decode state (see <Blue HT>Figure 3). To change states, register

Figure 5. Decoding States



Idle Mode

This is the state entered after a hardware or software reset. In this state, the embedded DSP does not decode, i.e. no data are processed. The chip is waiting for the RUN command, and during this state all configuration registers must be initialized. In this state, even if the chip is not processing data, the DACs clocks can be output, which enables to setup the external DACs. Once the PCMCLK, SCLK and LRCLK clocks are configured, it is-possible to output them by setting the MUTE register.I

Table 2. Idle mode. play and mute commands effects

Play	Mute	Clock (SCLK, LRCLK) State	PCM Output
X	0	Not running	0
X	1	Running	0

Note: 1. The PLAY command has no effect in this state as the decoder is not running. It can however be sent and it will be taken into account as soon as the decoder enters the decode state.

Decode Mode

This state is entered after the RUN command has been sent (i.e. RUN register = 1). In this mode, the data are processed. The decoder can play sound, or mute the outputs, by using the PLAY and MUTE registers:

- To decode streams, the PLAY register must be set. When decoding, the sound will be sent to outputs if the MUTE register is reset. The outputs are muted if the MUTE register is set.
- To stop decoding, the PLAY register should be reset. Resuming decoding is performed by writing PLAY to 1 again

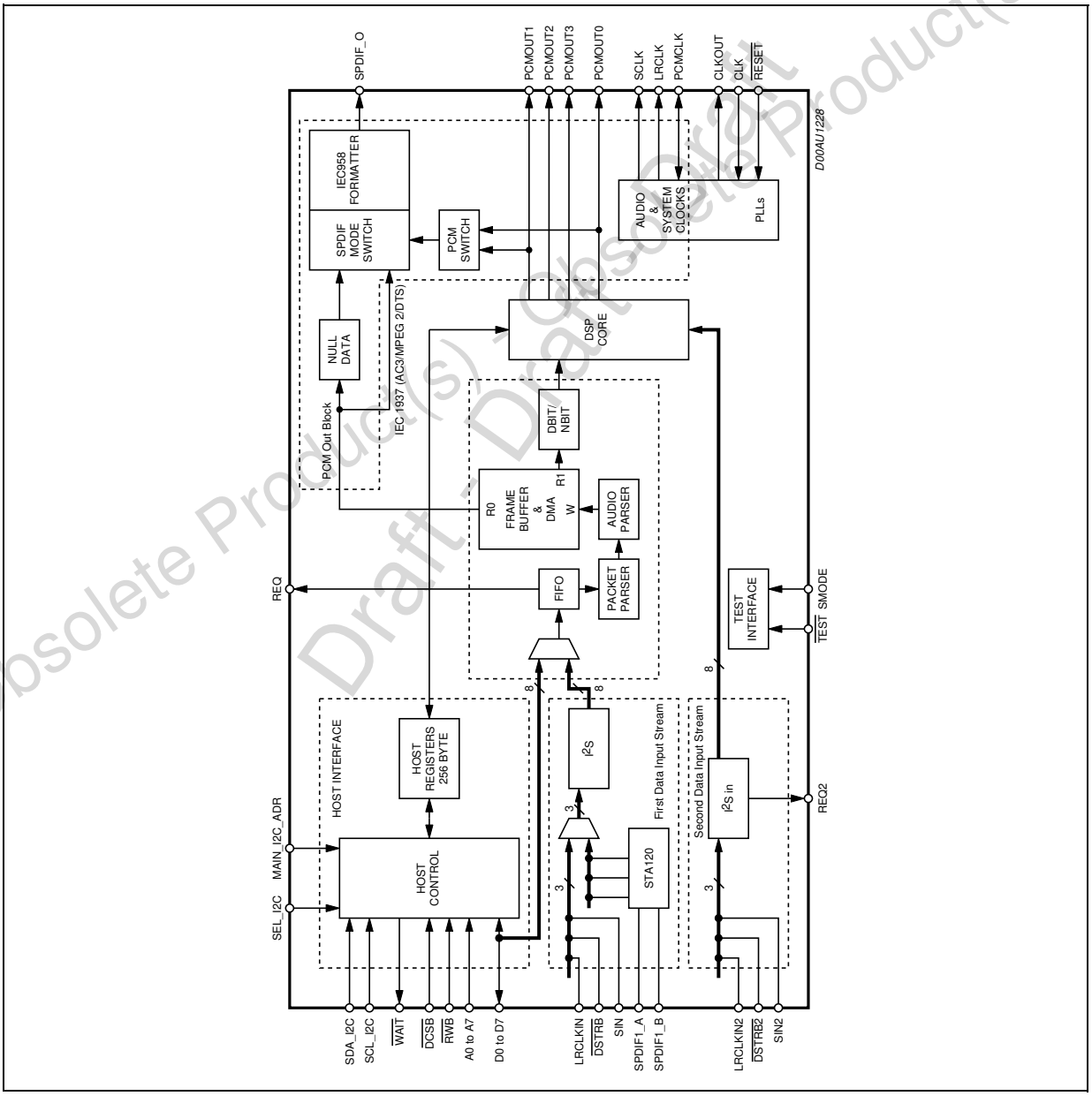
Table 3. Decode Mode. Play and Mute commands effects

Play	Mute	Clock State	PCM Output	Decoding
0	0	Not running	0	No
0	1	Running	0	No
1	0	Running	Decoded Samples	Yes
1	1	Running	0	Yes

Note: 1. It is not possible to change configuration registers in this state. It is necessary to soft reset the chip before. Only the following registers can be changed “on-the-fly”: PCM_SCALE, BAL_LR, BAL_SUR, OCFG, DOWNMIX registers.

4.4 Data input interface description.

Figure 6. Block Diagram of Data Flow



Two independent inputs are available on the STA310.

The main one allows to enter input data stream through through:

- A serial interface (referred to as Data Serial Interface),
- And a parallel interface (referred to as Data Parallel Interface).

The choice is performed by the register SIN_SETUP.

4.4.1 Data serial interface

When the serial mode is selected, the bitstreams can be entered into the STA310 through either:

- a four-signal data interface or ,
- trough a SPDIF input (no external circuit is required).

The four-signal data interface (see Figure 5) provides:

- An input data line SIN,
- An input clock $\overline{\text{DSTR}}$,
- A word clock input LRCLKIN
- And a hand-shake output signal $\overline{\text{REQ}}$.
-

Note: 1. Only 16-bit PCM streams are supported. For 20-bit or 24-bit PCM, the 4 or 8 least significant bits are ignored

.

The specifications of those signals can be configured by the means of the register CAN_SETUP.

Two modes exist in serial mode, one that uses the LRCLKIN pin and one that does not use the LRCLKIN pin.

4.4.1.1 Modes without the LRCLKIN pin

In this mode the signal LRCLKIN is not used by the STA310. The input data SIN is sampled on the rising edge of $\overline{\text{DSTR}}$. When the STA310 input buffer is full the $\overline{\text{REQ}}$ signal is asserted. The polarity of $\overline{\text{REQ}}$ signal is programmable through the register SIN_SETUP. The data must be sent most significant bits first.

When the decoder cannot accept further data the $\overline{\text{REQ}}$ is de-asserted and the $\overline{\text{DSTR}}$ clock must be stopped as soon as possible to avoid data loss. After the $\overline{\text{REQ}}$ is de-asserted, the decoder is still able to accept data for a limited number of clock cycles.

The maximum number of data that can be transmitted with respect to the change of $\overline{\text{REQ}}$ is given by the following formula: Nbits = 23 - 6 * $F_{\overline{\text{DSTR}}}$/33MHz, where: $F_{\overline{\text{DSTR}}}$ is the $\overline{\text{DSTR}}$ clock frequency, (max is 33 MHz).

4.4.1.2 Modes using the LRCLKIN pin

When receiving data from an A/D converter or from an S/PDIF receiver, the signal LRCLKIN is used.

The LRCLKIN signal is used to make the distinction between the left and right channels. Any edge of the LRCLKIN signal indicates a word boundary.

The data transfer between the input interface and the FIFO is done on a byte basis. After the edge (rising or falling) of the LRCLKIN, a new byte is transferred to the first stage of the STA310 every 8 $\overline{\text{DSTR}}$ clock cycles.

If the number of time slots is not a multiple of 8, the remaining data is lost. The polarity of LRCLKIN and $\overline{\text{DSTR}}$ is programmable.

The LRCLKIN can be delayed by one time slot, in order to support PCM delayed mode. All these configurations are programmable through the CAN_SETUP register.

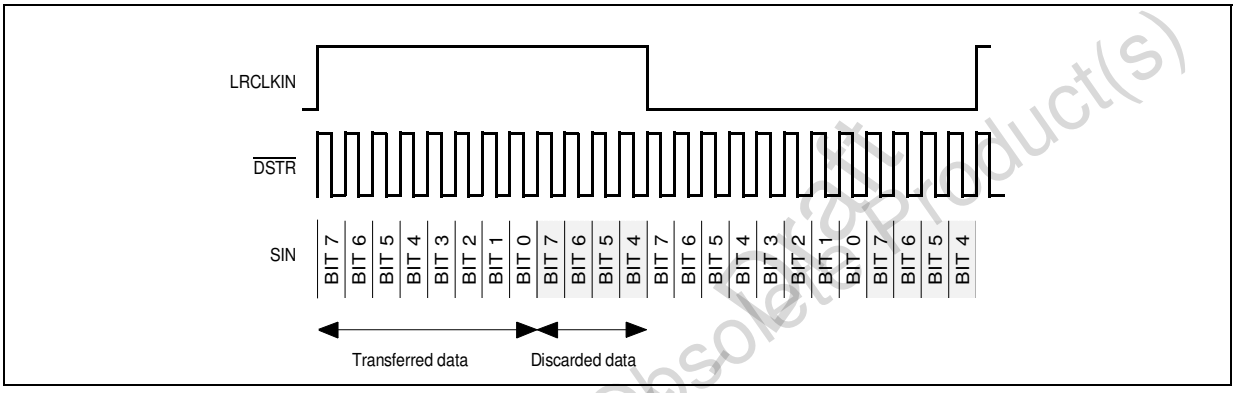
The register CAN_SETUP has 4 significant bits, and each bit has a specific meaning, see *CAN_SETUP* on page 41.

Only the first byte is transferred to the STA310 because the number of time slots is 12 (8 + 4). SIN and LRCLKIN are sampled on the falling edge of DSTR In this case SIN_SETUP = 3 and CAN_SETUP = LeftFirstChannel + FallingStrobe + AllSlot = 2 + 4 + 8 = 14

Table 4.

	When Set	When Clear	Name
Bit 0	The input data is one slot delayed with respect to LRCLKIN	The input data is not delayed	DelayMode
Bit 1	First channel when LRCLKIN is set	First channel when LRCLKIN is reset	LeftFirstChannel
Bit 2	Data are sampled on falling edge of DSTR	Data are sampled on rising edge of DSTR	FallingStrobe
Bit 3	All the bytes are extracted	Only the first 16 data bits are extracted	AllSlot

Figure 7.



Example 2: Only the first 2 bytes are transferred to the STA310 because the number of slots is 20 (16 + 4). SIN and LRCLKIN are sampled on the falling edge of DSTR. The data is in delayed mode.
The register configuration is SIN_SETUP=3 and CAN_SETUP = DelayMode + LeftFirstChannel + FallingStrobe + AllSlot = 1 + 2 + 4 + 8 = 15.
This mode is a specific mode where only the first 16 data bits are transferred. The remaining bits are discarded.
The register configuration is SIN_SETUP = 3 and CAN_SETUP = DelayMode + FallingStrobe = 1 + 4 = 5.

4.4.1.3 SPDIF Input

A true SPDIF Input **SPDIF** (PCM audio samples) or **IEC-61937** (compressed data) is selectable as a main serial input.

4.4.1.4 Autodetected formats

The STA310 cut 2.0 is able the following audio format changes on the s/pdif input

Table 5. Audio Format detection

BEFORE	AFTER
AC3	PCM
AC3	MPEG
MPEG	AC3
MPEG	PCM
PCM	AC3
PCM	MPEG

4.4.1.5 Second Input

A second independent input allows to input bitstreams in serial mode.

This second input can be used, to input audio stream from a microphone, while we decode a data stream trough the main input.

4.4.2 Data parallel interface

Two ways are available to input data in parallel mode:

- Either through the parallel data bus, shared with the external controller,
- Or through the DATAIN register

4.4.2.1 Using the parallel data bus

In this mode the data must be presented on the 8-bit parallel host data bus D[7..0]. Note that this bus is shared with the external controller. On the rising clock of $\overline{DSTR}$ the data byte is sampled by the STA310. The signal $\overline{REQ}$ is used to signal when the input FIFO is full. When $\overline{REQ}$ is de-asserted the transfer must be stopped to avoid data loss.

After the $\overline{REQ}$ is de-asserted, the decoder is still able to accept data for a limited number of clock cycles.

The maximum number of data that can be transmitted with respect to the change of $\overline{REQ}$ is given by the following formula: Nbits = 23 - 6 * F_{DSTR}/33MHz, where: F_{DSTR} is the $\overline{DSTR}$ clock frequency, (max is 33 MHz).

The signals $\overline{DSTR}$ and $\overline{DCSB}$ are used to make the distinction between Stream Data (strobed by $\overline{DSTR}$) and Control Data (strobed by $\overline{DCSB}$). To avoid conflicts, the $\overline{DSTR}$ signal and the $\overline{DCSB}$ signal must respect given timing constraints.

4.4.2.2 Using the DATAIN register

The data can be input by using the control parallel interface as if accessing any other register.

The signal $\overline{DCSB}$ is therefore used. When using this register to input data stream, there is no need to byte-align the data.

Figure 8.

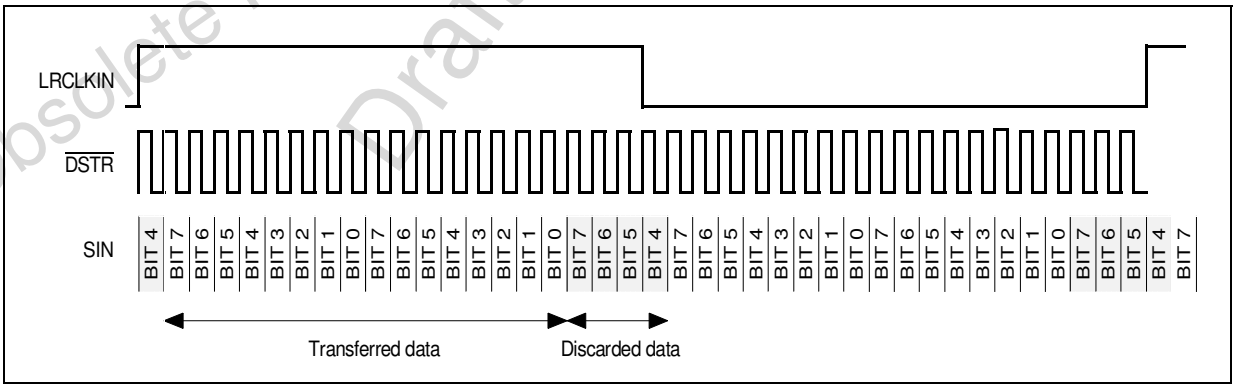
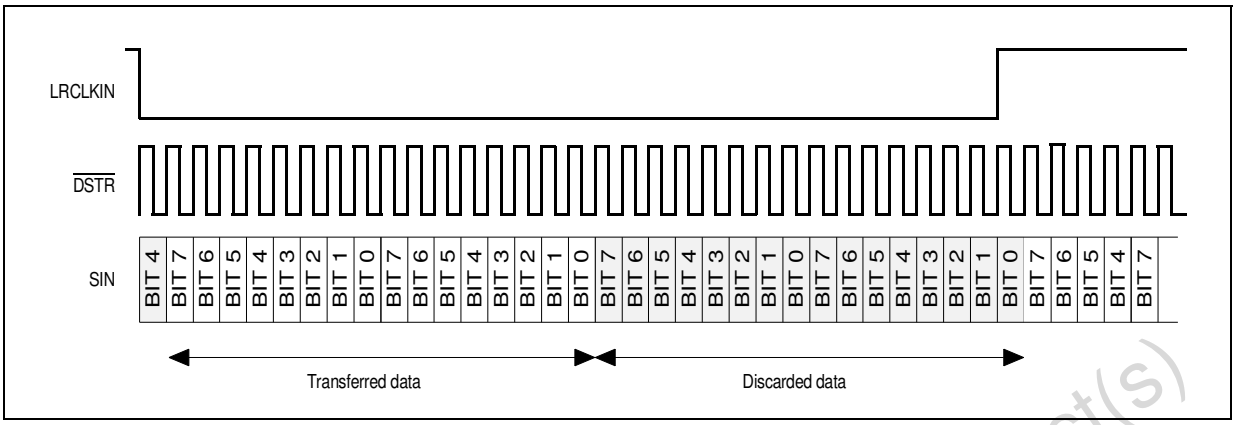


Figure 9.



4.5 Streams parsers

The parsing stage is operated by two parts: the packet parser and the audio parser.

The packet parser unpacks stream, sorts packets and transmit data to the audio parser. The audio parser verifies the stream syntax, extracts non-audio data and sends audio data to the frame buffer.

Packet parser

Before unpacking packets and transmitting data, the packet parser needs to detect the packet start by recognizing the packet synchronization word. It is possible to force the parser to search for two packet synchronization words before starting to unpack and transmit.

This is done by setting the register PACKET_LOCK to 1. Otherwise, the packet parser will start handling the stream once it has detected information matching the packet synchronization word.
The packet parser is also able to perform selective decoding: it can decode audio packets that are matching a specified Id. This Id is specified in AUDIO_ID and AUDIO_ID_EXt registers, and the function is enabled by setting the AUDIO_ID_EN register.

Audio parser

The audio parser needs to detect the audio synchronization word corresponding to the type of stream that must be decoded. It is possible to force the audio parser to detect more than one synchronization word before parsing.
This is done by setting the SYNC_LOCK register to a value between 1 and 3 - number of supplementary sync words to detect before considering to be synchronized.

The status of synchronization of both parsers is provided in the register SYNC_STATUS. Each time the synchronization status of one of the two parsers changes, the interrupt SYN is generated (if enabled) and the status can be read in SYNC_STATUS.

4.6 Decoding modes

4.6.1 AC-3

The STA310 is Dolby Digital certified for class A products. The decoder must be programmed so to specify the stream format as AC-3 encoded: register DECODESEL = 0.
In the sections below are provided the modes specific to the AC-3 decoding.

4.6.1.1 Compression modes

Four compression modes are provided in the STA310:

- Custom A (also named custom 0 in Dolby specifications),
- Custom D (also named custom 1 in Dolby specifications),
- Line mode,
- RF mode.

These modes refer to different implementation of the dialog normalization and dynamic range control features. The mode is selected by programming the register COMP_MOD to the appropriate value.

Line Mode

In Line Mode (COMP_MOD = 2), the dialog normalization is always enabled. It is done by the decoder itself and the dialog is reproduced at a constant level.

The dynamic range control variable encoded in the bitstream is used and can be scaled by the two scaling registers HDR (for high-level cut compression) and LDR (for low-level boost compression). In case of 2/0 downmix, the high-level cut compression is not scalable.

RF Mode

In RF Mode (COMP_MOD=3), the dialog normalization is always performed by the decoder. The dialog is reproduced at a constant level.

The dynamic range control and heavy compression variables encoded in the bitstream are used, but the compression scaling is not allowed. This means that the HDR and LDR registers can not be used in this mode. A +11dB gain shift is applied on the output channels.

Custom A Mode

In Custom A mode (COMP_MOD=0), the dialog normalization is not performed by the decoder and must be done by another circuit externally.

The dynamic range control variable encoded in the bitstream is used and can be scaled by the two scaling registers HDR (for high-level cut compression) and LDR (for low-level boost compression).

Custom D Mode

In Custom D mode (COMP_MOD=1), the dialog normalization is performed by the decoder. The dynamic range control variable encoded in the bitstream is used and can be scaled by the two scaling registers HDR (for high-level cut compression) and LDR (for low-level boost compression).

4.6.1.2 Karaoke mode

The AC-3 decoder is karaoke aware and capable.

A karaoke bitstream can be composed of 5 channels: L for Left, R for Right, M for guide Melody, V1 for vocal track 1 and V2 for Vocal track 2.

- When in karaoke aware mode, the channels L,R and M are reproduced, and the channels V1 and V2 are reproduced at a level fixed by the bitstream.
- When in karaoke capable mode, it is possible to choose to reproduce one, two or none of the two incoming vocal tracks, V1 and V2.

The karaoke decoder is activated by the use of KARAMODE register, which specifies the downmix for the different modes. This register replaces DOWNMIX register. It is however possible to consider the incoming karaoke channels as any other multichannel stream and output it with a downmix specified in DOWNMIX register. For details, refer to the Digital Audio Compression AC-3 ATSC standard, annex C.

4.6.1.3 Dual Mode

The Dual Mode corresponds to a mode where two completely independent mono program channels (e.g. bilingual) are encoded in the bitstream, referenced to as channel 1 and channel 2.

The possible ways to output channels on left/right outputs are:

- Output channel 1 on both L/R outputs,
- Output channel 2 on both L/R outputs,
- Mix channels 1 and 2 to monophonic and output on both L/R,
- Output channel 1 on Left output, and channel 2 on Right output.

This channels downmix is specified in the register DUALMODE.

4.6.2 MPEG

The STA310 is able to decode MPEG-1 layerI and layerII encoded data, as well as MPEG-2 layer I, layer II data without extension (i.e. 2-channel streams).

The MPEG input format should be specified in the DECODESEL register:

- DECODESEL=1 for MPEG1. The MC bit in MC_OFF register should be set.
- DECODESEL=2 for MPEG2. The MC bit in MC_OFF register should be set.

4.6.3 MP3

The STA310 is able to decoder MPEG2 layer III (MP3) data.

The MP3 input format aboved be specified in the DECODESEL register:

- DECODESEL=9 for MP3.

4.6.3.1 Dual Mode

The Dual Mode corresponds to a mode where two completely independent mono program channels (e.g. bilingual) are encoded in the 2-channel incoming bitstream, referenced to as channel 1 and channel 2.

The audio decoder allows to:

- Output channel 1 on both L/R outputs,
- Output channel 2 on both L/R outputs,
- Mix channels 1 and 2 to monophonic and output on both L/R,
- Output channel 1 on Left output, and channel 2 on Right output.

The output configuration is chosen by special downmix for dual mode through register MPEG_DUAL.

4.6.3.2 Decoding flow

Figure 10. AC-3 Decoding Flow

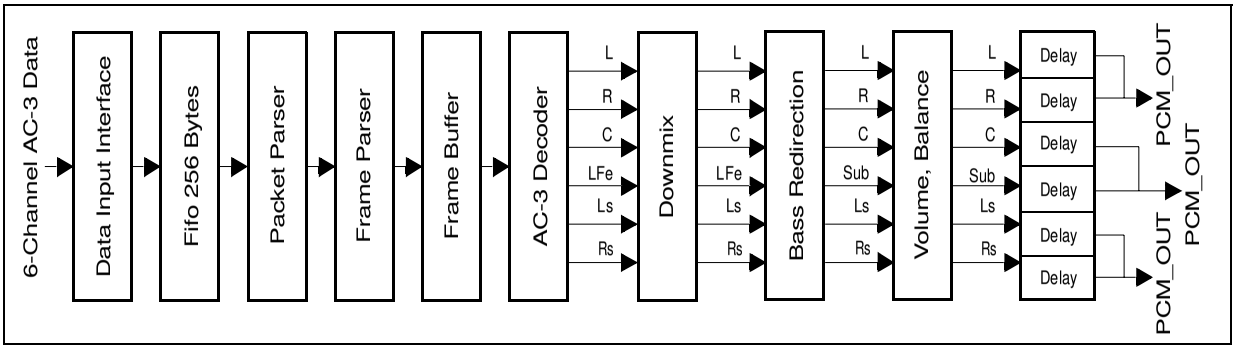
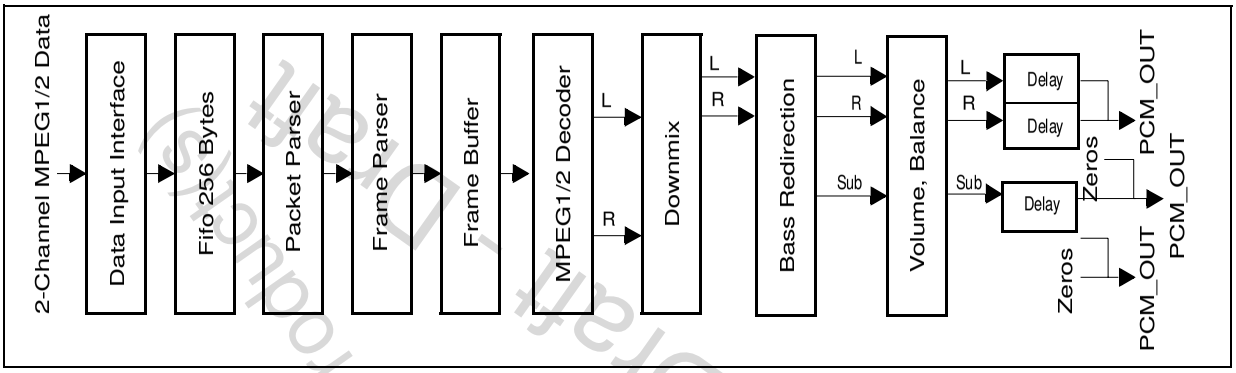


Figure 11. MPEG Decoding Flow



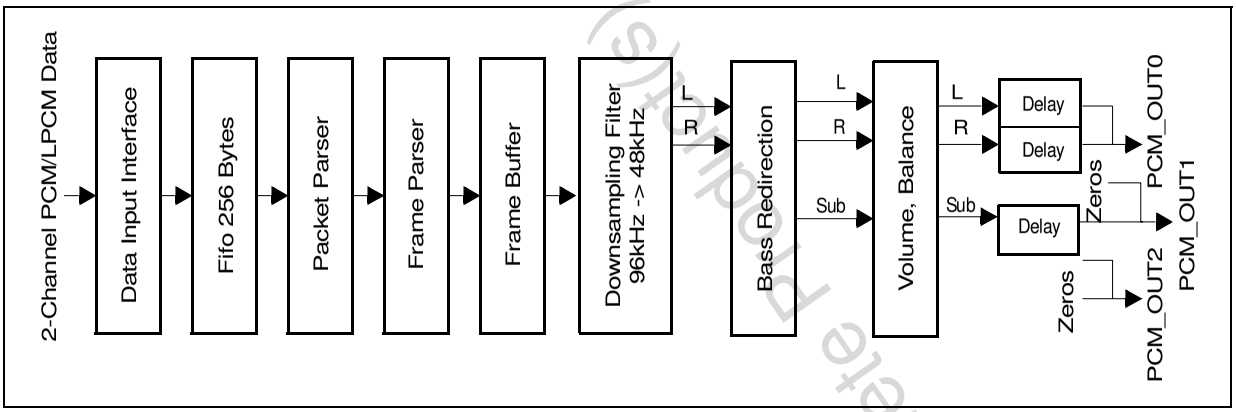
4.6.4 PCM/LPCM

The decoder supports PCM (2-channels) and LPCM Video (8-channels) and Audio (6-channels) streams. This is selected by DECODESEL=3.

4.6.4.1 Downsampling filter

When decoding PCM/LPCM streams encoded at 96kHz, it is possible to use a filter that downsamples the stream from 96kHz to 48kHz. The chip can not output streams at 96kHz. The register DWSMODE is used to configure the use of this filter.

Figure 12. PCM/LPCM Decoding flow



4.6.5 MLP

MLP is a lossless coding system for use on digital audio data originally represented as linear PCM. MLP is mandatory in DVD Audio. It allows transmission and storage of up to 6 channels. each up to 24 bits precision and with sample rates between 44.1 KHz and 192KHz.

- DECODESEL = 8

4.6.6 CDDA

- DECODESEL = 5

4.6.7 Beep Tone

- DECODESEL = 7

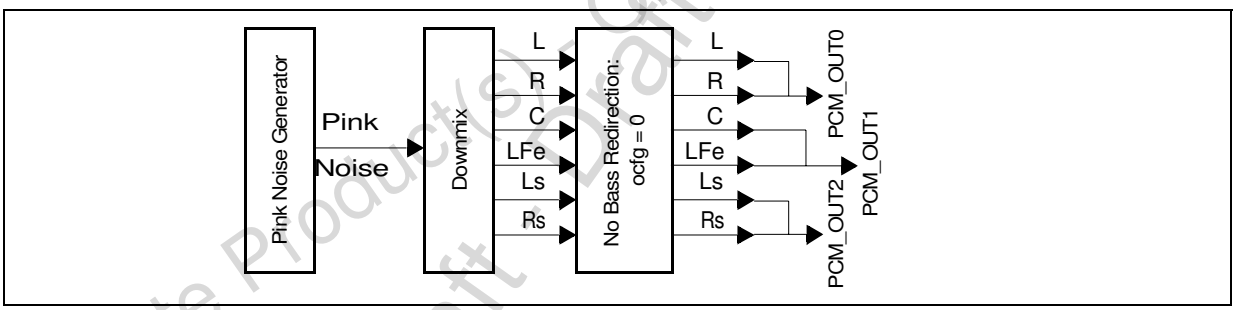
4.6.8 Pink noise generator

The pink noise generator can be used to position the speakers in the listening room so to benefit of the best listening conditions.

The decoder must be programmed so to generate pink noise by writing 4 in the DECODESEL register. The DOWNMIX register is used to select independently the channels on which the pink noise will be output.

When generating pink noise, the output configuration should be: OCFG=0 and PCM_SCALE=0.

Figure 13. Pink Noise Generator Flow



4.7 Post Processing

The following post processing alghorithms are available

4.7.1 Prologic

Pro Logic Compatible Downmix

The STA310 can decode an AC-3 multichannel bitstream and encode it to provide a 2-channel Pro Logic compatible output (Lt, Rt). These 2 channels are the result of a specific downmix referred to as Pro Logic compatible. This downmix is selected by the register DOWNMIX. The 2 channels can be used as the input of a Pro Logic decoder and player (e.g. home theatre).

Pro Logic Decoding

The STA310 can decode a 2-channel Pro Logic bitstream. The 2 channels could come from a CD player, an AC-3 2-channel bitstream or an MPEG1 bitstream. The 2-channel bitstream can be converted into a 4-channel output (L, R, C, S). The surround (S) is simultaneously sent on Ls and Rs channels. A Pro Logic downmix en-