# LM4560

***LM4560 Advanced PCI Audio Accelerator***


Literature Number: SNAS029

National Semiconductor

December 1998

# LM4560
# Advanced PCI Audio Accelerator

## General Description

LM4560 is an advanced PCI audio accelerator providing full legacy compatibility, wavetable synthesis, DirectMusic, DirectSound, and DirectSound3D on a single chip for the high-performance, cost-sensitive consumer market. It supports Sound Blaster functionality and is fully PC98 compliant.

The LM4560 integrates a 64-voice wave table engine with per voice effect processing capability. It supports the upcoming Microsoft™ DirectMusic API and is fully compatible with DLS Level 1 (downloadable samples) specification. The LM4560 is optimized for Microsoft Windows™ 98 and Windows NT5.0 WDM streaming architecture with re-routable end-point support. The LM4560 includes DirectSound3D acceleration hardware for ITD (Interaural Time Difference), IID (Interaural Intensity Difference), Pan, Delay, and Doppler hardware.

The LM4560 supports Sound Blaster Pro/16 DOS games while improving gaming audio quality. The LM4560 supports both the legacy analog gameport and a Digital Enhanced Game Port. When coupled with DirectInput™ driver, the Digital Enhanced Game Port can save up to 12% of the CPU overhead nominally required by a conventional analog game port. The LM4560 employs a high precision 26-bit digital mixer, providing an accurate 20-bit output and higher than 90 dB signal-to-noise ratio when used with high quality AC97 Rev 1.03 and Rev 2.0 codecs.

The LM4560 is designed with aggressive power management. It is PCI Bus Power Management Interface (PPMI 1.0)-compliant. The LM4560 supports multiple Rev 2.0 AC97 codecs, which are useful for notebook docking systems. With a low power, 3.3V process and a space conscious 100 TQFP package, the LM4560 is also well suited for Notebook systems.

In summary, the LM4560 provides a balanced combination of features and performance to the end-user. By combining PCI bus mastering for DirectSound acceleration, Hardware Wave table synthesizer, Digital Enhanced Game Port and DirectSound 3D audio acceleration. It delivers high performance, high quality audio, high-end features with efficient power management in a space-efficient 100 TQFP package.

## Features

- PCI 2.1-compliant with Bus Mastering optimized for multiple stream operation
- On-chip per voice cache minimizes PCI bandwidth
- Up to 20X improvement over ISA DMA on PCI bus bandwidth utilization

**Advanced Wavetable Synthesizer**

- Wavetable synthesis with 64-voices polyphony, which supports all combinations of stereo/mono, 8-/16-bits, and signed/unsigned samples
- Per channel volume, pitch shift, left/right pan, tremolo, vibrato and envelope control (32 channels in hardware)
- Per channel effect process ing and effect volume control for reverb, chorus, and echo
- DLS1-compliant Downloadable Samples support

**Legacy Compatibility**

- Legacy game audio support with SoundBlaster Pro/16 compatibility on the PCI bus
- Legacy DMA support on PCI Bus with DDMA-enabled or standard (non-DDMA) PCI chipsets
- FM through realtime FM-to-wavetable conversion
- MPU-401 compatible UART for external or internal synthesis
- General MIDI/GS command interpretation for wavetable & effect synthesis

**High Quality Audio and AC97 Support**

- CD quality audio with higher than 90 dB signal-to-noise ratio using an external high quality AC97 codec
- AC97 support with full duplex, independent sample rate converter for audio recording and playback
- On-chip sample rate converter ensures all internal operation at 48 kHz
- High precision internal 26-bit digital mixer with 16- and 20-bit digital audio output

**Advanced Streaming Architecture**

- Microsoft WDM Streaming architecture compliant and “Re-routable endpoint” support
- Three stereo capture channels
- AC97 stereo recording channel through AC-link

**DirectSound 3D**

- 64-voices DirectSound channels
- 32-voices DirectSound3D accelerator with IID, ITD, and Doppler effects on 3D positional audio buffers
- DirectSound accelerator for volume, pan, and pitch shift control on streaming or static buffers

**Telephony & Modem**

- Full duplex speaker phone with AC97 2.0 audio-modem codec
- Fax-modem with host based software

**Extras**

- Fully Plug and Play PCI controller and software
- Digital Enhanced Game port enables an analog joystick to emulate digital joystick performance using the National Semiconductor-provided DirectInput driver. This eliminates up to 12% CPU overhead wasted on joystick polling.

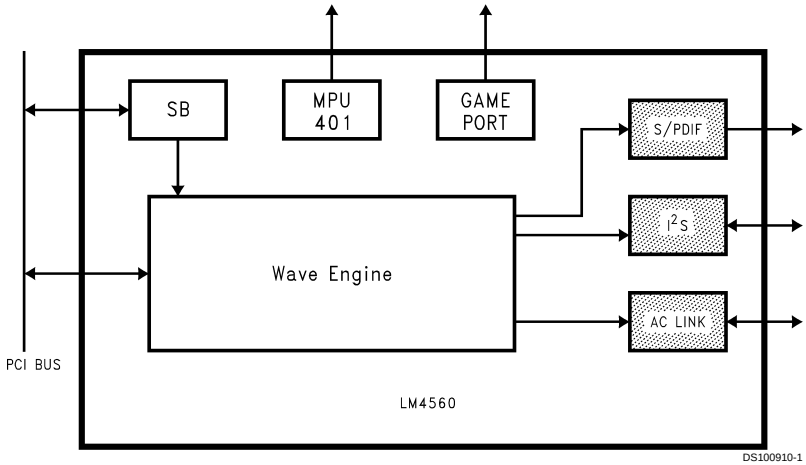
## Features (Continued)

- DirectX timer for video/audio synchronization
- Forward pin-compatible with future PCI audio accelerators
- 100-pin TQFP package
- 3.3V operation

**Software Support**

- Complete DirectX driver suite (DirectSound3D, DirectSound, DirectMusic, and DirectInput) for Windows 95 and Windows 98/NT 5.0
- Configuration installation, and diagnostics under real mode DOS, WindowsWindows 95, and Windows 98 DOS box
- Windows 3.1, 95, NT4.0, Windows 98/NT5.0 configuration, installation, and mixer program
- 1, 2, 4, or 8 Mbytes General MIDI (GM)/General Sound (GS) compliant sample Library

## Block Diagram




## Table of Contents

## Table of Contents (Continued)

# 1.0 Pin Description

## 1.1 PACKAGE DIAGRAM

**Top View**

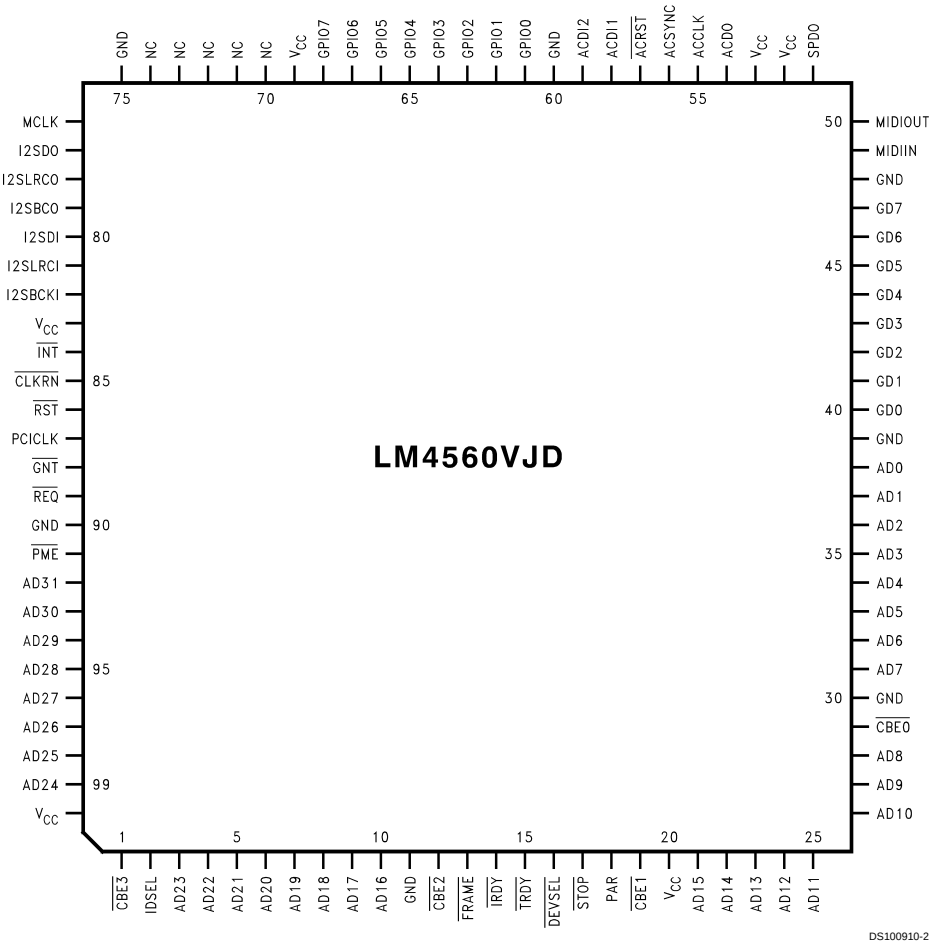



**FIGURE 1. 100-Pin TQFP Package**

## 1.0 Pin Description (Continued)

### 1.2 PIN DESCRIPTION

| Symbol | Pin(s) | Type | Description |
|---|---|---|---|
| $V_{DD}$ | 20, 39, 52, 53, 69, 83, 100 | IN | **3.3V Power Supply** |
| $V_{SS}$ | 11, 30, 48, 60, 75, 90 | IN | **Ground** |
| **PCI BUS INTERFACE SIGNALS (51)** | | | |
| AD[31:0] | 92–99, 3–10, 21–28, 31–38 | I/O | **PCI Address/Data Bus.** A time-multiplexed address data bus. |
| C/$\overline{BE}$[3:0] | 1, 12, 19, 29 | I/O | **PCI Command/Byte Enable.** Defines the type of AD bus transaction type. |
| $\overline{FRAME}$ | 13 | I/O | **Cycle Frame.** A PCI transaction begins and ends with the $\overline{FRAME}$ signal. |
| $\overline{IRDY}$ | 14 | I/O | **Initiator Ready.** An active low indicates the cycle initiator is ready to send or receive data. |
| $\overline{TRDY}$ | 15 | I/O | **Target Ready.** An active low indicates that the target is read to complete the current transaction. |
| $\overline{DEVSEL}$ | 16 | I/O | **Device Select.** An active low indicates the target has decoded its address. |
| $\overline{STOP}$ | 17 | I/O | **STOP.** An active low indicates the target wants the initiator to stop data during the current data phase. |
| PAR | 18 | I/O | **Parity.** Generates an even parity for the AD[31.0]. |
| $\overline{RST}$ | 86 | IN | **Reset.** A low active signal which resets the PCI device. |
| $\overline{INT}$ | 84 | OUT | **Interrupt.** An active low signals in interrupt to the CPU. |
| PCICLK | 87 | IN | **Clock.** The clock that drives the PCI timing. |
| $\overline{GNT}$ | 88 | IN | **Grant.** An active low signals the master has access to the PCI bus. |
| $\overline{REQ}$ | 89 | OUT | **Request.** An active low indicates the master wants access to the PCI bus. |
| $\overline{CLKRUN}$ | 85 | I/O | **Clock Run.** An active low signal used for power management on motherboards only. It is not assigned a pin on the PCI connector. |
| $\overline{PME}$ | 91 | OUT | **Power Management Event.** An active low signal used for power management for add in cards or motherboards. It is assigned a pin on the PCI connector. |
| TC | 78 | IN | **Terminal Count.** This input is asserted by the DMA controller to indicate the end of a DMA transfer. The signal is only effective during a DMA access cycle. |
| IDSEL | 2 | IN | **Initialization Device Select.** An active high allows reads to the PCI devices configuration registers. |
| **MPU-401 INTERFACE SIGNALS (2)** | | | |
| MIDIOUT | 50 | OUT | **MIDI Data Out.** Sends midi data to the midi connector. |
| MIDIIN | 49 | IN | **MIDI Data In.** Receives midi data from an optocoupler. |
| **AC97 CODEC INTERFACE SIGNALS (6)** | | | |
| $\overline{ACRST}$ | 57 | OUT | **AC97 Master Reset.** An active low which resets the internal circuitry of AC97 codecs. |
| ACCLK | 55 | IN | **AC97 Bit Clock.** A 12.288 MHz clock from the codec. This is used to synchronize the data streams to and from the codecs. |
| ACSYNC | 56 | I/O | **AC97 Sync.** Used to start the data frame used to format the serial data to and from the codecs. |
| ACDI1 | 58 | IN | **Primary CODEC Serial Data Input.** This receives serial data in from the primary codec. |
| ACDI2 | 59 | IN | **Secondary CODEC Serial Data Input.** This receives serial data in from the secondary codec. |

## 1.0 Pin Description (Continued)

| Symbol | Pin(s) | Type | Description |
|---|---|---|---|
| **AC97 CODEC INTERFACE SIGNALS (6)** | | | |
| ACDO | 54 | OUT | **AC97 serial data out.** This output sends serial data to codec(s). |
| **I2S INTERFACE (7)** | | | |
| I2SBCKI | 82 | IN | **I2S Receiver Bit Clock** |
| I2SLRCKI | 81 | IN | **I2S Receiver L/R Clock** |
| I2SDI | 80 | IN | **I2S Receiver Data** |
| I2SBCKO | 79 | OUT | **I2S Transmitter Bit Clock** |
| I2SLRCKO | 78 | OUT | **I2S Transmitter L/R Clock** |
| I2SDO | 77 | OUT | **I2S Transmitter Data** |
| MCLK | 76 | OUT | **I2S Transmitter Master Clock** |
| **S/PDIF TRANSMITTER (1)** | | | |
| SPDO | 51 | OUT | **S/PDIF transmitter output** |
| **GAME PORT PINS (8)** | | | |
| GD[7:4] | 47:44 | IN | **Gameport data** |
| GD[3:0] | 43:40 | IN | **Gameport data** |
| **GENERAL-PURPOSE I/O (8)** | | | |
| GPIO[7:0] | 68:61 | I/O | **General-purpose input/output** |
| **NO CONNECTIONS (5)** | | | |
| NC | 70:74 | | No connections |

## 2.0 Functional Description

### 2.1 DEVICE OVERVIEW

**2.1.1** The LM4560 is composed of several functional blocks. There is a PCI bus interface, a wave engine, I/O interfaces and legacy support hardware.

**2.1.1.1** The PCI bus interface includes all the logic that talks to the PCI bus, handles the PCI bus cycles, reports PCI configuration block and PCI power management. The EEPROM interface for supporting the SubSystem and SubVendor ID fields for PCI card vendors is located here.

**2.1.1.2** The wave engine supports all the logic necessary to synthesize wave table voices / instruments. The sample rate converter is used to pitch shift the samples and rate convert pcm streams. For example converting from 44.1 ksps CD audio to 48ksps DAT pcm. The wave engine is also used to play back WDM streams and for the legacy PC audio functions. There are three mixers in the wave engine, the main mixer, and mixers for chorus and reverb sends.

**2.1.1.3** The I/O ports include the AC97 Rev 2.0 I/O, the I2S I/O, the SPDIF out, GPIOs, gameport and MPU401 midi I/O.

**2.1.1.3.1** The AC97 port supports two AC97 codecs. The primary codec must be Rev 1.03, 2.0 or 2.1. The secondary codec must be Rev 2.0 or 2.1.

**2.1.1.3.2** The I2S port supports data out at 48 KSPS only and inputs at rates from 8 to 48 KSPS. Output can come from I2S L/R Output Buffer or SURR L/R Output FIFO. The input data rate can be determined by reading the 13 lsbs of register 0xCC. The sample rate = (I2S_DELTA)*60. The transmitter sends 20 bits stereo and the receiver receives 20 stereo. When the I2S data in receives more than 20 bits, the lsb's are truncated. If the receiver receives less than 20 bits, zeros are appended to the lsbs.

**2.1.1.3.3** The SPDIF output only supports Mode 0 consumer 20 bit audio data that is sampled at 48 kHz. If the receiver needs fewer bits, then the lsbs will be truncated.

**2.1.1.3.4** The GPIO pins are 5V tolerant programmable I/O pins. They can be programmed to either inputs or output via a register at offset 7Ch. On power up all the pins default to inputs.

If any of the GPIO pins are used for hardware control pins, the best programming procedure would be to leave the i/o pins as inputs, then set the output bits to highs or lows, then change the selected I/O pins to outputs. This will avoid the possibility of glitches on the output. Since the pins power up as inputs, they will be floating and so 30 kΩ resistors are needed to put the pin into a know state on power up.

The inputs can be read from the same register 7Ch. The CPU must poll the 7Ch to see if any of the inputs have changed state.

**2.1.1.3.5** The Game port pins are 5V tolerant I/O pins and support the standard dual analog joysticks.

The enhanced mode works will all normal analog input joysticks. The CPU can read internal counters rather than measuring the time constants of each of the X, Y inputs with software timing loops.

**2.1.1.3.6** The LM4560 supports the standard MPU 401 uart mode midi interface.

# 3.0 Register Description

### 3.1 PCI CONFIGURATION REGISTER MAP

| Offset | +3h | +2h | +1h | +0h |
|---|---|---|---|---|
| 00h | Device ID | | Vendor ID | |
| 04h | Status | | Command | |
| 08h | Class Code | | | Revision ID |
| 0Ch | BIST | Header Type | Latency Timer | Cache Line Size |
| 10h | Audio IO Base Address | | | |
| 14h | Audio Memory Base Address | | | |
| 18–28h | RSVD | | | |
| 2Ch | Subsystem ID | | Subsystem Vendor ID | |
| 30h | RSVD | | | |
| 34h | RSVD | | | Cap_Ptr |
| 38h | RSVD | | | |
| 3Ch | MAX_LAT | MIN_GNT | Interrupt Pin | Interrupt Line |
| 40h | DDMA Slave CFG | | | |
| 44h | PM_Timer | PM_CFG | LEGACY_DMA | LEGACY_IOBASE |
| 48h | RSVD | | INT_VEC | INTA_SNOOP_ENA |
| DCh | PMC | | PM_Next_Ptr | PM_Cap_ID |
| E0h | Power Value Data | PMCSR_BSE | PMCSR | |

#### 3.1.1 PCI Configuration Register Description

**3.1.1.1 Device ID & Vendor ID**

**PCI Configuration**

**Address:** 00h–03h

**Default:** 001B100Bh

**Description:** read/write, can be written only when CFG46h[6]=1

Bit 31–16 device ID: default 001Bh

Bit 15–0 vendor ID: default 100Bh

**3.1.1.2 Status & Command:**

**PCI Configuration**

**Address:** 04h–07h

**Default:** 02900000h

**Description:**

Bit 2–0: R/W

Bit 28: TA Received target abort. Write 1 to clear.

Bit 29: MA Received master abort. Write 1 to clear.

Bit 20: PM PCI Power Management support, hardwired to 1

Bit 23, 25: hardwired to 1

The rest bits: hardwired to 0

**3.1.1.3 Class Code & Revision ID:**

**PCI Configuration**

**Address:** 08h–0Bh

**Default:** 04010001h

**Description:** read only

Bit 7–0: 01 revision ID

Bit 31–24: 04 Base class: Multimedia

Bit 23–16: 01 Sub-class: Audio device

Bit 15–8: 00 Interface

**3.1.1.4 BIST, Header Type, Latency Timer & Cache Line Size:**

**PCI Configuration**

**Address:** 0Ch–0Fh

**Default:** 00000000h

**Description:**

Bit 15–12: R/W

The rest bits: hardwired to 0

**3.1.1.5 Audio IO Base Register:**

**PCI Configuration**

**Address:** 10h–13h

**Default:** 00000001h

**Description:**

Bit 31–8: R/W Audio IO base

Bit 7–1: Hardwire to 0

Bit 0: Hardwire to 1

**3.1.1.6 Audio MEM Base Register:**

**PCI Configuration**

**Address:** 14h–17h

**Default:** 00000000h

**Description:**

Bit 31–12: R/W Audio MEM base

Bit 11–0: Hardwire to 0

**3.1.1.7 Subsystem ID & Subsystem vendor ID:**

**PCI Configuration**

**Address:** 2Ch–2Fh

**Default:** 001B100Bh

**Description:** read only

Bit 31–16: subsystem ID: default 001Bh

Bit 15–0: subsystem vendor ID: default 100Bh

## 3.0 Register Description (Continued)

**3.1.1.8 PCIPM Capability List Pointer Register:**

**PCI Configuration**

**Address:** 34h–37h

**Default:** 000000DCh

**Description:** Read Only

**3.1.1.9 Max_Lat, Min_Gnt, Interrupt Pin & Interrupt Line:**

**PCI Configuration**

**Address:** 3Ch–3Fh

**Default:** 18020100h

**Description:**

Bit 7–0: INT line R/W

Bit 15–8: INT pin hardwired to 01

Bit 23–16: Min_Gnt hardwired to 02

Bit 31–24: Max_Gnt hardwired to 18

**3.1.1.10 DDMA Slave Configuration Register**

**PCI Configuration**

**Address:** 40h–43h

**Type:** Read/Write

**Default:** 00000000h

Bit 0: DDMA Slave Channel Access Enable Control

0 disabled

1 enabled

When disabled, the DDMABase is not usefull and the PCM sample playback control registers cannot be accessed through DDMA Slave channel method.

When enabled, LM4560 can behave like a DDMA Slave channel device. DDMA Master will transfer the legacy DMA controller channel specific information to the related DDMA Slave channel control register when software trying to program the legacy DMA controller register.

Bit 2..1 Legacy DMA Transfer Size Control, Read Only as 00

00 8 bit transfer, legacy

Bit 3 Non Legacy Extended Addressing Control (Fully 32 bit Addressing)

0 disabled

1 enabled

Bit 31..4 DDMABase

**3.1.1.11 Legacy audio/ power management configuration:**

**PCI Configuration**

**Address:** 44h–47h

**Type:** Read/Write

**Default:** E2000000h

**legacy I/O decoding 44h**

bit 0:

0: SBBase = 0220h–022Fh

1: SBBase = 0240h–024Fh

bit 1

0: SBBase disable

1: SBBase enable

bit 2

0: ADLIBBase = 0388h–038Bh

1: ADLIBBase = 038Ch–038Fh

bit 3

0: ADLIBBase disable

1: ADLIBBase enable

bit 4

0: GAMEBase = 0200h–0207h

1: GAMEBase = 0208h–020Fh

bit 5

0: GAMEBase disable

1: GAMEBase enable

bit 6

0: MPU401Base = 0330h–0333h

1: MPU401Base = 0300h–0303h

bit 7

0: MPU401Base disable

1: MPU401Base enable

**legacy DMA decoding 45h**

bit 0

0: DMA channel 1 trapping

1: DMA channel 0 trapping

bit 1

0: DMA trapping

disable 1: DMA trapping enable

bit 2

0: DMA status handle mode A (slave only)

1: DMA status handle mode B (bus master)

bit 3

0: DMA status retry OK

1: DMA status retry error

If bit 3 is set, bus interface will not respond to IO8 operation any more unless the status retry error bit is cleared by writing 1 to this bit.

bit 4 DMAREG_RD_EN_

0: Response to DMAREG(00h–03h, 83h/87h) Read when CFG45[1] is 1;

1: Never response to DMAREG(00h–03h, 83h/87h) Read.

bits 5–7 reserved

when DMA trapping is enable, chip will decode the following I/O port

DMA channel 1 trapping

read 2,3

write snoop 2,3

write snoop 8–Fh

write snoop 83h

**DMA channel 0 trapping**

read 0,1

write snoop 0,1

write snoop 8–Fh

write snoop 87h

when DMA trapping is enable, the chip will handle DMA status read (I/O read port 8) depending on the DMA status mode bit.

DMA status handle mode A:

LM4560 will decode I/O read port 8 if StatusRDY is active, otherwise, it will ignore the cycle.

DMA status handle mode B:

## 3.0 Register Description (Continued)

When StatusRDY is not active, the chip will retry DMA status reads if it is not the current active bus master. Whenever chip retry the DMA status read from other bus master, it will also generate a bus request for the DMA status read. When the DMA status read cycle generated by chip is terminated normally, the chip will write the status data by asserting the StatusWR signal.

If chip retries a DMA status read from other bus master 3 times without getting the bus ownership or proper data, it will set the status error bit high which will terminate the pending DMA status read request internally and ignore the all DMA status read cycle by the other bus master.

When audio engine receives the StatusWR signal, it will assert the StatusRDY signal and allow the chip to decode I/O read port 8 normally. The audio engine will de-assert the StatusRDY after each DMA status read.

**Note:** All I/O decoding is 16 bits, write snooping happens only once even with multiple write retry cycles. Write snooping means chip will decode the cycle to the audio engine without generating the $\overline{\text{DEVSEL}}$ signal or $\overline{\text{TRDY}}$ on the PCI bus.

**Power Management Configuration (PM_CFG) 46h**

Bit 0 (DCC_EN) Dynamic Clock Control Enable

0: Disable

1: Enable. $\overline{\text{CLKRUN}}$ scheme will be enabled.

Bit 1 (DC_PM_EN_) Digital Controller Power Management Enable

0: Enable When enabled, Audio_clk can be shut off or turn on according to PM_ST.

1: Disable

Bit 2 (DC_RST) Digital Controller Software Reset

0: normal

1: Reset Digital Controller

Bit 3 (AC_PM_EN_) Analog CODEC Power Management Enable

0: Enable

If enabled, AC97 bit clock can be shut off according to PM_ST

1 Disable

Bit 4 (WAKE_EN1) Primary CODEC Wake-up Enable

Read/Write. Powered with Vaux. Cleared when H/W reset or S/W reset.

0: disable

1: enable

When CODEC_PD = 1, BCLK keeps low, a rising edge of ACDI1 will set WAVE_EV to high

Bit 5 (WAKE_EN2) Secondary CODEC Wake-up Enable

Read/Write. Powered with Vaux. Cleared when H/W reset or S/W reset.

0: disable

1: enable

When CODEC_PD = 1, BCLK keeps low, a rising edge of ACDI2 will set WAVE_EV to high

Bit 6 (ID_WR_EN) Chip IDs write enable

0: Vendor ID, Device ID, Subsystem vendor ID & Subsystem ID are read only

1: Vendor ID, Device ID, Subsystem vendor ID & Subsystem ID are writeable.

Bit 7 (TIMER_PME_EN) Inactivity Timer assert $\overline{\text{PME}}$ enable

0: Disable

1: Enable

If enabled, when the Inactivity Timer is expired, $\overline{\text{PME}}$ will be asserted.

**Inactivity Timer Expiration Control 47h**

bit 0–7 Inactivity timer expiration base (in seconds)

Each time when audio engine enters into D2 state, the Inactivity timer will load the base count from this register and start counting at 1s clock rate. When the MSB of the counter goes from high to low timer has expired. When not in the D2 state, the timer is reset.

**3.1.1.12 INT Acknowledge Snoop Register:**

**PCI Configuration**

**Address:** 48h–4Bh

**Default:** 00000000h

**Description:** Read/Write

Bit 0 (INTA_SNOOP_ENA) Interrupt Acknowledge Snooping Enable bit.

0: Disable

1: Enable

Bit 15–7 (INT_VEC) Interrupt Vector to be matched.

All other bits are reserved.

**3.1.1.13 Power Management Capability Register (PMC):**

**PCI Configuration**

**Address:** DCh–DFh

**Default:** 66010001h

**Description:** Read Only

Bit 7–0 (PM_Cap_ID) Power management capability identifier, read only as 01h.

Bit 15–8 (PM_Next_Ptr) Next data structure item list pointer in the PCI header, read only as 00h

Bit 31–16 (PM_CAP) Power management capability register, read only as E611h.

Bit 31–27 (PME_Support) $\overline{\text{PME}}$ supported PM_ST, read only as 01100b, indicates that $\overline{\text{PME}}$ can be asserted in D2, D3 hot.

Bit 26 (D2_Support) Read only as 1, indicates D2 supported.

Bit 25 (D1_Support) Read only as 1, indicates D1 supported.

Bit 24–22 Reserved. Read only as 000b.

Bit 21 (DSI) Device Specific Initialization. Read only as 0.

Bit 20 (Vaux) Auxiliary Power Source. Read only as 0.

Bit 19 (PME_clk) PME clock. Read only as 0, indicates that no PCI clock is required to generate $\overline{\text{PME}}$.

Bit 18–16 (Version) Read only as 001b, indicates PPMI v1.0 compliance.

**3.1.1.14 Power management control/status register (PMCSR) & PMCSR_BSE & Data:**

**PCI Configuration**

**Address:** E0h–E3h

**Default:** 00000000h

**Description:** Read/Write

Bit 31–24 (Data) Read only as 00h.

Bit 23–16 (PMCSR_BSE) Read only as 00h.

Bit 15–0 (PMCSR) Power Management Control/Status Register

Bit 15 (PME_Status) Read/Write-Clear.

## 3.0 Register Description (Continued)

0: (Default) Normal ($\overline{PME}$ is controlled by bit[8] PME_En)

1: $\overline{PME}$ can be asserted independent of bit[8] (PME_En).

Writing 0 to this bit has no effect.

Writing 1 to this bit will clear this bit, and also cause chip to stop asserting $\overline{PME}$.

Bit 14–13 (Data_Scale) Read only as 00b.

Bit 12–9 (Data_Select) Read only as 0000b.

Bit 8 (PME_En) Read/Write.

0: (Default) $\overline{PME}$ is disabled to be asserted.

1: $\overline{PME}$ is enabled to be asserted.

Bit 7–2 Reserved. Read only as 000000b.

Bit 1–0 (PM_ST) Power State. Read/Write.

Read will return current Power State, write will set to new state.

00 D0

01 D1

10 D2

11 D3 hot

**3.2 AUDIO PROCESSOR REGISTER MAP:**

| IO Offset | +3h | +2h | +1h | +0h |
|---|---|---|---|---|
| 00h | DMAR3 | DMAR2 | DMAR1 | DMAR0 |
| 04h | DMAR7 | DMAR6 | DMAR5 | DMAR4 |
| 08h | DMAR11 | DMAR10 | DMAR9 | DMAR8 |
| 0Ch | DMAR15 | DMAR14 | DMAR13 | DMAR12 |
| 10h | SBR3/SBR1 | SBR2 | SBR1/SBR3 | SBR0 |
| 14h | RSVD | SBR6 | SBR5 | SBR4 |
| 18h | RSVD | SBR7 | RSVD | RSVD |
| 1Ch | SBR10 | SBR9 | RSVD | SBR8 |
| 20h | MPUR3 | MPUR2 | MPUR1 | MPUR0 |
| 24h–2Ch | RSVD | | | |
| 30h | RSVD | RSVD | GAMER1 | GAMER0 |
| 34h | GAMER2 | | | |
| 38h | GAMER3 | | | |
| 3Ch | RSVD | | | |
| 40h | ACWR | | | |
| 44h | ACRD | | | |
| 48h | SCTRL | | | |
| 4Ch | ACGPIO | | | |
| 50h | ASR0 | | | |
| 54h | RSVD | ASR2 | RSVD | ASR1 |
| 58h | ASR3 | | | |
| 5Ch | ASR6 | ASR5 | RSVD | ASR4 |
| 60h | AOPLSR0 | | | |
| 70h | SPDIF_CS | | | |
| 74h | RSVD | | | |
| 78h | SubSystem ID | | SubVendor ID | |
| 7Ch | PCI ID read | GPcontrol | GPO | GPI |
| 80h | START_A | | | |
| 84h | STOP_A | | | |
| 88h | DLY | | | |
| 8Ch | SIGN_CSO | | | |
| 90h | CSPF_A | | | |
| 94h | CEBC | | | |
| 98h | AIN_A | | | |
| 9Ch | EINT | | | |
| A0h | GC | | | CIR |
| A4h | AINTEN_A | | | |
| A8h | MUSICVOL | | WAVEVOL | |
| ACh | SBDELTA/SBDELTA_R | | RSVD | |

## 3.0 Register Description (Continued)

| IO Offset | +3h | +2h | +1h | +0h |
|---|---|---|---|---|
| B0h | MISCINT | | | |
| B4h | START_B | | | |
| B8h | STOP_B | | | |
| BCh | CSPF_B | | | |
| C0h | SBDMAL | | SBDMAC | |
| C4h | SCE2R | RSVD | SBDD | SBCTRL |
| C8H | STIMER | | | |
| CCh | LFO_CTRL_B | LFO_CT_B | RSVD + $I^2S$_DELTA | |
| D0h | ST_TARGET | | | |
| D4h | RSVD | | | |
| D8h | AINT_B | | | |
| DCh | AINTEN_B | | | |
| Bank A: Channel Register | | | | |
| ARAM_A (CIR<32) | | | | |
| E0h | CSO | | FMS+ALPHA(11:8) | ALPHA(7:0) |
| E4h | CPTR + LBA | | | |
| E8h | ESO | | DELTA | |
| ECh | LFO_CTRL | LFO_CT | FMC+RVOL[6:1] | RVOL[0]+CVOL |
| F0h | GVSEL + PAN | VOL | CTRL + Ec(11:8) | Ec(7:0) |
| F4h | EBUF1 | | | |
| F8h | EBUF2 | | | |
| FCh | RSVD | | | |
| Bank B: Channel Register | | | | |
| ARAM_B (CIR>=32) | | | | |
| E0h | CSO | | FMS+ALPHA(11:8) | ALPHA(7:0) |
| E4h | CPTR + LBA | | | |
| E8h | ESO | | DELTA | |
| ECh | ATTRIBUTE | | FMC+RVOL[6:1] | RVOL[0]+CVOL |
| ERAM_B (CIR>=32) | | | | |
| F0h | GVSEL + PAN | LFO_INIT(Bank A) | CTRL + VOL(11:8) | VOL(7:0) |
| F4h | RSVD | | | |
| F8h | RSVD | | | |
| FCh | RSVD | | | |

**3.1.1.15 I/O Access Mode:**

**3.1.1.15.1 Audio I/O Base:**

The Audio Processor requires a 256-byte PCI I/O space. The base address (called the Audio I/O Base) resides in PCI Configuration Register 10h.

All audio processor registers (00h–FFh) can be accessed by host in an I/O cycle with address: Audio I/O Base+offset.

**3.1.1.15.2 Legacy SB I/O Base:**

If CFG_REG44h[0] is '0', SB I/O Base is 220h; if CFG_REG44h[0] is '1', SB I/O Base is 240h.

There are 16 byte registers in SB I/O Base. Physically, those registers are aliased to Audio Base Reg10h–1Fh.

**3.1.1.15.3 Legacy ADLIB I/O Base:**

If CFG_REG44h[2] is '0', ADLIB I/O Base is 388h; if CFG_REG44h[2] is '1', ADLIB I/O Base is 38Ch.

There are 4 byte registers in ADLIB I/O Base. Physically, those registers are alaised to Audio Base Reg10h–13h.

**3.1.1.15.4 Legacy Gameport I/O Base:**

If CFG_REG44h[4] is '0', Gameport I/O Base is 200h; if CFG_REG44h[4] is '1', Gameport I/O Base is 208h.

There are 8 byte registers in Gameport I/O Base. Physically, those registers are identical to Audio Base Reg30h–37h.

**3.1.1.15.5 Legacy MPU401 I/O Base:**

Controlled by CFG_REG44h[6], Audio Processor response to either of two legacy MPU401 I/O address spaces. If CFG_REG44h[6] is '0', MPU401 I/O Base is 330h; if CFG_REG44h[6] is '1', MPU401 I/O Base is 300h.

There are 4 byte registers in MPU401 I/O Base. Physically, those registers are identical to Audio Base Reg20h–23h.

## 3.0 Register Description (Continued)

**3.1.1.15.6 Legacy DMA Snooping:**

If CFG_REG45h[1] is '1', Audio Processor will trap either of Legacy DMA channel 0 or 1 by snooping legacy DMA register 00h–0Fh write cycle and response to read cycle.

Physically, these 16 byte registers are identical to Audio Base Reg00h–0fh.

**3.1.1.15.7 DDMA Base:**

Audio Processor is also a DDMA slave which has a DDMA base in CFG_REG40h and has 16 byte registers. Physically, these 16 byte registers are identical to Audio Base Reg00h–0fh.

**3.1.1.16 Memory Access Mode:**

Audio Processor requires a 4 K-byte PCI Memory space. The base address (we call it **Audio MEM Base**) resides in PCI Configuration Register 14h.

**3.1.1.16.1 Memory Mapped I/O:**

The 256 byte Audio IO space is mapped to 000h–0FFh of Audio MEM space bit to bit exactly. All audio processor registers can be accessed by host in a memory cycle with address: Audio MEM Base+offset.

**3.1.1.16.2 On-chip Memory Direct R/W:**

On-chip SRAM, such as ARAM, ERAM, can be accessed by host directly (instead of index mode). The ARAM & ERAM are mapped to 800h–FFFh.

**4K-byte Memory Address Map**

| | |
|---|---|
| 000h–0FFh | 256 byte register |
| 100h–1FFh | Reserved |
| 200h–<br>.<br>.<br>.<br>.<br>7FFh | Reserved |
| 800h–<br>.<br>.<br>.<br>.<br>FFFh | ARAM & ERAM |

**On-Chip ARAM, ERAM Address Map**

ARAM and ERAM are mapped in range of 800h–FFFh:

| | |
|---|---|
| 800h | Channel 0: REG_E0h |
| 804h | Channel 0: REG_E4h |
| 808h | Channel 0: REG_E8h |
| 80Ch | Channel 0: REG_ECh |
| 810h | Channel 0: REG_F0h |
| 814h | Channel 0: REG_F4h |
| 818h | Channel 0: REG_F8h |
| 81Ch | Channel 0: REG_FCh |
| 820h | Channel 1: REG_E0h |
| 824h | Channel 1: REG_E4h |
| 828h | Channel 1: REG_E8h |
| 82Ch | Channel 1: REG_ECh |
| 830h | Channel 1: REG_F0h |
| 834h | Channel 1: REG_F4h |
| 838h | Channel 1: REG_F8h |
| 83Ch | Channel 1: REG_FCh |
| .<br>.<br>.<br>. | .<br>.<br>.<br>. |

**3.3 AUDIO PROCESSOR REGISTER DESCRIPTIONS:**

**3.3.1 DMA Register:**

These 16-byte registers can be accessed on Audio Base (I/O or MEM), on DDMA Base or in DMA Snooping mode.

## DMA Register Map

| Name | Description |
|---|---|
| DMAR0–3 | Legacy DMA Playback Buffer Base Register |
| DMAR4–6 | Legacy DMA Playback Byte Count Register |
| DMAR7 | Legacy DMA Playback Misc. Register |
| DMAR8 | Legacy DMA Controller Command/Status Register |
| DMAR9 | Reserved Register |
| DMAR10 | Legacy DMA Single Channel Mask Port |
| DMAR11 | Legacy DMA Channel Operation Mode Register |
| DMAR12 | Legacy DMA Controller First_Last Clear Port |
| DMAR13 | Legacy DMA Controller Master Clear Port |
| DMAR14 | Legacy DMA Controller Clear Mask Port |
| DMAR15 | Legacy DMA Controller Multi-Channel Mask Register |

LDATA: Data bus

**3.4.1.1 DMAR0 (Legacy DMA Playback Buffer Base Register Port1)**

**Address:** DDMASlaveBase + 0h or AudioBase + 0h or 0000h / 0002h

**Size:** 8 bits

**Type:** Read/Write

**Default:** 00h

Write: Legacy DMA Playback Buffer Base Address 7–0

Legacy DMA Playback Buffer Current Transfer Address 7–0

Read: Legacy DMA Playback Buffer Current Transfer Address 7–0

The PCI bus interface circuit should response to I/O read to 0000h or 0002h on the PCI bus only when DMASnoopEn is active.

## 3.0 Register Description (Continued)

#### 3.4.1.2 DMAR1 (Legacy DMA Playback Buffer Base Register Port2)

**Address:** DDMASlaveBase + 1h or AudioBase + 1h or 0000h / 0002h

**Size:** 8 bits

**Type:** Read/Write

**Default:** 00h

Write: Legacy DMA Playback Buffer Base Address 15–8

Legacy DMA Playback Buffer Current Transfer Address 15–8

Read: Legacy DMA Playback Buffer Current Transfer Address 15–8

The PCI bus interface circuit should response to I/O read to 0000h or 0002h on the PCI bus only when DMASnoopEn is active.

#### 3.4.1.3 DMAR2 (Legacy DMA Playback Buffer Base Register Port3)

**Address:** DDMASlaveBase + 2h or AudioBase + 2h or 0087h / 0083h

**Size:** 8 bits

**Type:** Read/Write

**Default:** 00h

Write: Legacy DMA Playback Buffer Base Address 23–16

Legacy DMA Playback Buffer Current Transfer Address 23–16

Read: Legacy DMA Playback Buffer Current Transfer Address 23–16

The PCI bus interface circuit should response to I/O read to 0087h or 0083h on the PCI bus only when DMASnoopEn is active.

#### 3.4.1.4 DMAR3 (Legacy DMA Playback Buffer Base Register Port4)

**Address:** DDMASlaveBase + 3h or AudioBase + 3h

**Size:** 8 bits

**Type:** Read/Write

**Default:** 00h

Write: Legacy DMA Playback Buffer Base Address 31–24

Legacy DMA Playback Buffer Current Transfer Address 31–24

Read: Legacy DMA Playback Buffer Current Transfer Address 31–24

This register is intended for system which has DDMA Master.

Any time when legacy DMA is running, this register must be reset to 0 by software driver.

#### 3.4.1.5 DMAR4 (Legacy DMA Playback Byte Count Register 1)

**Address:** DDMASlaveBase + 4h or AudioBase + 4h or 0001h / 0003h

**Size:** 8 bits

**Type:** Read/Write

**Default:** 00h

Write : Legacy DMA Playback Byte Base Count 7–0

Legacy DMA Playback Current Byte Count 7–0

Read: Legacy DMA Playback Current Byte Count 7–0

The PCI bus interface circuit should response to I/O read to 0003h or 0001h on the PCI bus only when DMASnoopEn is active.

#### 3.4.1.6 DMAR5 (Legacy DMA Playback Byte Count Register 2)

**Address:** DDMASlaveBase + 5h or AudioBase + 5h or 0001h / 0003h

**Size:** 8 bits

**Type:** Read/Write

**Default:** 00h

Write: Legacy DMA Playback Byte Base Count 15–8

Legacy DMA Playback Current Byte Count 15–8

Read: Legacy DMA Playback Current Byte Count 15–8

The PCI bus interface circuit should response to I/O read to 0003h or 0001h on the PCI bus only when DMASnoopEn is active.

#### 3.4.1.7 DMAR6 (Legacy DMA Playback Byte Count Register 3)

**Address:** DDMASlaveBase + 6h or AudioBase + 6h

**Size:** 8 bits

**Type:** Read/Write

**Default:** 00h

Write: Legacy DMA Playback Byte Base Count 23–16

Legacy DMA Playback Current Byte Count 23–16

Read: Legacy DMA Playback Current Byte Count 23–16

This register is intended for system which has DDMA Master.

Any time when legacy DMA playback is not running, this register must be reset to 0 by software driver.

#### 3.4.1.8 DMAR7 (Legacy DMA Playback Misc. Register)

**Address:** DDMASlaveBase + 7h or AudioBase + 7h

**Size:** 8 bits

**Type:** Read/Write

**Default:** 00h

#### 3.4.1.9 DMAR8 (Legacy DMA Controller Command / Status Register)

**Address:** DDMASlaveBase + 8h or AudioBase + 8h or 0008h

**Size:** 8 bits

**Type:** Read Only

**Default:** 00h

Read: status register for implemented legacy 8237-A DMA channel.

Implementation of this register maintains the compatibility with legacy 8237-A status register. However, when reading this register, the return value should be different for I/O read to (DDMASlaveBase + 8h), I/O read to (AudioBase +8h) and I/O read to (0008h). I/O read to (DDMASlaveBase + 08h) is normally initiated by DDMA Master. I/O read to (AudioBase + 08h) is normally initiated by our debug program. The DDMA Master will take the responsibility to combine the return value of each DMA Slave Channel in the system and return the final resultant byte to response to the PCI I/O read to 0008h initiated by Host/PCI Bridge. The PCI bus interface circuit should response to I/O read to 0008h on the PCI bus only when DMASnoopEn is active.

## 3.0 Register Description (Continued)

#### 3.4.1.10 DMAR10 (Legacy DMA Single Channel Mask Port)

**Address:** AudioBase + 0Ah or 000Ah

**Size:** 8 bits

**Type:** Write Only

**Default:** 00h

Write: channel mask register for implemented legacy 8237-A DMA channel.

Writing to this register will affect the legacy DMA operation of the LM4560, implementation of this register maintains the register compatibility with legacy 8237-A DMA channel mask register. For system which has a DDMA Master, it is the DMA Master's responsibility to update the legacy channel mask bit, DMAR15.0 with address (DMASlaveBase + Fh) when a I/O write to 000Ah occurred on PCI Bus. When snooping a legacy 8237-A register operation is enabled, any I/O write to 000Ah should be snooped to DMAR15.0.

#### 3.4.1.11 DMAR11 (Legacy DMA Channel Operation Mode Register)

**Address:** DMASlaveBase + 0Bh or AudioBase + 0Bh or 000Bh

**Size:** 8 bits

**Type:** Read/Write

**Default:** 00h

Write: channel mode register for implemented legacy 8237-A DMA channel.

Writing to this register will affect the legacy DMA operation of the LM4560, implementation of this register maintains the register compatibility with legacy 8237-A DMA channel mode register for system with or without DDMA Master. For system which has DDMA Master, it is the DMA Master's responsibility to update this register when an I/O write to 000Bh occurred on PCI Bus. When snooping legacy 8237-A register operation is enabled, any I/O write to 000Bh should be snooped to this register if the channel number matches the snooping legacy DMA channel number.

Read : This register can only be read out through AudioBase + 0Bh port.

#### 3.4.1.12 DMAR12 (Legacy DMA Controller First_Last Flag Clear Port)

**Address:** AudioBase + 0Ch or 00Ch

**Size:** 0 bits

**Type:** Write Only

Write: first_last flag clear register for implemented legacy 8237-A DMA channel.

Writing to this register will clear the flag signal First_Last. Implementation of this register maintains the register compatibility with legacy 8237-A DMA controller for system without DDMA Master. For system which has DDMA Master, it is the DMA Master's responsibility to implement this flag. When snooping legacy 8237-A register operation is enabled, any I/O write to 000Ch should clear First_Last flag.

#### 3.4.1.13 DMAR13 (Legacy DMA Controller Master Clear Port)

**Address:** DMASlaveBase + 0Dh or AudioBase + 0Dh or 000Dh

**Size:** 0 bits

**Type:** Write Only

Write : master clear register for implemented legacy 8237-A DMA channel.

Writing to this register has the effect of hardware reset to the implemented legacy 8237-A DMA channel. Implementation of this register maintains the register compatibility with legacy 8237-A DMA controller for system with or without DDMA Master. For system which has DDMA Master, it is the DMA Master's responsibility to write to this register when a write to legacy 8237-A master clear register (I/O write to 000Dh) is on the PCI Bus. When snooping legacy 8237-A register operation is enabled, any I/O write to 000Dh should clear several legacy flags such as First_Last flag.

#### 3.4.1.14 DMAR14 (Legacy DMA Controller Clear Mask Port)

**Address:** AudioBase + 0Eh or 000Eh

**Size:** 0 bits

**Type:** Write Only

Write: multi-channel mask clear port for implemented legacy 8237-A DMA channel.

Writing to this register will affect the legacy DMA operation. Implementation of this register maintains the register compatibility with legacy 8237-A DMA multi-channel clear mask register. For system which has DDMA Master, it is the DMA Master's responsibility to update the legacy channel mask bit DMAR15.0 with address (DMASlaveBase + Fh) when a I/O write to 000Eh occurred on PCI Bus. When snooping legacy 8237-A register operation is enabled, any I/O write to 000Eh will reset DMAR15.0 to 0.

#### 3.4.1.15 DMAR15 (Legacy DMA Controller Multi-Channel Mask Register)

**Address:** DMASlaveBase + 0Fh or AudioBase + 0Fh or 000Fh

**Size:** 1 bit

**Type:** Write Only

**Default:** 0b

Write: multi-channel mask register for implemented legacy 8237-A DMA channel.

Implementation of this register maintains the register compatibility with legacy 8237-A DMA controller for system with or without DDMA Master. For system which has DDMA Master, it is the DMA Master's responsibility to write DMAR15 when a write to legacy 8237-A multi-channel mask register (I/O write to 000Fh) is on the PCI Bus. When snooping legacy 8237-A register operation is enabled, any I/O write to 000Fh should update the mask flag for the implemented legacy 8237-A DMA channel.

### 3.4.2 Legacy Sound Blaster/Adlib Register:

These 16-byte registers can be accessed on Audio Base (I/O or MEM), SB Base, or ADLIB Base.

#### 3.4.2.1 SBR0 (Legacy FmMusic Bank 0 Register Index / Legacy FmMusic Status)

**Address:** AudioBase + 10h or SBBase + 0h or SBBase + 08h or ADLIBBase + 0h

**Size:** 8 bits

**Type:** Read/Write

**Default:** 00h

**Write**

## 3.0 Register Description (Continued)

| | | |
|---|---|---|
| Bit7..0 | X | Legacy FmMusic Bank 0 Register Index |

**Read**

| | | |
|---|---|---|
| Bit 7 | 1 | FmMusic Timer Interrupt Flag (Equal to Bit 6 + Bit 5) |
| Bit 6 | 1 | FmMusic Timer 1 Overflowed Flag |
| Bit 5 | 1 | FmMusic Timer2 Overflowed Flag |
| Bit 4..0 | 0 | Reserved |

**Relative Internal Function Register File**

In order to emulate the legacy FmMusic(YMF262 or OPL3) function, a 512 bytes register file (RAM) must be implemented. By legacy access method, this register file has two banks and the bank index is specified by SBR0 and SBR2 respectively. This register file is byte-wide format, read/write RAM which has no high speed operation requirement.

**Relative Internal Functional Register Extracted From Legacy FmMusic Bank 0 Register File**

**FmMusic-TIMER1**

**Bank Index:** 02h

**Size:** 8 bits

**Type:** Read/Write

**Default:** 00h

| | | |
|---|---|---|
| Bit 7..0 | X | Timer1 Preset Value |

If enabled, Timer1 counter will increase every 1024 AC97 bitclock (12.288 MHz). When overflow occurs, this value is reloaded into the counter.

**FmMusic-TIMER2**

**Bank Index:** 03h

**Size:** 8 bits

**Type:** Read/Write

**Default:** 00h

| | | |
|---|---|---|
| Bit 7..0 | X | Timer2 Preset Value |

If enabled, Timer2 counter will increase every 4096 AC97 bitclock (12.288 MHz). When overflow occurs, this value is reloaded into the counter.

**FmMusic-Timer-CONTROL**

**Bank Index:** 04h

**Size:** 8 bits

**Type:** Read/Write

**Default:** 00h

| | | |
|---|---|---|
| Bit 7 | 1 | Reset Bit 7–5 of Legacy FmMusic Status Register |
| Bit 6 | 1 | Reset Timer1 Overflow Flag |
| Bit 5 | 1 | Reset Timer2 Overflow Flag |
| Bit 4–2 | 0 | Reserved |
| Bit 1 | 1 | Enable Timer 2 |
| Bit 0 | 1 | Enable Timer 1 |

Bit 7–5 must be self-cleared to 0 after it is written as 1.

When bit 1 or 0 is set from 0 to 1, the corresponding timer counter will load its preset value and start counting. When these bits are zero, the respective timer counter will stop counting. If bit 1 is set 1, bit 7 and 5 of FmMusic Status register will be set 1 when timer2 is overflowed. If bit 0 is set 1, bit 7 and 6 of FmMusic Status register will be set 1 when timer1 is overflowed.

**3.4.2.2 SBR1 (Legacy FmMusic Bank 0 Register Data Port)**

**Address:** AudioBase + 11h or SBBase + 1h or SBBase + 09h or ADLIBBase + 1h or AudioBase + 13h or SBBase + 3h or ADLIBBase + 3h

**Size:** 8 bits

**Type:** Read/Write

**Default:** XXh

| | | |
|---|---|---|
| Bit 7..0 | X | Legacy FmMusic Bank 0 Register (indexed by SBR0) Data |

When writing to this register, if SBR0 is B0h–B8h and bit 5 of the content (indexed by SBR0) is changed from 0 to 1 or vice versa, or SBR0 is BDh and any one of bit 4–0 of the content (indexed by SBR0) is changed from 0 to 1 or vice versa, an OPL3 Bank0 Key On/Off Dirty Flag will be set in the Audio Status Registers ASR0 and AOPLSR0.

**3.4.2.3 SBR2 (Legacy FmMusic Bank 1 Register Index)**

**Address:** AudioBase + 12h || SBBase + 2h || ADLIBBase + 2h

**Size:** 8 bits

**Type:** Read/Write

**Default:** 00h

| | | |
|---|---|---|
| Bit 7..0 | X | Legacy FmMusic Bank 1 Register Index |

**3.4.2.4 SBR3 (Legacy FmMusic Bank 1 Register Data Port)**

**Address:** AudioBase + 11h or SBBase + 1h or ADLIBBase + 1h or AudioBase + 13h or SBBase + 3h or ADLIBBase + 3h

**Size:** 8 bits

**Type:** Read/Write

**Default:** XXh

| | | |
|---|---|---|
| Bit 7..0 | X | Legacy FmMusic Bank 1 Register (indexed by SBR2) Data |

When write to this register, if SBR2 is B0h–B8h and bit 5 of the content (indexed by SBR2) is changed from 0 to 1 or vice versa, an OPL3 Bank1 Key On/Off Dirty Flag will be set in Audio Status Registers ASR0 and AOPLSR0.

**3.4.2.5 SBR4 (Legacy Sound Blaster Mixer Register Index)**

**Address:** AudioBase + 14h or SBBase + 4h

**Size:** 8 bits

**Type:** Read/Write

**Default:** 00h

| | | |
|---|---|---|
| Bit 7..0 | X | Legacy SB16 / SBPRO Mixer Register Index |

**3.4.2.6 SBR5 (Legacy Sound Blaster Mixer Register Data Port)**

**Address:** AudioBase + 15h or SBBase + 5h

**Size:** 8 bits

**Type:** Read/Write

**Default:** XXh

## 3.0 Register Description (Continued)

| | | |
|---|---|---|
| Bit 7..0 | X | Legacy SB16 / SBPRO Mixer Register (indexed by SBR4) Data Port |

#### 3.4.2.7 SBR6 (Legacy Sound Blaster ESP Reset Port)

**Address:** AudioBase + 16h or AudioBase + 17h or SBBase + 6h or SBBase + 7h

**Size:** 1 bit

**Type:** Write Only

| | | |
|---|---|---|
| Bit 0 | 1 | Enter Legacy SB16 / SBPRO ESP Reset State |
| | 0 | Escape From SB16 / SBPRO ESP Reset State |

ESP Reset should do the following things:

Reset ESP to no operation status and clear ESP Busy Flag.

b. Stop wave engine SB channel operation.

Reset any flags that may affect the next command execution.

#### 3.2.8 SBR7 (Legacy Sound Blaster ESP Data Port)

**Address:** AudioBase + 1Ah or AudioBase + 1Bh or SBBase + Ah or SBBase + Bh

**Size:** 8 bits

**Type:** Read Only

**Default:** 00h

| | | |
|---|---|---|
| Bit7..0 | 1 | Data returned by Legacy SB16 / SBPRO ESP Read Operation |

#### 3.4.2.9 SBR8 (Legacy Sound Blaster Command / Status Port)

**Address:** AudioBase + 1Ch or AudioBase + 1Dh or SBBase + Ch or SBBase + Dh

**Size:** 8 bits

**Type:** Read/Write

**Default:** 00h

**Write**

| | | |
|---|---|---|
| Bit7..0 | X | The Command (Operator) or Data (Operand) Written to Legacy SB ESP. |

**Read**

| | | |
|---|---|---|
| Bit 7 | 0 | Legacy SB ESP is Available For Next Command / Data |
| | 1 | Legacy SB ESP is Busy |
| Bit 6..0 | X | Reserved |

After the command / data has been written to the ESP Command / DATA port, bit 7 of this status register will be set to 1 (busy). After ESP has processed the written command / data and waiting for the next one, bit 7 of this status register will be reset to 0 (not busy). Any acknowledge byte must be readback before any new command is issued. ESP will be set busy after this port has ever been written and will be set not busy if the command/status has been read four times.

#### 3.4.2.10 SBR9 (Legacy Sound Blaster ESP Data Ready / IRQ Acknowledge Port 1)

**Address:** AudioBase + 1Eh or SBBase + Eh

**Size:** 1 bit

**Type:** Read Only

**Default:** 00h

| | | |
|---|---|---|
| Bit 7 | 0 | Data is not available on SBR7 |
| | 1 | Data is available on SBR7 |
| Bit 6..0 | X | Reserved |

Reading this register will clear the interrupt generated by the ESP for NON-BX type legacy SB DMA command. After SBR7 has been read, bit 7 of this register will reset to 0 (no data) until the next read data is available and set bit 7 of this register.

#### 3.4.2.11 SBR10 (Legacy Sound Blaster ESP Data Ready / IRQ Acknowledge Port 2)

**Address:** AudioBase + 1Fh or SBBase + Fh

**Size:** 1 bit

**Type:** Read Only

**Default:** 00h

| | | |
|---|---|---|
| Bit 7 | 0 | Data is not available on SBR7. |
| | 1 | Data is available on SBR7. |
| Bit 6..0 | X | Reserved |

Reading this register will clear the interrupt generated by the ESP for BX type legacy SB DMA command. After SBR7 has been read, bit 7 of this register will reset to 0 (no data). If the next read data is available at SBR7, bit 7 of this register will again be set to 1.

#### 3.4.3 Legacy MPU-401 Register

These 4-byte registers can be accessed on Audio Base (I/O or MEM), or MPU401 Base.

#### 3.4.3.1 MPUR0 (Legacy MPU-401 Data Port / IRQ Acknowledge Port)

**Address:** AudioBase + 20h or MPU401Base + 0h

**Size:** 8 bits

**Type:** Read/Write

**Default:** FEh

**Read**

| | |
|---|---|
| Bit 7..0 | MPU-401 Acknowledge Byte or External MIDI Input Data in MIDI-IN FIFO; |

**Write**

| | |
|---|---|
| Bit 7..0 | MIDI Output Data |

When internal loopback mode is enabled, reading this port will not update the MIDI-IN FIFO read counter.

#### 3.4.3.2 MPUR1 (Legacy MPU-401 Command / Status Port)

**Address:** AudioBase + 21h or MPU401Base + 1h

**Size:** 8 bits

**Type:** Read/Write

**Default:** 80h

Read MPU-401 Status

## 3.0 Register Description (Continued)

| | | |
|---|---|---|
| Bit 7 | 0 | Ack. Byte is available or External MIDI Input Data is Available in MDI-IN FIFO; |
| | 1 | NO Acknowledfe Byte or External MIDI Input Data; |
| Bit 6 | 0 | Ready for MIDI Data Output or New MIDI Command |
| | 1 | MIDI-OUT FIFO is Full |
| Bit 5 | 0 | MIDI-IN FIFO is not Full |
| | 1 | MIDI-IN FIFO is Full |
| Bit 4 | 0 | MPU401 engine is at PASS-THRU mode |
| | 1 | MPU401 engine is at UART mode |
| Bit 3..0 | | RESERVED |

Write MPU-401 Command

| | |
|---|---|
| Bit 7..0 | Command to MPU-401 MIDI Controller; |

**3.4.3.3 MPUR2 (MPU-401 Operation Control / Status Register)**

**Address:** AudioBase + 22h or MPU401Base + 2h

**Size:** 8 bits

**Type:** Bit 7..3 Read/Write
Bit 2.00 Read Only

**Default:** 10h

Read MPU-401 MIDI Engine Current Status

| | | |
|---|---|---|
| Bit 7 | 0 | MIDI-IN FIFO Source From External MIDI-IN Pad; |
| | 1 | MIDI-IN FIFO Source From MIDI-OUT FIFO; |
| Bit 6 | 0 | External MIDI-OUT Pad Source From MIDI-OUT FIFO; |
| | 1 | External MIDI-OUT Pad Source From External MIDI-IN Pad; |
| Bit 5 | 0 | Regular MIDI Clock is being used; |
| | 1 | Fast MIDI Clock (12.288 MHz) is being used; |
| Bit 4 | 0 | MPUR0 Disconnect From MIDI-OUT FIFO; |
| | 1 | Connect MPUR0 to MIDI-OUT FIFO; |
| Bit 3 | 0 | Interrupt will be generated When MIDI-IN FIFO is not Empty; |
| | 1 | Interrupt will not be generated When MIDI-IN FIFO is not Empty; |
| Bit 2 | 0 | MPU401 Midi-out buffer full flag is not masked at loop back mode; |
| | 1 | MPU401 Midi-out buffer full flag is masked at loop back mode; |
| Bit 1 | 0 | MIDI-OUT FIFO is Empty; |
| | 1 | MIDI-OUT FIFO is Not Empty; |
| Bit 0 | 0 | MIDI-IN FIFO is Empty; |
| | 1 | MIDI-IN FIFO is not Empty |

Write MPU-401 MIDI Engine Operation Control

| | | |
|---|---|---|
| Bit 7 | 0 | MIDI-IN FIFO Source From External MIDI-IN Pad; |
| | 1 | MIDI-IN FIFO Source From MIDI-OUT FIFO; |
| Bit 6 | 0 | External MIDI-OUT Pad Source From MIDI-OUT FIFO; |
| | 1 | External MIDI-OUT Pad Source From External MIDI-IN Pad; |
| Bit 5 | 0 | Regular MPU401 MIDI Clock is being used; |
| | 1 | Fast MPU401 MIDI Clock (12.288 MHz) is being used; |
| Bit 4 | 0 | Disconnect MPUR0 From MIDI-OUT FIFO When at Pass-Thru Mode; |
| | 1 | Connect MPUR0 to MIDI-OUT FIFO When at Pass-Thru Mode; |
| Bit 3 | 0 | Genarate Interrupt When MIDI-IN FIFO is not Empty; |
| | 1 | Do Not Generate Interrupt When MIDI-IN FIFO is not Empty; |
| Bit 2 | 0 | External MIDI-OUT Source from Internal MIDI-OUT |
| | 1 | Force External MIDI-OUT Output Pad to Stay at High Level |
| Bit 1..0 | | RESERVED |

**3.4.3.4 MPUR3 (MPU-401 MIDI-IN FIFO Access Port)**

**Address:** AudioBase + 23h or MPU401Base + 3h

**Size:** 8 bits

**Type:** Read Only

**Default:** 00h

Read

| | |
|---|---|
| Bit 7..0 | MIDI Data Serialized In MIDI-IN FIFO |

**3.4.3.5 Implementation Note:**

After power up reset, MPU-401 MIDI engine is at pass-through mode. MPU-401 MIDI engine will only execute the following two commands when at pass-through mode.

**Command Code:** 3Fh

**Function:** Enter_UART Mode from Pass-Through Mode

**Behavior:** Return acknowledge byte (0FEh) in MPUR0, generate an interrupt if switch to UART mode successfully. Reading MPUR0 will clear this interrupt.

**Command Code:** FFh

**Function:** MIDI Reset

**Behavior:** Return acknowledge byte (0FEh) in MPUR0, generate an interrupt and stay in Pass-Through mode. Reading MPUR0 will clear this interrupt.

MPU-401 MIDI engine will only execute the following command when at UART mode.

**Command Code:** FFh

**Function:** Enter Pass-Through Mode from UART Mode

**Behavior:** Flush MIDI-IN FIFO, go to Pass-Through Mode;

## 3.0 Register Description (Continued)

When MPU-401 MIDI engine is at internal loopback operation state (MPUR3.7 is set 1), MPUR1.7 is masked from MIDI-IN FIFO state automatically. This means if MIDI-IN FIFO is not empty, MPUR1.7 is still reading as 1. When MIDI-IN FIFO is full, MIDI clock will be stopped until MIDI-IN FIFO is not full.

#### 3.4.4 Legacy Gameport Register

These 16-byte registers can be accessed on Audio Base (I/O or MEM), or Gameport Base.

##### 3.4.4.1 GAMER0 (Gameport Control Register)

**Address:** AudioBase + 30h

**Size:** 2 bits

**Type:** Read/Write

**Default:** 0b

Bit 7

0 Disable Enhanced Digital Gameport;

1 Enable Enhanced Digital Gameport;

Bit 6

0 Disable Testmode for Enhanced Mode Gameport

1 Enable Testmode for Enhanced Mode Gameport

Bit 5..0 X Reserved;

When Bit 6 is set, the gamecounter will overflow every 1024 AC97 Bitclk. Bit 6 is only useful when Bit 7 is set. Bit 7 will be automatically cleared if there are any I/O operation to GAMER1.

##### 3.4.4.2 GAMER1 (Legacy Gameport I/O Register)

**Address:** AudioBase + 31h or GAMEBase + 0 - 7h

**Size:** 8 bits

**Type:** Read/Write

**Default:** 00h

Write

| | | |
|---|---|---|
| Bit 7..0 | X | Trigger the Legacy Gameport I/O Read Cycle |

Read

| | | |
|---|---|---|
| Bit 7 | 0 | Joystick B Button 1 Pressed (Input Pad Stay at Low Level); |
| | 1 | Joystick B Button 1 Released (Input Pad Stay at High Level); |
| Bit 6 | 0 | Joystick B Button 0 Pressed (Input Pad Stay at Low Level); |
| | 1 | Joystick B Button 0 Released (Input Pad Stay at High Level); |
| Bit 5 | 0 | Joystick A Button 1 Pressed (Input Pad Stay at Low Level); |
| | 1 | Joystick A Button 1 Released (Input Pad Stay at High Level); |
| Bit 4 | 0 | Joystick A Button 0 Pressed (Input Pad Stay at Low Level); |
| | 1 | Joystick A Button 0 Released (Input Pad Stay at High Level); |
| Bit 3 | 0 | Joystick B Y-Axies Input Pad Stay at High Level |
| | 1 | Joystick B Y-Axies Input Pad Stay at Low Level |
| Bit 2 | 0 | Joystick B X-Axies Input Pad Stay at High Level |
| | 1 | Joystick B X-Axies Input Pad Stay at Low Level |
| Bit 1 | 0 | Joystick A Y-Axies Input Pad Stay at High Level |
| | 1 | Joystick A Y-Axies Input Pad Stay at Low Level |
| Bit 0 | 0 | Joystick A X-Axies Input Pad Stay at High Level |
| | 1 | Joystick A X-Axies Input Pad Stay at Low Level |

##### 3.4.4.3 GAMER2 (Enhanced Gameport Position Register 1)

**Address:** AudioBase + 34h

**Size:** 32 bits

**Type:** Read Only

**Default:** 00000000h

Bit 31..16 Joystick A Y-Axies Position Latch Value (16 bit unsigned)

Bit 15..0 Joystick A X-Axies Position Latch Value (16 bit unsigned)

##### 3.4.4.4 GAMER3 (Enhanced Gameport Position Register 2)

**Address:** AudioBase + 38h

**Size:** 32 bits

**Type:** Read Only

**Default:** 00000000h

Bit 31..16 Joystick B Y-Axies Position Latch Value (16 bit unsigned)

Bit 15..0 Joystick B X-Axies Position Latch Value (16 bit unsigned)

##### 3.4.4.5 Implementation Note

Any I/O operation to legacy GAMEBase will function as normal gameport mode. At enhanced gameport mode (GAMER0.7 is set 1), when the 16 bit game counter overflowed, the external pad for joystick A&B X-Y Axies State will stop drived low, then game counter will counting up from zero with AC97 bitclock (12.288 MHz ). Each game position latch (4 * 16 bits) will store the game counter value when its corresponding axies state pad recharged from low to high. When the game counter overflowed, the game position latch will be set to the overflowed value (0FFFFh) if its corresponding axies state pad is still low.

**The game position latches can be read in word or double word format. These latches should be frozen when reading and be freed after reading.**

#### 3.4.5 Serial Interface Control Register:

These 16-byte registers can only be accessed on Audio Base (I/O or MEM).

##### 3.4.5.1 ACWR (AC-97 Mixer Write Register)

**Address:** AudioBase + 40h

**Size:** 32 bits

**Type:** Read/Write

## 3.0 Register Description (Continued)

**Default:** 00000000h

**Write**

| | | |
|---|---|---|
| Bit 7..0 | X | index of the AC-97 mixer register to be written; |
| | | Bit 7 = 0 for Primary CODEC; |
| | | Bit 7 = 1 for Secondary CODEC. |
| Bit 14..7 | X | reserved |
| Bit 15 | 0 | do nothing |
| | 1 | write AC-97 mixer register (indexed by bit 7..0) with bit 31..16; |
| Bit 31..16 | X | data to be written into AC-97 mixer register; |

**Read**

| | | |
|---|---|---|
| Bit 6..0 | X | index of the AC-97 mixer register to be written; |
| Bit 14..7 | X | reserved |
| Bit 15 | 0 | ready to write AC-97 mixer register |
| | 1 | busy writing AC-97 mixer (indexed by Bit 7..0); |
| Bit 31..16 | X | data to be written into AC-97 mixer register. |

#### 3.4.5.2 ACRD (AC-97 Mixer Read Register)

**Address:** AudioBase + 44h

**Size:** 32 bits

**Type:** Read/Write

**Default:** 00000000h

**Write**

| | | |
|---|---|---|
| Bit 7..0 | X | index of the AC-97 mixer register to be read; |
| | | Bit 7=0 for Primary CODEC; |
| | | Bit 7=1 for Secondary CODEC. |
| Bit 14..7 | X | reserved |
| Bit 15 | 0 | do nothing |
| | 1 | read AC-97 mixer register (indexed by bit 7..0) with bit 31..16; |
| Bit 31..16 | X | reserved |

**Read**

| | | |
|---|---|---|
| Bit 6..0 | X | index of the AC-97 mixer register to be read; |
| Bit 14..7 | X | reserved |
| Bit 15 | 0 | bit 31..16 is valid data of the AC mixer register (indexed by bit 7..0) |
| | 1 | busy reading AC-97 mixer register (indexed by Bit 7..0); |
| Bit 31..16 | X | AC-97 mixer register contents. |

#### 3.4.5.3 SCTRL (Serial INTF Control Register)

**Address:** AudioBase + 48h

**Size:** 32 bits

**Type:** Read/Write

**Default:** 00014000h

Bit 0 (WRST_CODEC) CODEC Warm Reset Command

0: Normal

1: Warm Reset CODEC

When write '1' to this bit, pin ACSYNC should be driven to high for at least 1µs.

Bit 1 (CRST_CODEC) CODEC Cold Reset Command

0: Normal

1: Cold Reset CODEC

When write '1' to this bit, pin $\overline{\text{ACRST}}$ should be driven to low for at least 1µs.

Bit 2 (MCLK_SEL) MCLK clock rate select for I2S Output

0: MCLK = 12.288M

1: MCLK = 6.144M

Bit 3 (PCMOUT_SEL) PCM Output Select (Primary/Secondary)

0: PCM Output up to Primary CODEC request

1: PCM Output up to Secondary CODEC request

Bit 4 (DBLRATE_EN) CODEC Double Rate Enable

0: Disable

1: Enable

Bit 5 (SPDIF_EN) S/PDIF Output Function Enable

0: Disable

If disabled, the clocks of SPDIF transmitter should be shut down.

1: Enable

Bit 6 (I2SOUT_EN) I2S Output Function Enable

0: Disable

If disabled, the clocks of I2S transmitter should be shut down.

1: Enable

Bit 7 (I2SIN_EN) I2S Input Function Enable

0: Disable

If disabled, the clocks of I2S receiver should be shut down.

1: Enable

Bit 8 (PCMIN_SEL) PCMIN Slot Select

0: Primary CODEC PCMIN slot input to PCMIN_A buffer

1: Secondary CODEC PCMIN slot input to PCMIN_A buffer

Bit 9 (LINE1IN_SEL) LINE1IN Slot Select

0: Primary CODEC LINE1IN slot input to LINE1IN buffer

1: Secondary CODEC LINE1IN slot input to LINE1IN buffer

Bit 10 (MIC_SEL) MIC Slot Select

0: Primary CODEC MIC slot input to MIC buffer

1: Secondary CODEC MIC slot input to MIC buffer

Bit 11 (LINE2IN_SEL) LINE2IN Slot Select

0: Primary CODEC LINE2IN slot input to LINE2IN buffer

1: Secondary CODEC LINE2IN slot input to LINE2IN buffer

Bit 12 (HSETIN_SEL) HSETIN Slot Select

0: Primary CODEC HSETIN slot input to HSETIN buffer

1: Secondary CODEC HSETIN slot input to HSETIN buffer

Bit 13 (GPIOIN_SEL) GPIOIN Slot Select

0: Primary CODEC GPIOIN slot input to GPIOIN buffer

1: Secondary CODEC GPIOIN slot input to GPIOIN buffer

Bit 15–14 Secondary CODEC ID

Default: 01

Bit 16 (PCMOUT_EN) PCMOUT L/R Slot Enable, Default: 1

0: Disable

1: Enable (Default)

Bit 17 (SURROUT_EN) SURROUT L/R Slot Enable

0: Disable

1: Enable

Bit 18 (CENTEROUT_EN) CENTEROUT Slot Enable

0: Disable

1: Enable

Bit 19 (LFEOUT_EN) LFEOUT Slot Enable

0: Disable

1: Enable

Bit 20 (LINE1OUT_EN) LINE1OUT Slot Enable

0: Disable

1: Enable

Bit 21 (LINE2OUT_EN) LINE2OUT Slot Enable

0: Disable

1: Enable (If DBLRATE_EN is 0)

Bit 22 (HSETOUT_EN) HSETOUT Slot Enable

0: Disable

1: Enable (If DBLRATE_EN is 0)

Bit 23 (GPIOOUT_EN) GPIOOUT Slot Enable

0: Disable

1: Enable (If DBLRATE_EN is 0)

Bit 24 (CODECA_RDY) Primary CODEC Ready flag (Read only)

0: Not ready

1: Ready

Bit 25 (CODECB_RDY) Secondary CODEC Ready flag (Read only)

0: Not ready

1: Ready

Bit 26 (CODEC_PD) CODEC Power Down State flag Read only.

0: Normal

1: CODEC is in power down mode

When PM_ST enters D3, this bit will be set.

Other bits are reserved

#### 3.4.5.4 ACGPIO (AC97 General Purpose IO Register)

**Address:** AudioBase + 4Ch

**Size:** 32 bits

**Type:** Read/Write

**Default:** 00000000h

Bit 0 Reseved.

Bit 1 (GP_IRQ1) Primary CODEC GPIO_INT register

This bit will be updated with Primary input Slot 12 bit 0 of every AC97 frame.

Bit 2 (GP_IRQ2) Secondary CODEC GPIO_INT register

This bit will be updated with Secondary input Slot 12 bit 0 of every AC97 frame.

Bit 3 (GP_INT1_En) Primary CODEC GPIO_INT Enable

0: Disable

1: Enable

Bit 4 (GP_INT2_En) Secondary CODEC GPIO_INT Enable

0: Disable

1: Enable

Bit 14–5 reserved

Bit 15 (COMMAND/STATUS)

This bit is status when read.

0 ready to output AC-97 Slot 12

1 busy

This bit is command when write

0 do nothing

1 output AC-97 Slot 12

Bit 31..16 (ACGPIO_OUT) data to be written into AC-97 through output Slot 12;

### 3.4.6 Misc and Status Register

These 16-byte registers can only be accessed on Audio Base (I/O or MEM).

#### 3.4.6.1 ASR0 (TSAudio Status Register)

**Address:** AudioBase + 50h

**Size:** 32 bits

**Type:** Read Only

**Default:** 00000000h

Bit 2..0 LegacyCMD

000 stop: No any operation. No contribution to Digital Mixer

001 run: Normal operation.

010 silent_DMA : SBCL will count; CA, CBC won't count.

No data fetching. No interpolation. No contribution to Digital Mixer

011 reserve

100 silent_SB: SBCL, CA & CBC will count as the same as run mode.

No data fetching. No interpolation. No contribution to Digital Mixer

101pause: SBCL, CA & CBC don't change.

let SBALPHA unchanged, CACHE_HIT=1

drive current LD (or LD_L, LD_R) to Digital Mixer

110 reserve

111 direct play: SBCL, CA & CBC don't change.

drive SBDD to Digital Mixer

Bit 3

0: SB DMA loop disable

1: SB DMA loop enable

Bit 4

0: playback

1: recording

Bit 5

0: unsigned data format

1: signed data format

Bit 6

0: mono

1: stereo

Bit 7

0: 8-bit data format

1: 16-bit data format

Bit 9..8

00: SB ESP Engine Command Port Not Busy

01: SB ESP Engine Command Port Busy

## 3.0 Register Description (Continued)

10: SB ESP DMA Test Busy

11: SB ESP Command Buffer Full

Bit 10

0: SB ESP Engine at Digital Audio Off State

1: SB ESP Engine at Digital Audio On State

Bit 11

0: SB ESP DMA Command is not valid

1: SB ESP DMA command is valid

Bit 12

0: SB ESP has no ack byte

1: SB ESP has ack byte that needs to be read out

Bit 13

0: SB ESP is not at Direct Recording Mode

1: SB ESP is at Direct Recording Mode

Bit 14

0: SB Mixer Register MX0E.1 is 0

1: SB Mixer Register MX0E.1 is 1

Bit 15

0: AC-97 codec is not ready

1: AC-97 codec is ready

Bit 16

1: OPL3 Bank

0: Key On/Off

Bit 17

1: OPL3 Bank1 Key On/Off

Bit 18

1: SB PRO Mixer Register Update

Bit 19

1: SB16 Mixer Register Update

Bit 20

1: SB Engine Sample Rate Set By Time Constant Most Recently

Bit 21

1: SB Engine Sample Rate Set By Frequency Most Recently

Bit 22

1: SB16 Command Captured Most Recently (Bx or Cx Type Command Captured)

Bit 23

1: SB PRO Command Captured Most Recently (Non-Bx or Cx Type Command Captured)

Bit 24

1: SB Mixer Soft-Reset

Bit 26..25

00: SB ESP is at get operator state

01: SB ESP is at get first operand state

11: SB ESP is at get second operand state

10: SB ESP is at get third operand state

Bit 27

1: SB ESP is at special DMA mode

Bit 28

(LREC_IRQ_MASK) Legacy Recording IRQ MASK

0: Generate IRQ when legacy recording block length expired.

1: Don't generate IRQ when legacy recording block length expired.

Bit 29 (MPU_BUF_SEL) MPU401 Output Buffer Select

0: 8-byte

1: 128-byte

Bit 31..30 (Read only) Chip Capability Bits

These two bits are connected to the invert of two bonding option pads (padt1, padt0). Padt1, padt0 are two input pad with pull-high resistor which can be bonded to GND.

00 (Default) Full functioned level (64-channel + effect + AC97 v2.0)

01 Enhanced level (64-channel + effect + AC97 v1.03)

10 Standard level (64-channel + AC97 v1.03)

11Entry level (32-channel + AC97 v1.03)

Bit 24..16 will be cleared after this register is read.

Only one bit of Bit 21 and Bit 20 can be set 1 by implemented SB ESP Engine at any time.

Only one bit of Bit 23 and Bit 22 can be set 1 by implemented SB ESP Engine at any time.

#### 3.4.6.2 ASR1 (Legacy Sound Blaster Frequency Read Back Register)

**Address:** AudioBase + 54h

**Size:** 16 bits

**Type:** Read Only

**Default:** 00h

Bit 15..0 Sample Frequency Set by SB Command 41h or 42h

#### 3.4.6.3 ASR2 (Legacy Sound Blaster Time Constant Read Back Register)

**Address:** AudioBase + 56h

**Size:** 8 bits

**Type:** Read Only

**Default:** 00h

Bit 7..0 Time Constant Value Set by SB Command 40h

#### 3.4.6.4 ASR3 ( TSAudio Scratch Register)

**Address:** AudioBase + 58h

**Size:** 32 bits

**Type:** Read/Write

**Default:** 00000000h

#### 3.4.6.5 ASR4 (TSAudio Version Control Register)

**Address:** AudioBase + 5Ch

**Size:** 8 bits

**Type:** Read Only

**Default:** 80h

#### 3.4.6.6 ASR5 (SB ESP Version High Byte Control Register)

**Address:** AudioBase + 5Eh

**Size:** 4 bits

**Type:** Read/Write

**Default:** 4h

#### 3.4.6.7 ASR6 (SB ESP Version Low Byte Control Register)

**Address:** AudioBase + 5Fh

**Size:** 4 bits

## 3.0 Register Description (Continued)

**Type:** Read/Write

**Default:** 2h

#### 3.4.7 OPL3 Channel Status Register

These 4-byte registers can only be accessed on Audio Base (I/O or MEM).

#### 3.4.7.1 AOPLSR0 (OPL3 Emulation Channel Key On/Off Trace Register)

**Address:** AudioBase + 60h

**Size:** 32 bits

**Type:** Read Only

**Default:** 00000000h

| | | |
|---|---|---|
| Bit 8–0 | 1 | Bank0 channel 8–0 key on/off event captured. |
| Bit 13–9 | 1 | OPL3 rhythm channel 4–0 key on/off event captured |
| Bit 14 | X | Reserved |
| Bit 15 | | (Whichbank) Read only |
| | 0 | Bank0 |
| | 1 | Bank1 |
| Bit 24–26 | 1 | Bank1 channel 8–0 key on/off event captured |
| Bit 31–25 | X | Reserved |

All the flags will be cleared after this register is read.

#### 3.4.8 S/PDIF & GPIO Register

This register can only be accessed on Audio Base (I/O or MEM).

#### 3.4.8.1 SPDIF_CS (S/PDIF Channel Status Register)

**Address:** AudioBase + 70h

**Size:** 32 bits

**Type:** Read/Write

**Default:** 02000000h

Bit 0 (PRO) Professional flag

Hardwired to 0

Bit 1 ($\overline{\text{Audio}}$) Audio content flag

Hardwired to 0

Bit 2 (Copy) Copyright

Read/Write, Default: 00b

Bit 5–3 (Emphasis)

Read/Write, Default: 000b

Bit 7–6 (Mode) Hardwired to 00b

Bit 15–8 (L & Category)

Read/Write, Default: 00h

Bit 19–16 (Source Num)

Read/Write, Default: 0h

Bit 23–20 (Channel Num)

Read/Write, Default: 0h

Bit 27–24(Fs) Sample rate

Hardwired to 2h (48KHz)

Bit 29–28 (Clock Acc) Clock Accuracy

Read/Write, Default: 00b

Bit 31–30 Reserved

Hardwired to 00b

#### 3.4.8.2 GPIO (General Purpose IO Register)

**Address:** AudioBase + 7Ch

**Size:** 32 bits

**Type:** Read/Write

**Default:** 00000000h

Bit 31..24 reserved

Bit 23..16 GPControl[7:0]

0: Input GPI[7:0] = GP_PIN[7:0]

1: Output GP_PIN[7:0] = GPO[7:0]

Bit 15..8 GPO[7:0]

Bit 7..0 GPI[7:0]

All reserved bits return 0 when read.

#### 3.4.9 Wave Engine Register

These 128-byte registers can be accessed on AudioBase (Audio I/O Base or Audio MEM Base).

64 voice channels are classified into two banks.

Bank A: channel 0–31 (optimized for MIDI)

Bank B: channel 32–63 (optimized for Wave, WDM Stream, DirectX buffer, $I^2S$, S/PDIF, MODEM, Handset, Recording, Microphone, Main Mixer Capture, Reverb Send, Chorus Send, AC97 SURR, AC97 CENTER/LFE)

Each channel in Bank A can only be programmed as a playback channel with individual EM (envelope modulation), individual LFO AM and individual LFO FM.

Channels in Bank B have more flexibility. Each of them can be programmed as a Normal PB channel with global LFO AM and LFO FM but without EM, or as a Special PB channel, or as a REC channel, or as a REC_PB channel. Bit[31:19] of RegEC_B is Channel ATTIBUTE.

#### 3.4.9.1 STAR_A (START Command and Status Register for Bank A)

**Address:** AudioBase + 80h

**Size:** 32 bits

**Type:** Read/Write

**Default:** 00000000h

This register and STOP_A are used as Bank A channel start/stop command register when they are written, and used as Bank A channel running/stopped status register when they are read. Bit n is for channel n.

Reading from this I/O port will return the running/stopped status of Bank A 32 voice channels.

0: Stopped.

When bit n is read as '0', it means any operation of channel n, including address generation, sample data fetching, interpolation, and envelope calculation is stopped. And this channel has no contribution to the digital mixer.

This bit will be reset from '1' to '0' in four cases.

(1) when a '1' is written to the corresponding bit in register STOP_A .

(2) when out of data, i.e. when sample loop disabled and CSO (Current Sample Offset) ≥ ESO (End Sample Offset).

when Ec (current envelope) drops down to −63.984375 dB.

when current envelope buffer is in delay-stop mode, and EDLY count down to '0'.

1: Running.

When bit n is read as '1', it means channel n is working.

This bit will be set from '0' to '1' only when a '1' is written to the corresponding bit in register START_A.

Writing to this I/O port means issuing a start command to address engine and envelope engine in expected channel.

0: Ignore.

A '0' written to bit n will not change the status of channel n.

1: Start.

A '1' written to bit n will start channel n's address engine and envelope engine and also set the status bit n to '1'.

#### 3.4.9.2 STOP_A (Channel STOP Command and Status Register for Bank A)

**Address:** AudioBase + 84h

**Size:** 32 bits

**Type:** Read/Write

**Default:** 00000000h

Reading from this I/O port will return the same value as from the last register START_A.

Writing to this I/O port means issuing a stop command to address engine and envelope engine in expected channel.

0: Ignore.

A '0' written to bit n will not change the status of channel n.

1: Stop.

A '1' written to bit n will stop channel n's address engine and envelope engine, and also reset the corresponding status bit to '0'.

#### 3.4.9.3 DLY (Delay Flag of Bank A)

**Address:** AudioBase + 88h

**Size:** 32 bits

**Type:** Read/Write

**Default:** 00000000h

When read, this register will show the delay status of each channel of Bank A. Bit n is for channel n.

0: normal

This bit will toggle from '1' to '0' when envelope engine change from a delay mode buffer to a non-delay mode buffer. *When channel n is stopped, bit n will be reset to '0'.*

1: channel is currently in delay mode (address engine kept stopped but envelope engine is running).

This bit will toggle from '0' to '1' only when envelope engine begin to deal with a delay mode buffer.

When write,

0: ignore (don't change)

1: set to '1'

#### 3.4.9.4 SIGN_CSO (Sign bit of CSO) (for Bank A only)

**Address:** AudioBase + 8Ch

**Size:** 32 bits

**Type:** Read/Write

**Default:** 00000000h

This register is used to store the sign bits of 32 channel's CSO of Bank A, with '0' means current sample address is greater than or equal to LBA(Loop Begin Address), while '1' means current sample address is less than or equal to LBA. This register can be programmed with an initial status and will be updated by address engine.

Write '0': ignore (don't change)

Write '1': set to '1'

*When channel n is stopped, bit n will be reset to '0'.*

#### 3.4.9.5 CSPF_A (Bank A Current Sample Position Flag)

**Address:** AudioBase + 90h

**Size:** 32 bits

**Type:** Read/Write

**Default:** 00000000h

This register will show a flag which indicates the Bank A's current sample is in a range between ESO/2 to ESO or in a range before ESO/2 (ESO is offset from loop begin to loop end). And this flag will be used for sample data double buffering control. Bit n is for channel n.

0: Before ESO/2

1: From ESO/2 to ESO

*When channel n is stopped, bit n will be reset to '0'.*

#### 3.4.9.6 CEBC (Current Envelope Buffer Control) (for Bank A only)

**Address:** AudioBase + 94h

**Size:** 32 bits

**Type:** Read/Write

**Default:** 00000000h

Reading from this register will return current envelope buffer flags of 32 channels of Bank A, which indicate currently envelope engine is using parameters from EBUF1 or EBUF2. Bit n is for channel n.

0: Buffer 1

1: Buffer 2

Writing '1' to bit n of this register will toggle the flag in channel n and force envelope engine to change buffer. Writing '0' to bit n won't change anything in channel.

0: Ignore

1: Toggle

*When channel n is stopped, bit n will be reset to '0'.*

#### 3.4.9.7 AINT_A (Bank A Address Engine Interrupt)

**Address:** AudioBase + 98h

**Size:** 32 bits

**Type:** Read/Write

**Default:** 00000000h

Any bits toggled from '0' to '1' will result in a IRQ.

Reading from this I/O port will return the address INT status of Bank A's 32 channels. Bit n is for channel n.

0: No INT

1: INT

This bit will be set in 2 cases:

When CSO (current sample offset) ≥ ESO (end sample offset), and ENDLP_IE (end of loop INT enable bit in Global Control register) =1 and AINTEN_A bit n is set 1 for channel n.

When CSO (current sample offset) ≥ ESO/2 (middle of ESO), and MIDLP_IE (middle of loop INT enable bit in Global Control register) =1 and AINTEN_A bit n is set 1 for channel n.

Writing '1' to bit n of this register will reset this bit.

0: Ignore.

A '0' written to bit n will not change the status of this bit.

1: reset

## 3.0 Register Description (Continued)

A '1' written to bit n will reset this bit.

#### 3.4.9.8 EINT (Envelope Engine Interrupt Eegister) (for Bank A only)

**Address:** AudioBase + 9Ch

**Size:** 32 bits

**Type:** Read/Write

**Default:** 00000000h

Any bits toggled from '0' to '1' will result in a IRQ.

Reading from this I/O port will return the envelope INT status of 32 channels of Bank A. Bit n is for channel n.

0: No INT

1: INT

This bit will be set in 2 cases:

When envelope buffer toggled, and ETOG_IE ( envelope toggle INT enable bit in Global Control register ) =1.

When Ec (current envelope) ≤ FFFh (−63.984375 dB), and EDROP_IE (envelope dropping to −63.984375 dB INT enable bit in Global Control register) =1.

Writing '1' to bit n of this register will reset this bit.

0: Ignore.

A '0' written to bit n will not change the status of this bit.

1: reset

A '1' written to bit n will reset this bit.

#### 3.4.9.9 GC & CIR (Global Control & Channel Index)

**Address:** AudioBase + A0h

**Size:** 32 bits

**Type:** Read/Write

**Default:** 00000000h

Bits 31–30 are used to control Legacy Recording channel when record to mono sample.

00: left

01: right

10: (left+right+1)/2

11: reserved.

Bits 29–28 are IO 0008-read handling control bits.

00: never assert StatusRDY

01: StatusRDY = DMATCReached

10: StatusRDY = DMATCReached | LegacyDRQ

11: in this case, handshaking with StatusWR and manipulation of return byte should been done.

StatusRDY keep '0' when initialization.

If(StatusWR == 1) {

StatusRDY = 1;

if(DMAChannel==0) {

ReturnByte[7:0]=

{InputByte[7:5], DMAR8[4], InputByte[3:1], DMAR8[0]};

}

else {

ReturnByte[7:0] =

{InputByte[7:6], DMAR8[5], InputByte[4],

InputByte[3:2], DMAR8[1], InputByte[0]};

}

}

if(DMASNOOPCS_==0 & ADR[7:0] = 8 & Data_rdy_ == 0 & StatusRDY==1)

StatusRDY = 0;

Bit 27 Test_loopback: This bit is used for wave engine loopback testing.

0: normal

1: force recording engine get new data from playback FIFO instead of aclink.

Bit 26 Debugging Mode

0: Normal

1: Chip is in Debugging Mode.

In Debugging Mode, 20 pins (including 8 pins of GPIO, 1 pin of SPDIF, 6 pins of I2S and 5 NC pins) are used as output to monitor 40 internal important signals.

Detail in Appendix B.

Bit 25–24 EXPROM Map Mode

00: 000h–1FFh of EXPROM is mapped to AudioMemBase 800h–FFFh low 16 bits;

800h–9FFh of EXPROM is mapped to AudioMemBase 800h–FFFh high 16 bits;

01: 200h–3FFh of EXPROM is mapped to AudioMemBase 800h–FFFh low 16 bits;

A00h–BFFh of EXPROM is mapped to AudioMemBase 800h–FFFh high 16 bits;

10: 400h–5FFh of EXPROM is mapped to AudioMemBase 800h–FFFh low 16 bits;

C00h–DFFh of EXPROM is mapped to AudioMemBase 800h–FFFh high 16 bits;

11: 600h–7FFh of EXPROM is mapped to AudioMemBase 800h–FFFh low 16 bits;

E00h–FFFh of EXPROM is mapped to AudioMemBase 800h–FFFh high 16 bits.

Bit 23 EXPROM Dump Mode Enable

0: Disable

1: Enable

If enabled, EXPROM(4096x12bit) is mapped to AudioMemBase according to bit[25:24], i.e. the content of EXPROM can be read out through AudioMem Read cycle.

Bit 22–21 Test mode bits

00: normal mode (chip works normally in this mode)

01: test mode 1

10: test mode 2

11: test mode 3

The detail descriptions on test mode 1, 2, and 3 are given in Appendix B.

Bit 20 Main Mixer Output Control

0 Main Mixer L/R → PCM L/R Output FIFO

1 Main Mixer L/R → MMC L/R Output Buffer

Bit 19 S/PDIF Out Control

0 S/PDIF L/R Output Buffer → S/PDIF L/R transmitter

1 PCM L/R Output FIFO → S/PDIF L/R transmitter

Bit 18 I2S Out Control

0 I2S L/R Output Buffer → I2S transmitter

1 SURR L/R Output FIFO → I2S transmitter

Bit 17 PCMIN_B Mixing Enable/Disable

0 PCMIN_B Mixing Disable

1 PCMIN_B Mixing Enable

## 3.0 Register Description (Continued)

**Note:** Controlled by PCMIN_SEL in Reg48h, either of Primary CODEC PCMIN slot or Secondary CODEC PCMIN slot will come into 3-level PCMIN_A buffer. And if PCMIN_B Mixing bit is enabled, the other slot will come into 1-level PCMIN_B buffer and will be mixed into Main Mixer.

Bit 16 64-Channel Mode

0 Legacy Mode

1 64-Channel Mode

Bit 15–10 is used for global control.

Bit 15 (EDROP_IE) is INT enable bit for current envelope dropping to −63.984375 dB.

0: disable

1: enable

Bit 14 (ETOG_IE) is INT enable bit for envelope buffer toggling.

0: disable

1: enable

Bit 13 (MIDLP_IE) is INT enable bit for middle of loop.

0: disable

1: enable

Bit 12 (ENDLP_IE) is INT enable bit for end of loop.

0: disable

1: enable

Bit 11 (UNDERUN_IE) is INT enable bit for playback underrun.

0: disable

1: enable

When playback FIFO is empty, if this bit is set as '1', an IRQ will be issued.

Bit 10 (OVERUN_IE) is INT enable bit for recording overrun.

0: disable

1: enable

When recording FIFO is full, if this bit is set as '1', an IRQ will be issued.

Bit 9 (Pause/Resume) is Pause/Resume command bit.

Read: 0: Engine hasn't been paused yet.

1: Engine has been paused already.

Write: 0: Resume Engine.

1: Pause Engine.

When host writes '1', this bit may not show '1' immediately. Engine will try to get paused as soon as possible. After engine has been paused already, this bit will be set to '1'. Once host writes '0', this bit will be reset to '0' immediately and engine will work normally.

Bit 8 (RST_Stimer) is used to reset playback sample timer counter.

When read, return 0;

write 1 will reset STimer.

Bits 5–0 (CIR) is the channel index which is used to select a channel for access. 00h selects channel 0, 1Fh selects channel 31, 3Fh selects channel 63.

All other bits are reserved.

#### 3.2.9.10 AINTEN_A (Bank A Address Engine Interrupt Enable)

**Address:** AudioBase + A4h

**Size:** 32 bits

**Type:** Read/Write

**Default:** 00000000h

This register will control address engine interrupt for each channel of Bank A. Bit n is for channel n.

0: disable address engine interrupt for channel n

1: enable address engine interrupt for channel n

#### 3.4.9.11 MUSICVOL & WAVEVOL (Global Music Volume & Global Wave Volume)

**Address:** AudioBase + A8h

**Size:** 32 bits

**Type:** Read/Write

**Default:** 00008080h

MUSICVOL (Bit 3–16) is global music left/right volume in format of 6.2

Bit 23–16 music left volume

0 0 dB (no attenuation)

FFh −63.75 dB (mute)

Bit 31–24 music right volume

0 0dB (no attenuation)

FFh −63.75 dB (mute)

WAVEVOL (Bit 15–0) is global wave left/right volume in format of 6.2

Bit 7–0 wave left volume

0 0 dB (no attenuation)

80h –32 dB (default)

FFh −63.75 dB (mute)

Bit 15–8 wave right volume

0 0 dB (no attenuation)

80h –32 dB (default)

FFh − 63.75 dB (mute)

#### 3.4.9.12 SBDELTA/DELTA_R (Sample Change Step for Legacy Playback & Recording)

**Address:** AudioBase + ACh

**Size:** 32 bits

**Type:** Read/Write

**Default:** 00000000h

Bit15–0 (SBDELTA/SBDELTA_R): SBDELTA: Fs/F48k in 4.12 format.

SBDELTA_R: F48k/Fs in 4.12 format.

Bit 31–16: Reserved.

#### 3.4.9.13 MISCINT (Miscellaneous Int & Status)

**Address:** AudioBase + B0h

**Size:** 32 bits

**Type:** Read/Write

**Default:** 00000000h

Bit[7:0] (read only) are interrupt request bits. All of these six bits form signal AUDIO_INT.

AUDIO_INT = PB_UNDERUN_IRQ | REC_OVERUN_IRQ | SB_IRQ | MPU401_IRQ | OPL3_IRQ | ADDRESS_IRQ | ENVELOPE_IRQ | ST_IRQ | ACGPIO_IRQ

Bit 0 (PB_UNDERUN_IRQ) is playback FIFO underrun IRQ bit. Active high.

Bit[0] = UNDERUN_IE & Bit[8].

Bit 1 (REC_OVERUN_IRQ) is recording overrun IRQ bit. Active high.

Bit[1] = OVERUN_IE & Bit[9].

Bit 2 (SB_IRQ) is sound blaster IRQ bit. Active high.

Bit[2] = sbirq (signal from Legacy Audio block)

Bit 3 (MPU401_IRQ) is MPU401 IRQ bit. Active high.

Bit[3] = mpu401irq (signal from Legacy Audio block)

Bit 4 (OPL3_IRQ) is OPL3 timer IRQ bit. Active high.

Bit[4] = timerirq & opltimer_ie

Bit 5 (ADDRESS_IRQ) is Wave-table Address Engine IRQ bit. Active high.

Bit[5] = ( | AINT_A[31:0] ) | ( | AINT_B[31:0] )

Bit 6 (ENVELOPE_IRQ) is Wave-table Envelope Engine IRQ bit. Active high.

Bit[6] = | EINT[31:0]

Bit 7 (ST_IRQ) is Sample Timer IRQ bit. Active high.

Bit[7] = ST_IRQ_En | ST_TARGET_REACHED

Bit [9:8] (read/write) are current status bits of wave-table & legacy audio engine.

Bit 8 (PB_UNDERUN) is playback FIFO underrun status bit. Active high.

This bit will be set to '1' if playback is running & FIFO is empty & f48 clock is coming.

Bit 9 (REC_OVERUN) is recording overrun status bit. Active high.

This bit will be set to '1' if recording is running & rec_req_ is active & data_rdy haven't come.

Bit 10 (mixer_underflow_flag) is a flag which indicates the result of mixer accumulator is less than 80000h.

This bit will be set to '1' once accumulator underflows.

Write '1' will clear this bit.

Bit 11 (mixer_overflow_flag) is a flag which indicates the result of mixer accumulator exceeds 7FFFFh.

This bit will be set to '1' once accumulator overflows.

Write '1' will clear this bit.

Bit 15 (ST_TARGET_REACHED) is a flag with '1' indicates STIMER counter has been equal to ST_TARGET.

This bit will be set to '1' once STIMER counter is equal to ST_TARGET.

Write '1' will clear this bit.

Bit 16 (PB_24K_MODE) is playback 48k/24k mode control bit.

0: (default) Wave engine drives sample to CODEC at 48 kHz

1: Wave engine drives sample to CODEC at 24 kHz (in this mode, Delta should be programmed twice as that in 48Khz mode).

Bit 17 (opltimer_ie) is OPL3 timer interrupt enable bit.

0: disable

1: enable

Bit 23 (ST_IRQ_En) is ST IRQ enable bit.

0: disable

1: enable

Bit 24 (ACGPIO_IRQ) is AC97 GPIO interrupt request.

ACGPIO_IRQ = Reg4Ch[1] & Reg4Ch[3] | Reg4Ch[2] & Reg4Ch[4].

All other bits are reserved bits.

#### 3.4.9.14 STAR_B (START Command and Status Register for Bank B)

**Address:** AudioBase + B4h

**Size:** 32 bits

**Type:** Read/Write

**Default:** 0000h

This register and STOP_B are used as Bank B channel start/stop command register when they are written, and used as Bank B channel running/stopped status register when they are read. Bit n is for channel n.

Reading from this I/O port will return the running/stopped status of Bank B 32 voice channels.

0: Stopped.

When bit n is read as '0', it means any operation of channel n, including address generation, sample data fetching, interpolation, and envelope calculation is stopped. And this channel has no contribution to the digital mixer. This bit will be reset from '1' to '0' in four cases.

(1) when a '1' is written to the corresponding bit in register STOP_B.

(2) when out of data, i.e. when sample loop disabled and CSO (Current Sample Offset) ≥ ESO (End Sample Offset).

when Ec (current envelope) drops down to −63.984375 dB.

when current envelope buffer is in delay-stop mode, and EDLY count down to '0'.

1: Running.

When bit n is read as '1', it means channel n is working. This bit will be set from '0' to '1' only when a '1' is written to the corresponding bit in register START_B.

Writing to this I/O port means issuing a start command to address engine and envelope engine in expected channel.

0: Ignore.

A '0' written to bit n will not change the status of channel n.

1: Start.

A '1' written to bit n will start channel n's address engine and envelope engine and also set the status bit n to '1'.

#### 3.4.9.15 STOP_B (Channel STOP Command and Status Register for Bank B)

**Address:** AudioBase + B8h

**Size:** 32 bits

**Type:** Read/Write

**Default:** 0000h

Reading from this I/O port will return the same value as from the last register START_B.

Writing to this I/O port means issuing a stop command to address engine and envelope engine in expected channel.

0: Ignore.

A '0' written to bit n will not change the status of channel n.

1: Stop.

A '1' written to bit n will stop channel n's address engine and envelope engine, and also reset the corresponding status bit to '0'.

#### 3.4.9.16 CSPF_B (Bank B Current Sample Position Flag)

**Address:** AudioBase + BCh

**Size:** 32 bits

**Type:** Read/Write

**Default:** 00000000h

## 3.0 Register Description (Continued)

This register will show a flag which indicates the Bank B's current sample is in a range between ESO/2 to ESO or in a range before ESO/2 (ESO is offset from loop begin to loop end). And this flag will be used for sample data double buffering control. Bit n is for channel n.

0: Before ESO/2

1: From ESO/2 to ESO

When channel n is stopped, bit n will be reset to '0'.

#### 3.4.9.17 SBBL & SBCL (Sound Blaster Base Block Length & Current Block Length)

**Address:** AudioBase + C0h

**Size:** 32 bits

**Type:** Read/Write

**Default:** 00000000h

SBBL(Bit 31–16) is sound blaster base block length

SBCL(Bit 15–0) is current value of sound blaster block length counter

If sound blaster DMA loop is enabled (SBCTRL[3]=1), every time when SBCL changed from 0 to FFFFh, a INT will be issued, the contents of SBCL is reloaded from SBBL, and DMA operation continues.

If sound blaster DMA loop is not enabled (SBCTRL[3]=0), every time when SBCL changed from 0 to FFFFh, a INT will be issued, the contents of SBCL is reloaded from SBBL, and set LegacyCMD to 101(pause).

SBCTRL bit 7 is used to determine the counter operation mode (byte count or word count). The counter is a count down counter.

#### 3.4.9.18 SBCTRL & SBE2R & SBDD (Sound Blaster Control)

**Address:** AudioBase + C4h

**Size:** 32 bits

**Type:** Read/Write

**Default:** 00000000h

**Description** SBE2R (Sound Blaster DMA Testing Byte Data Register)

SBDD (Sound Blaster Direct Playback Date Register)

SBCTRL (Sound Blaster Control)

SBE2R(Bit 31–24) is sound blaster DMA testing byte command data port (write only)

Any time after Bit 31–24 has ever been written, E2Status (source from wave engine) will be set high. E2Status will be cleared after the testing byte has been sent to the system location.

SBDD(Bit 15–8) is sound blaster direct mode playback data port

SBCTRL(Bit 7–0) is legacy sound blaster voice in/out control register

Bit 7

0 8 bit data format

1 16 bit data format

Bit 6

0 mono

1 stereo

Bit 5

0 unsigned data format

1 signed data format

Bit 4

0 playback

1 recording

Bit 3 sound blaster DMA loop enable control

0 loop disabled.

1 loop enabled.

Bit 2..0 LegacyCMD

000 stop: No any operation. No contribution to Digital Mixer

001 run: Normal operation.

010 silent_DMA : SBCL will count; CA, CBC won't count. No data fetching. No interpolation. No contribution to Digital Mixer

011 reserve

100 silent_SB : SBCL, CA & CBC will count as the same as run mode. No data fetching. No interpolation. No contribution to Digital Mixer

101 pause: SBCL, CA & CBC don't change.

let SBALPHA unchanged, CACHE_HIT=1 drive current LD (or LD_L, LD_R) to Digital Mixer

110 reserve

111 Direct_playback: SBCL, CA & CBC don't change.

drive SBDD to Digital Mixer

All other bits are reserved bits.

#### 3.4.9.19 STimer (Playback Sample Timer)

**Address:** AudioBase + C8h

**Size:** 32 bits

**Type:** Read/Write

**Default:** 00000000h

Bit 31–0 (STimer) will show current state of the sample timer counter which will count up every f48k clock and will be reset when RST_Stimer bit being written. Active high.

#### 3.4.9.20 LFO_B And I2S_DELTA (Bank B Low Frequency Oscillator Control)

**Address:** AudioBase + CCh

**Size:** 32 bits

**Type:** Read/Write

**Default:** 00000000h

Bit 31–27 Reserved–Read Only 00000b

Bit 26–16 is used for Bank B LFO control.

Bit 26 (LFO_E_B) is Bank B LFO enable bit.

0: disabled

1: enabled

Bits 25–24 (LFO_R_B) is clock rate select of Bank B LFO counter.

00: LFO counter clock rate is 48 kHz

01: LFO counter clock rate is 48 kHz/4

10: LFO counter clock rate is 48 kHz/16

11: LFO counter clock rate is 48 kHz/64

Bits 23–16 (LFO_INIT_B)is the initial value of the Bank B LFO counter which will count down to 0 then reload.

Bit 15–13 reserved.

Bit 12–0 (I2S_DELTA) (Read only) This register returns the auto-detected DELTA of I2S input ($f_{i2s}/f_{48k}$).

## 3.0 Register Description (Continued)

**3.4.9.21 ST_TARGET (Sample Timer Target)**

**Address:** AudioBase + D0h

**Size:** 32 bits

**Type:** Read/Write

**Default:** 00000000h

Bit 31–0 (ST_TARGET) is used to store a pre-set value. Once STIMER counter reaches that value, an IRQ called ST_IRQ will be issued if ST_IRQ_En = 1.

**3.4.9.22 AINT_B (Bank B Address Engine Interrupt)**

**Address:** AudioBase + D8h

**Size:** 32 bits

**Type:** Read/Write

**Default:** 00000000h

Any bits toggled from '0' to '1' will result in a IRQ.

Reading from this I/O port will return the address INT status of Bank B's 32 channels. Bit n is for channel n.

0: No INT

1: INT

This bit will be set in 2 cases:

When CSO (current sample offset) ≥ ESO (end sample offset), and ENDLP_IE (end of loop INT enable bit in Global Control register) =1 and AINTEN_B bit n is set 1 for channel n.

When CSO (current sample offset) ≥ ESO/2 (middle of ESO), and MIDLP_IE (middle of loop INT enable bit in Global Control register) =1 and AINTEN_B bit n is set 1 for channel n.

Writing '1' to bit n of this register will reset this bit.

0: Ignore.

A '0' written to bit n will not change the status of this bit.

1: reset

A '1' written to bit n will reset this bit.

**3.4.9.23 AINTEN_B(Bank B Address Engine Interrupt Enable)**

**Address:** AudioBase + DCh

**Size:** 32 bits

**Type:** Read/Write

**Default:** 00000000h

This register will control address engine interrupt for each channel of Bank B. Bit n is for channel n.

0: disable address engine interrupt for channel n

1: enable address engine interrupt for channel n

**3.4.9.24 E0h (CS0 & ALPHA & FMS) (for Bank A & Bank B)**

This register can be accessed in index mode or direct access mode.

**Address:** AudioBase + E0h (index mode) || Audio MEM Base + 800h + 20h*CIR (direct access mode, CIR: channel index)

**Size:** 32 bits

**Type:** Read/Write

**Default:** XXXXXXXXh

**Description:** CSO — Current Sample Offset (16 bits)

ALPHA — Sample interpolation coefficient (12 bits)

FMS — Frequency Modulation Step (4 bit)

Bits 31–16 (CSO) is the offset of current sample relative to loop begin sample.

Bits 15–4 (ALPHA) is sample interpolation coefficient, which stands for the linear interpolation ratio between current sample and the next one.

Bits 3–0 (FMS) is Frequency Modulation Step.

**3.4.9.25 E4h (LBA) (for Bank A & Bank B)**

This register can be accessed in index mode or direct access mode.

**Address:** AudioBase + E4h (index mode) || Audio MEM Base + 804h + 20h*CIR (direct access mode, CIR: channel index)

**Size:** 32 bits

**Type:** Read/Write

**Default:** XXXXXXXXh

**Description:** Loop Begin Address (31 bits) & CPTR — Cache Pointer (1 bit)

Bit 31 (CPTR) is reserved for internal use of cache control.

Bits 30–0 is the linear address of loop begin sample.

It should be word aligned when sample type is 16-bit Mono or 8-bit Stereo; and should be double word aligned when sample type is 16-bit Stereo.

**3.4.9.26 E8h (ESO & DELTA) (for Bank A & Bank B)**

This register can be accessed in index mode or direct access mode.

**Address:** AudioBase + E8h (index mode) || Audio MEM Base + 808h + 20h*CIR (direct access mode, CIR: channel index)

**Size:** 32 bits

**Type:** Read/Write

**Default:** XXXXXXXXh

**Description:** ESO — End Sample Offset (16 bits)

DELTA — Sample rate ratio (16 bits)

Bits 31–16 (ESO) is the offset of loop end sample relative to loop begin sample.

Bits 15–0 (DELTA) is sample change step in format 4.12 (Four bits integer, 12 bits fraction), which stands for the frequency ratio: Fs/48 kHz, while Fs is the sum of sample rate and pitch shifting rate.

**3.4.9.27 ECh_A (Bank A LFO_CTRL & LFO_CT & FMC & RVOL & CVOL) (Bank A Only)**

This register can be accessed in index mode or direct access mode.

**Address:** AudioBase + ECh (index mode) || Audio MEM Base + 80Ch + 20h*CIR (direct access mode, CIR: channel index) (CIR<32)

**Size:** 32 bits

**Type:** Read/Write

**Default:** XXXXXXXXh

**Description:** LFO_CTRL — Per Channel LFO Control (8 bit)

LFP_CT — Per Channel LFP working counter (8 bits)

FMC — Per Channel FM control (2 bit)

RVOL — Reverb Send Linear Volume (7 bit)

CVOL — Chorus Send Linear Volume (7 bit)

Bit[31:28] (SIN) Sine wave value.

Bit[27] (SIN_S) sign bit of sine wave.

0: positive

1: negative

Bit[26] (SIN_D) SIN counter direction bit.

0: up

1: down

Bit[25:24] (LFO_R) LFO counter clock rate select bits.

00: 48 kHz

01: 48 kHz/4

10: 48 kHz/16

11: 48 kHz/64

Bit[23:16] (LFO_CT) LFO working counter.

Bit[15:14] (FMC) FM modulation control bits.

00: FMA = (FMS * SIN) >> 3

01: FMA = (FMS * SIN) >> 2

10: FMA = (FMS * SIN) >> 1

11: FMA = (FMS * SIN) >> 0

Bit[13:7] (RVOL) Reverb Send Linear Volume

format: 1.6, 7Fh stands for 2x gain, 40h stands for no gain no attenuation, 00h stands for mute.

Bit[6:0] (CVOL) Chorus Send Linear Volume

format: 1.6, 7Fh stands for 2x gain, 40h stands for no gain no attenuation, 00h stands for mute.

#### 3.4.9.28 ECh_B (Bank B ATTRIBUTE & FMC & RVOL & CVOL) (Bank B Only)

This register can be accessed in index mode or direct access mode.

**Address:** AudioBase + ECh (index mode) || Audio MEM Base + 80Ch + 20h*CIR (direct access mode, CIR: channel index) (CIR≥32)

**Size:** 32 bits

**Type:** Read/Write

**Default:** XXXXXXXXh

**Description:** FMC — Per Channel FM control (2 bit)

RVOL — Reverb Send Linear Volume (7 bit)

CVOL — Chorus Send Linear Volume (7 bit)

Bit[31:19] (ATTRIBUTE) Channel attribute

Bit[31:30] PB/REC Select

00 (Normal PB) Normal playback

This is a normal playback channel in Bank B with Global Volume, Channel Volume, PAN, SRC, FM/AM features. In this case, bit[29:19] doesn't matter.

01 (Special PB) Special playback

This channel can be one of several kinds of special playback channels. Bit[29:26] is used to select special playback type; bit[24:25] is used to select data flow from channel to FIFO; and bit[23:19] is used to enable/disable individual functions.

10 (REC) Recording to system memory

This channel can be one of several kinds of recording channels. Bit[29:26] is used to select recording type; bit[25:24] is used to control how MONO sample is generated when recording; bit[23] is used to enable/disable SRC; bit[22:19] doesn't matter.

11 (REC_PB) Recording to system memory and playback to mixer

This channel is a Recording channel which records sample data to system memory and playback to Main Mixer in the mean time. In this case, bit[29:26] is used to select recording type; bit[25:24] is used to control how MONO sample is generated when recording; and bit[23:19] is used to enable/disable individual functions.

Bit[29:26] Channel Type Select

when Bit[31:30] is 00: (Normal PB)

xxxx reserved

when Bit[31:30] = 01: (Special PB)

0000 playback to MODEM LINE1 Output FIFO

0001 playback to MODEM LINE2 Output FIFO

0010 playback to PCM L/R Output FIFO

0011 playback to HSET Output FIFO

0100 playback to I2S L/R Output Buffer

0101 playback to CENTER/LFE Output FIFO

0110 playback to SURR L/R Output FIFO

0111 playback to SPDIF L/R Output FIFO

other reserved

when Bit[31:30] = 1x: (REC or REC_PB)

0000 recording from MODEM LINE1 Input FIFO 0001

recording from MODEM LINE2 Input FIFO

0010 recording from PCM L/R Input FIFO

0011 recording from HSET Input FIFO

0100 recording from I2S L/R Input FIFO

0101 recording from MIC Input FIFO

0110 main mixer capture from PCM L/R Output FIFO

0111 main mixer capture from MMC L/R Output Buffer

1000 Reverb Send

1001 Chorus Send

other reserved

BIT[25:24]Special Playback Channel to FIFO data flow select / Recording to MONO control

When channel is in Special PB mode, this register is used to select input source of a stereo playback slot pairs such as PCM L/R, SURR L/R, CENTER/LFE, I2S L/R and SPDIF L/R. The input source of L/R can be from one channel or can be from two independent channels.

When channel is in REC or REC_PB mode, this register is used to control how MONO sample is generated.

when Bit[31:30] = 00 (Normal PB)

xx never used

when Bit[31:30] = 01 (Special PB)

00 Channel L/R to FIFO L/R

In this case, channel is acting as a stereo channel, data flow is like

Channel Left → FIFO Left

Channel Right → FIFO Right

01 Channel L to FIFO L

Data flow:

Channel Left → FIFO Left

10 Channel R to FIFO R

Data flow:

Channel Right → FIFO Right

11 reserved

## 3.0 Register Description (Continued)

when Bit[31:30] = 1x (REC or REC_PB)

00: left,

01: right

10: (left+right+1)/2

11: reserved.

Bit[23] SRC Enable

0 disable

1 enable

Bit[22] FM and AM Enable

0 disable

1 enable

Bit[21] PAN Enable

0 disable

1 enable

Bit[20] Channel Volume Enable

0 disable

1 enable

Bit[19] Global Volume Enable

0 disable

1 enable

Bit[18:16] reserved

Bit[15:14] (FMC) FM modulation control bits.

00: FMA = (FMS * SIN) >> 3

01: FMA = (FMS * SIN) >> 2

10: FMA = (FMS * SIN) >> 1

11: FMA = (FMS * SIN) >> 0

Bit[13:7] (RVOL) Reverb Send Linear Volume

format: 1.6, 7Fh stands for 2x gain, 40h stands for no gain no attenuation, 00h stands for mute.

Bit[6:0] (CVOL) Chorus Send Linear Volume

format: 1.6, 7Fh stands for 2x gain, 40h stands for no gain no attenuation, 00h stands for mute.

#### 3.4.9.29 F0h_A (Bank A GVSEL & PAN & VOL & CTRL & Ec) (for Bank A only)

This register can be accessed in index mode or direct access mode.

**Address:** AudioBase + F0h (index mode) || Audio MEM Base + 810h + 20h*CIR (direct access mode, CIR: channel index) (CIR<32)

**Size:** 32 bits

**Type:** Read/Write

**Default:** XXXXXXXXh

**Description:** GVSEL — Global Volume Select (1 bit)
PAN — Positioning Attenuation (7 bits)
VOL — Volume Attenuation (8 bits)
CTRL — Control (4 bits)
Ec — Curent Envelope (12 bit)

Bit 31 (GVSEL) is global volume select bit.

0: select MUSICVOL

1: select WAVEVOL

Bits 30–24 (PAN) is Positioning attenuation control.

Bit 30 selects attenuated channel.

0: left,

1: right.

Bits 29–24 is the attenuation value in format of 4.2. 3Fh stand for mute.

Bits 23–16 (VOL) is channel volume attenuation in format of 5.3. 00h stands for 0 dB attenuation, FFh stands for mute.

Bits 15–12 are control bits.

Bit 15 selects 8/16 bit sample data

0: 8-bit data

1: 16-bit data

Bit 14 selects mono/stereo sample data

0: mono

1: stereo

Bit 13 selects unsigned/signed sample data

0: unsigned

1: signed

Bit 12 is loop mode enable bit.

0: disable

1: enable

Bit 11–0 (Ec) is current envelope in format of 6.6 (Six bits integer and six bits fraction). 00h stands for 0 dB, FFh stands for −63.984375 dB.

#### 3.4.9.30 F0h_B (Bank B GVSEL & PAN & VOL & CTRL & Bank A LFO_INIT)

This register can be accessed in index mode or direct access mode.

**Address:** AudioBase + F0h (index mode) || Audio MEM Base + 810h + 20h*CIR (direct access mode, CIR: channel index) (CIR<32)

**Size:** 32 bits

**Type:** Read/Write

**Default:** XXXXXXXXh

**Description:** GVSEL — Global Volume Select (1 bit)
PAN — Positioning Attenuation (7 bits)
VOL — Volume Attenuation (12 bits)
CTRL — Control (4 bits)
LFO_INIT — Bank A per channel LFO counter initial and reload value(8 bit)

Bit 31 (GVSEL) is global volume select bit.

0: select MUSICVOL

1: select WAVEVOL

Bits 30–24 (PAN) is Positioning attenuation control.

Bit 30 selects attenuated channel.

0: left,

1: right.

Bits 29–24 is the attenuation value in format of 4.2. 3Fh stand for mute.

Bits 23–16 (LFO_INIT) is Bank A per channel LFO counter initial and reload value.

**Note:** Any time when host write to RegECh[26:16] (LFO_CT), LFO_INIT should be written with the same value.

Bits 15–12 are control bits.

Bit 15 selects 8/16 bit sample data

0: 8-bit data

1: 16-bit data

Bit 14 selects mono/stereo sample data

0: mono

1: stereo

## 3.0 Register Description (Continued)

Bit 13 selects unsigned/signed sample data

0: unsigned

1: signed

Bit 12 is loop mode enable bit.

0: disable

1: enable

Bit 11–0 (VOL) is channel volume attenuation in format of 6.6. 000h stands for 0 dB attenuation, FFFh stands for mute.

#### 3.4.9.31 F4h_A (EBUF1) (Bank A Only)

This register can be accessed in index mode or direct access mode.

**Address:** AudioBase + F4h (index mode) || Audio MEM Base + 814h + 20h*CIR (direct access mode, CIR: channel index) (CIR<32)

**Size:** 32 bits

**Type:** Read/Write

**Default:** XXXXXXXXh

**Description:** Envelope Buffer 1

This register and next one provide envelope double buffer.

Bit 31–30 (AMS_H) is Amplitude Modulation Step High part.

Bits 29–28 (EMOD) define operation mode.

00: DEC mode (ramp from 0 dB to −64 dB)

In this mode, bits 7–0 of this register are used as ECNT which stores current state of a 8-bit counter; bits 15–8 of this register are used as EINIT which provides initial value of that 8-bit counter; bits 27–16 of this register are used as EAMT which is the absolute ramping amount with range from 0 dB to 63 and 63/64 dB. Every 48 kHz clock, ECNT decrease 1; every time when ECNT=00h, it reload EINIT, EAMT decrease 1, and Ec decrease 1; every time when EAMT=00h, envelope engine will toggle buffer flag in global register CEBC.

01: INC mode ( ramp from −64 dB to 0 dB )

In this mode, the layout of this register is completely the same as in DEC mode. Engine works in the same way except that the ramp direction is from −64 dB to 0 dB.

10: Delay mode

In this mode, bits 27–26 are used to select sub-mode:

00: Delay_hold

01: Delay_start

10: Delay_stop

11: reserved

19–0 is used as EDLY which store the current state of a 20-bit delay counter, bits 25–20 are of no use. Every 48 kHZ clock, EDLY decrease 1. During all the time this buffer active, Ec keep unchanged.

In Delay_hold sub-mode, when EDLY =00000h, engine will toggle current buffer flag in global register CEBC.

In Delay_start sub-mode, when EDLY =00000h, engine will reset DLY flag register.

In Delay_stop sub-mode, when EDLY =00000h, engine will reset start/stop flag register.

11: Still mode

In this mode, Ec keep unchanged, buffer never toggle automatically. Only when CEBC is written, buffer may toggle.

#### 3.4.9.32 F8h_A (EBUF2) (Bank A Only)

This register can be accessed in index mode or direct access mode.

**Address:** AudioBase + F8h (index mode) || Audio MEM Base + 818h + 20h*CIR (direct access mode, CIR: channel index) (CIR<32)

**Size:** 32 bits

**Type:** Read/Write

**Default:** XXXXXXXXh

**Description:** Envelope Buffer 2

EBUF2 is totally as the same as EBUF1 except that bits 31–30 are AMS_L (Amplitude Modulation Step Low part).

## 4.0 Functional Description

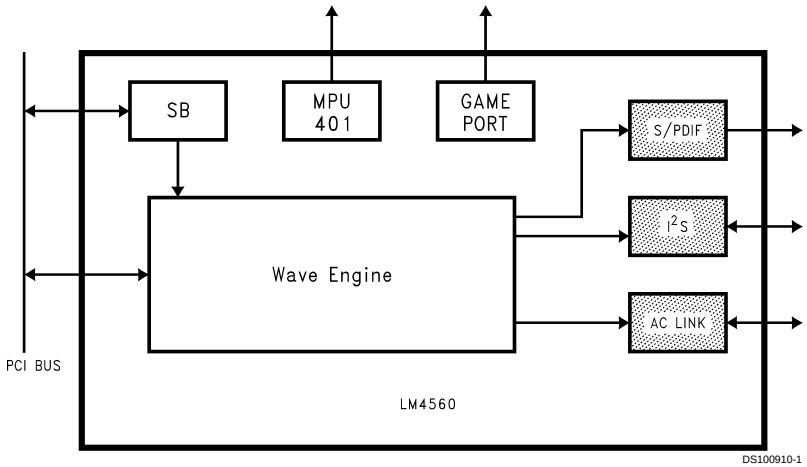



**FIGURE 2. LM4560 Block Diagram**

## 4.0 Functional Description (Continued)

### 4.1 WAVE ENGINE

**Feature:**

- 64 voice synthesis
- Each voice channel typically consumes 0.25% PCI bandwidth
- 8/16 bits, mono/stereo, unsigned/signed samples at arbitrary sample rate
- Unlimited length of wave samples
- Internal 26 bits computation, DAC driven up to 20 bits
- Double buffering envelope curve
- Pitch shifting
- 0–6 Hz low frequency oscillator
- FM/AM modulation
- Chorus, Reverb effect
- Recording/Playback full-duplex
- Main Mixer capture
- Microphone channel capture
- $I^2S$ interface
- S/PDIF interface
- Modem data rate conversion

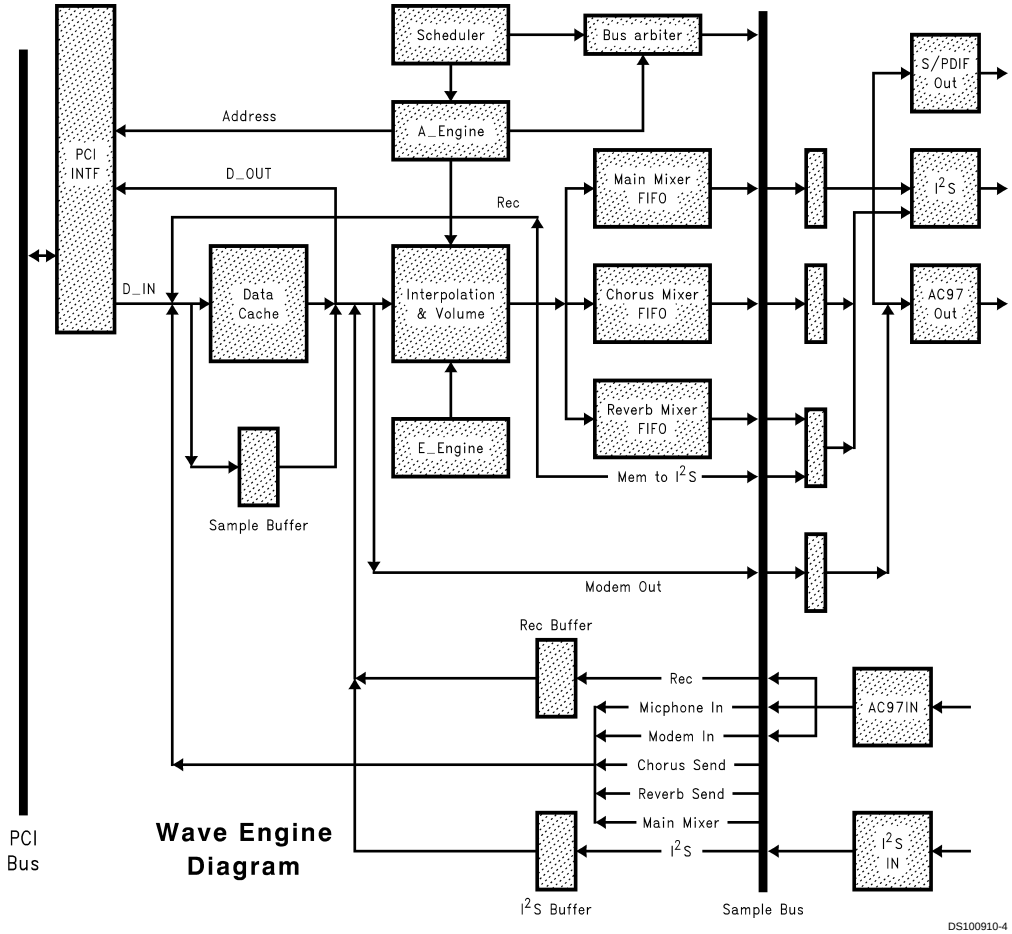



**FIGURE 3. Testing Specification Standard**

#### 4.1.1 Scheduler

Wave engine process SRC (Sample Rate Conversion), EM (Envelope Modulation), FM (Frequency Modulation), AM (Amplitude Modulation) and DM (Digital Mixing) channel by channel. A central controller called *Scheduler* will decide the processing sequence of up to 64 playback channels (including up to 10 special channels) and 1 legacy playback/recording channel.

To optimize PCI bus utilization, Scheduler decides each channel's priority according to the following rules:

- Legacy playback/recording channel always has the highest priority.
- Channels which require PCI bus cycle, i.e. cache miss channels, have higher priority.

- Channels which don't require PCI bus cycle, i.e. cache hit channels and I2StoMIX channel, have lower priority.
- Channels which depend on other channel's result, i.e. Main Mixer Capture channel, Chorus Send channel and Reverb Send channel, have the lowest priority.

Basically, all operation of one channel include address generation, data reading/writing, interpolation, per channel LFO, low frequency FM/AM, Envelope calculation, PAN, volume adjusting and mixer accumulation. We call it Channel Operation. After all channel's operation are done, the mixing result will be sent to FIFO. We call this Mixing Loop.

Before each Mixing Loop, Scheduler scans cache hit/miss flags of all active channels to decide the processing sequence of the current Loop.

#### 4.1.2 Address Engine

Address generation, data reading/writing, per channel LFO, low frequency FM and interpolation coefficient calculation are processed by Address Engine.

Terms:

CCI (6 bits) — Current Channel Index

CPTR (1 bit) — Cache Pointer

ESO (16 bits) — Ending Sample Offset (relative to loop begin sample)

CSO (16 bits) — Current Sample Offset (relative to loop begin sample)

CSO_SIGN (1 bit) — Sign bit of CSO

CAO (18 bits) — Current Address Offset (relative to loop begin address)

LBA (32 bits) — Loop Begin Address

CSA (32 bits) — Current Sample Address

FMS (4 bits) — Frequency Modulation Step

FMC (2 bit) — Frequency Modulation Control

SIN (4 bits) — current state of sin counter

FMA (8 bits) — Frequency Modulation Amount

BL (2 bits) — Burst read Length

DELTA (15 bits) — Sample increment number per 48 kHz clock (format 3.12)

ALPHA (12 bits) — Interpolation coefficient, i.e. fractional part of CSO

NEWCSO (16 bits) — the next CSO

NEWALPHA (12 bits) — the next ALPHA

The procedure of address engine is described as following:

(1) Load RegE0h // read FMS, CSO, ALPHA from ARAM

Load RegE4h // read LBA from ARAM

if (8 bit && MONO) CAO = CSO;

else if (16 bit && STEREO) CAO = CSO << 2;

else CAO = CSO << 1;

FMA = SIN*FMS;

(3) Load RegE8h // read ESO, DELTA, CPTR from ARAM

if(Cache miss)

{ // calculate address CSA = LBA + (CSO_SIGN==0) ? CAO : −CAO; // 32 bit addition }

BL is decided by CSA[1:0] and data_type[2:1];

(data_type[2:0] refers to bit[15:13] of channel register F0h)

| BL | CSA[1:0]=00 | CSA[1:0]=01 | CSA[1:0]=10 | CSA[1:0]=11 |
|---|---|---|---|---|
| Data_type[2:1]=00 | 1 | 1 | 1 | 2 |
| Data_type[2:1]=01 | 1 | 2 | 2 | 2 |
| Data_type[2:1]=10 | 1 | 2 | 2 | 2 |
| Data_type[2:1]=11 | 2 | 3 | 3 | 3 |

**BL (Burst Length)**

Update RegE8h with new CPTR

Update RegECh with new LFO_CTRL & LFO_CT

## 4.0 Functional Description (Continued)

**4.1.3 Envelope engine**

Low frequency AM, Envelope calculation and PAN are processed by Envelope Engine.

Terms:

Ec (12 bits) — Current envelope value in format of 6.6
AMS_H (2 bits) — Higher 2 bits of AMS
AMS_L (2 bits) — Lower 2 bits of AMS
AMS (4 bits) — Amplitude Modulation Step in format of 0.004 (right aligned with Ec)
SIN (4 bits) — current state of sin counter
AMA (8 bits) — Amplitude Modulation Amount in format of 2.6
VOL (8 bits) — Volume Attenuation per channel in format of 5.3
GVSEL (1 bit) — Global volume select bit
MUSICVOL_L (8 bits) — global music volume attenuation Left (6.2)
MUSICVOL_R (8 bits) — global music volume attenuation Right (6.2)
WAVEVOL_L (8 bits) — global wave volume attenuation Left (6.2)
WAVEVOL_R (8 bits) — global wave volume attenuation Right (6.2)
Eva (12 bits) — Current envelope after Volume attenuation
ATTNU_L (8 bits) — Left attenuation.
ATTNU_R (8 bits) — Right attenuation.
PAN (7 bits) — Positioning Attenuation (4.2 with one select bit)
EL (12 bits) — Left channel envelope value after PAN(6.6)
ER (12 bits) — Right channel envelope value after PAN(6.6)
VL (12 bits) — Left channel volume after anti-log operation
VR (12 bits) — Right channel volume after anti-log operation
EMOD (3 bits) — Envelope operation mode
ECNT (8 bits) — Envelope ramp counter
EINIT (8 bits) — Initial value of ECNT
EAMT (12 bits) — Envelope ramp amount
EDLY (20 bits) — Envelope delay counter
CEBF (1 bit) — Current Envelope Buffer Flag

**4.1.4 Interpolation & Volume Adjusting**

This multiply-add block process interpolation & volume adjusting.

Terms:

D1 (16 bits) — Current sample data (in MONO mode)
D1_L (16 bits) — Current sample data Left (in Stereo mode)
D1_R (16 bits) — Current sample data Right (in Stereo mode)
D2 (16 bits) — Sample data next to D1 (in MONO mode)
D2_L (16 bits) — Sample data next to D1_L (in Stereo mode)
D2_R (16 bits) — Sample data next to D1_R (in Stereo mode)
D (16 bits) — Data after interpolation (in MONO mode)
D_L (16 bits) — Data after interpolation Left (in Stereo mode)
D_R (16 bits) — Data after interpolation Right (in Stereo mode)
VL (12 bits) — Dry volume (Left)
VR (12 bits) — Dry volume (Right)
CVOL(7 bits) — Chorus Send Volume
RVOL(7 bits) — Reverb Send Volume
VDl (signed 20 bits) — volume adjusted Dry data (left)
VDr (signed 20 bits) — volume adjusted Dry data (right)
CSDl (signed 17 bits) — Chorus Send data (left)
CSDr (signed 17 bits) — Chorus Send data (right)
RSDl (signed 17 bits) — Reverb Send data (left)
RSDr (signed 17 bits) — Reverb Send data ( right )

**1) Data conversion:**

## 4.0 Functional Description (Continued)

If sample is stereo, there are two data inputs — D_L and D_R, if sample is mono, only one data input — D. All this data should be converted to 16-bit signed format.

Conversion from unsigned data to signed data should be implemented by inverting the MSB.

| Decimal | 16-bit unsigned | 16-bit signed |
|---|---|---|
| 32767 | FFFFh | 7FFFh |
| 2 | 8002h | 0002h |
| 1 | 8001h | 0001h |
| 0 | 8000h | 0000h |
| −1 | 7FFFh | FFFFh |
| −2 | 7FFEh | FFFEh |
| −32768 | 0000h | 8000h |

**conversion from unsigned data to signed data**

Conversion from 8-bit data to 16-bit data should be implemented by scaling data range from [−128, 127] to [−32768, 32767], i.e. adding eight '0' to LSB.

| Decimal in | 8-bit unsigned | 8-bit signed | Decimal out | 16-bit signed |
|---|---|---|---|---|
| 127 | FFh | 7Fh | 32512 | 7F00h |
| 2 | 82h | 02h | 512 | 0200h |
| 1 | 81h | 01h | 256 | 0100h |
| 0 | 80h | 00h | 0 | 0000h |
| −1 | 7Fh | FFh | −256 | FF00h |
| −2 | 7Eh | FEh | −512 | FE00h |
| −128 | 00h | 80h | −32768 | 8000h |

**conversion from 8-bit data to 16-bit data**

#### 4.1.5 Digital Mixer

There are three Digital Mixer — Main, Chorus and Reverb.

Terms:

ADl — (signed 26 bits) Main Mixer accumulator data (left channel)

ADr — (signed 26 bits) Main Mixer accumulator data (right channel)

CADl — (signed 23 bits) Chorus Mixer accumulator data (left channel)

CADr — (signed 23 bits) Chorus Mixer accumulator data (right channel)

RADl — (signed 23 bits) Chorus Mixer accumulator data (left channel)

RADr — (signed 23 bits) Chorus Mixer accumulator data (right channel)

(1) Main Mixer:

20-bit VDl, VDr should be expanded to 26-bit before addition.

Before ADl, ADr are sent to M_FIFO, 26-bit data should be converted to 20-bit with overflow and underflow consideration.

(2) Chorus Mixer:

17-bit CSDl, CSDr should be expanded to 23-bit before addition.

Before CADl, CADr are sent to C_FIFO, 23-bit data should be converted to 16-bit with overflow and underflow consideration.

17-bit RSDl, RSDr should be expanded to 23-bit before addition.

Before RADl, RADr are sent to R_FIFO, 23-bit data should be converted to 16-bit with overflow and underflow consideration.

#### 4.1.6 LFO

LFO (Low Frequency Oscillator) is used to generate a low frequency sine wave, with which the sample wave is modulated in both frequency domain and amplitude domain.

Each channel of Bank A has it's own LFO, but all Bank B channels share one global LFO.

An 8-bit programmable counter (LFO counter) is employed for frequency control. It's clock rate can be selected to be 48 kHz, 12 kHz, 3 kHz or 750 Hz .

A 4-bit incremental/decremental counter (SIN counter) is used to generate sine wave — sin[3:0] with one sign bit — SIN_S. Everytime an 8-bit LFO counter counts down to '0', the SIN counter will count up or down once and the LFO counter will reload LFO_INIT.

The SIN counter works like the following:

| SIN[3:0] | SIN_S |
|---|---|
| 0 | 0 |
| 1 | 0 |
| 2 | 0 |
| 3 | 0 |
| 4 | 0 |
| 5 | 0 |
| 6 | 0 |
| 7 | 0 |
| 8 | 0 |
| 9 | 0 |
| A | 0 |
| B | 0 |
| C | 0 |
| D | 0 |
| E | 0 |
| F | 0 |
| E | 0 |
| D | 0 |
| C | 0 |
| B | 0 |
| A | 0 |
| 9 | 0 |
| 8 | 0 |
| 7 | 0 |
| 6 | 0 |
| 5 | 0 |
| 4 | 0 |
| 3 | 0 |
| 2 | 0 |
| 1 | 0 |
| 0 | 1 |
| 1 | 1 |
| 2 | 1 |
| 3 | 1 |
| 4 | 1 |
| 5 | 1 |
| 6 | 1 |
| 7 | 1 |
| 8 | 1 |
| 9 | 1 |
| A | 1 |
| B | 1 |
| C | 1 |
| D | 1 |
| E | 1 |
| F | 1 |
| E | 1 |
| D | 1 |
| C | 1 |

| SIN[3:0] | SIN_S |
|---|---|
| B | 1 |
| A | 1 |
| 9 | 1 |
| 8 | 1 |
| 7 | 1 |
| 6 | 1 |
| 5 | 1 |
| 4 | 1 |
| 3 | 1 |
| 2 | 1 |
| 1 | 1 |

**One cycle of SIN counter**

#### 4.1.7 Recording

A down sampling SRC is used for recording.

#### 4.1.8 PCI Buffer/Sample Cache

PCI Buffer (or called Sample Cache) is a 128x32 bit SRAM which is designed to reduce consumption of PCI bus bandwidth. Each wave channel has two double words of buffer. The playback channel uses it as a pre-fetch buffer, while the recording channel uses it as a post-write buffer.

#### 4.1.9 CODEC Buffer/Sample FIFO

Between the Wave Engine and the CODEC, there's another SRAM (128x20 bit) called the CODEC Buffer (or the Sample FIFO) which is for synchronization between wave engine and serial interface for optimization of PCI bus utilization.

This 128-word CODEC buffer is partitioned into several sections:

| Section | Size | Address | Description |
|---|---|---|---|
| **PLAYBACK** | | | |
| PCM_L Out FIFO | | | PCM Left Playback FIFO |
| PCM_R Out FIFO | | | PCM Right Playback FIFO |
| SURR_L Out FIFO | | | Surround Left Playback FIFO |
| SURR_R Out FIFO | | | Surround Right Playback FIFO |
| CENTER Out FIFO | | | Center Playback FIFO |
| LFE Out FIFO | | | LFE Playback FIFO |
| LINE1 Out FIFO | | | MODEM Line1 Output FIFO |
| LINE2 Out FIFO | | | MODEM Line2 Output FIFO |
| HSET Out FIFO | | | Handset Output FIFO |
| I2S_L Out Buffer | | | I2S Left Output Buffer |
| I2S_R Out Buffer | | | I2S Right Output Buffer |
| SPDIF_L Out Buffer | | | S/PDIF Left Output Buffer |
| SPDIF_R Out Buffer | | | S/PDIF Right Output Buffer |
| **RECORDING** | | | |
| PCM_L In Buffer | | | PCM Left Recording Buffer |
| PCM_R In Buffer | | | PCM Right Recording Buffer |
| LINE1 In Buffer | | | Modem Line1 Input Buffer |
| LINE2 In Buffer | | | Modem Line2 Input Buffer |
| MIC In Buffer | | | Microphone Input Buffer |
| HSET In Buffer | | | Handset Input Buffer |
| REVERB_L SEND Buffer | | | Reverb Left Send Buffer |
| REVERB_R SEND Buffer | | | Reverb Right Send Buffer |
| CHORUS_L SEND Buffer | | | Chorus Left Send Buffer |
| CHORUS_R SEND Buffer | | | Chorus Right Send Buffer |

**Table CODEC Buffer Partitioning**

#### 4.1.10 Legacy Channel Playback/Recording

This function block is intended to emulate legacy DMA using PCI bus master engine.

All parameters used by this Legacy Channel are stored in register.

**TABLE 1. LBL (Legacy Burst Length)**

| LBL CA[1:0] | 00 | 01 | 10 | 11 |
|---|---|---|---|---|
| SB_Data_type[2:1]=00 | 1 | 1 | 1 | 2 |
| SB_Data_type[2:1]=01 | 1 | 2 | 2 | 2 |
| SB_Data_type[2:1]=10 | 1 | 2 | 2 | 2 |
| SB_Data_type[2:1]=11 | 2 | 3 | 3 | 3 |

**TABLE 2. LBL1 (Legacy Burst Length)**

| LBL1 BA[1:0] | 00 | 01 | 10 | 11 |
|---|---|---|---|---|
| SB_Data_type[2:1]=00 | 1 | 1 | 1 | 1 |
| SB_Data_type[2:1]=01 | 1 | 1 | 1 | 2 |
| SB_Data_type[2:1]=10 | 1 | 1 | 1 | 2 |
| SB_Data_type[2:1]=11 | 1 | 2 | 2 | 2 |

### 4.2 LEGACY AUDIO

The legacy Audio includes the SB Engine, OPL3 trapping, MPU-401 interface and Gameport.

#### 4.2.1 SB Engine

##### 4.2.1.1 Legacy SB ESP I/O Procedure

###### 4.2.1.1.1 SB ESP Read Data Procedure

###### 4.2.1.1.2 SB ESP Write Procedure

###### 4.2.1.1.3 SB ESP Reset Procedure

###### 4.2.1.1.4

###### 4.2.1.1.5 Legacy SB Interrupt Acknowledge Procedure

##### 4.2.1.2 Legacy Sound Blaster Compatible Digitized Sound Transfer Method

###### 4.2.1.2.1 Direct Mode

SB ESP is programmed to do audio input/output on each command. All delay time is controlled by CPU delay loop or timer interrupt. Only 8-bit mono input/output is supported.

Input:

a. ESP_Write(20h)

b. Next 8-bit mono PCM = ESP_Read()

c. Wait until next sample time, go to a.

###### 4.2.1.2.2 DMA Mode

- Normal DMA Mode

  SB ESP is programmed to make one transfer with a specified block size. At the end of transfer, the SB ESP

will generate an interrupt and wait for next command.

- Continuous DMA Mode

  SB ESP is programmed to make continuous transfer to/from CODEC. After each transfer of a specified block size, the SB ESP will generate an interrupt and continue the next transfer of the same block size after the interrupt is acknowledged.

  There are two ways to terminate continuous DMA mode transfer.

  Program the SB ESP to switch to normal DMA mode transfer. At the end of the current DMA transfer, the SB ESP will exit from continuous DMA mode and continue to transfer using the specified normal DMA mode.

  Send the exit continuous command. The SB ESP will exit continuous DMA mode at the end of current block and terminate the transfer.

- Special DMA Mode

  Once SB ESP is in the special DMA mode, it will not accept any further commands or data until the DMA mode is terminated by the following two methods:

  Non-continuous special DMA mode will exit special DMA mode automatically at the end of transfer.

  For continuous special DMA mode, a SB ESP_Reset() is needed to exit special DMA mode. The SB ESP_Reset() will only stop special DMA transfer, all other parameters remain the same.

**4.2.1.3 Legacy Sound Blaster Compatible Digitized Sound Sampling Rate Setup**

**4.2.1.3.1**

**4.2.1.3.2 Set Sampling Rate By Time Constant**

Time constant = 256d − (1,000,000d / (channel * sampling rate))

**4.2.1.3.3 Set Sampling Rate By Sampling Frequency**

Frequency = 4 kHz to 48 kHz, either mono or stereo

**4.2.1.4 Legacy Sound Blaster Esp Engine Digital Audio Command Set**

0xh Reserved

1xh Set Audio Output Mode for 8 Bit Playback

2xh Set Audio Input Mode for 8 Bit Recording

3xh Reserved

4xh Set Sample Rate and Continuos/Special DMA Block Length

5xh Reserved

6xh Reserved

7xh SB ADPCM Playback

8xh Output Silence

9xh 8 Bit Special DMA Mode Playback/Record

Axh Reserved

Bxh 16-Bit DMA Audio I/O

Cxh 8-Bit DMA Audio I/O

Dxh Control DMA and Speaker

Exh SB ESP Version and Diagnostic Test

Fxh SB ESP Test

All 8-bit SB ESP command is unsigned PCM playback/recording command except Cxh command.

All 8-bit SB ESP command is mono playback/recording command except command 14h, 1Ch , 9xh and Cxh

**4.2.1.5 8-Bit Playback Only Command 1Xh**

SB ESP uses SBMX0E.1 to do mono/stereo playback.

SBMX0E.1 = 0 for mono playback

1 for stereo playback

**4.2.1.5.1 Command Bit Definition**

Bit 30 Direct or Normal DMA Mode

1 Continuous DMA Mode

Bit 20 Direct Mode

1 DMA Mode

Bit 10 8-Bit Data PCM

1 Reserved

Bit 00 Normal

1 Reserved

**4.2.1.5.2 Command 10h**
Function: 8-bit direct mode playback

**4.2.1.5.3 Command 14h**
Function: 8-bit normal DMA playback
Length = number of bytes to be transferred - 1
SB ESP will generate an interrupt after the specified length of data has been transferred.

**4.2.1.5.4 Command 16h/17h**
Function: SB ADPCMDMA playback
Length = number of bytes to be transferred - 1
These two commands are not supported.

**4.2.1.5.5 Command 1Ch**
Function: 8-bit continuous DMA playback
SB ESP will generate an interrupt after each block length of data has been transferred.

**4.2.1.5.6 Command 1Eh/1Fh**
Function: SB ADPCMDMA playback
These two commands are not supported.

**4.2.1.6 8-Bit Recording Only Command 2xh**
SB ESP uses SBMX0E.1 to do mono/stereo recording.
SBMX0E.1 = 0 for mono recording.
1 for stereo recording.

**4.2.1.6.1 Command Bit Definition**
Bit 3
0 Direct Mode or Normal DMA Mode
1 Continuous DMA Mode
Bit 2
0 Direct Mode
1 DMA Mode
Bit 1 X reserved
Bit 0 X reserved

**4.2.1.6.2 Command 20h**
Function: 8-bit direct mode recording

**4.2.1.6.3 Command 24h**
Function: 8-bit normal DMA recording
Length = number of bytes to be transferred - 1
SB ESP will generate an interrupt after the specified length of data has been transferred.

**4.2.1.6.4 Command 2Ch**
Function: 8-bit continuous DMA recording
SB ESP will generate an interrupt after each block length of data has been transferred.

**4.2.1.7 Set Sample Rate And Continuous/Special Dma Block Length Command 4xh**

**4.2.1.7.1 Command 40h**
Function: Set Sample Rate by Time Constant
Time_Constant = 256d − (1,000,000d / (Channel * Sampling Rate))
Channel = 1 for mono or SB16 command, 2 for stereo of SBPRO command

**4.2.1.7.2 Command 41h**

Function: Set Playback Sample Rate by Frequency

Sampling Frequency = 4 kHz to 48 kHz, either mono or stereo

**4.2.1.7.3 Command 42h**

Function: Set Recording Sample Rate by Frequency

Sampling Frequency = 4 kHz to 48 kHz, either mono or stereo

**4.2.1.7.4 Command 48h**

Function: Set Block Length for Continuous and Special DMA

Length = number of bytes to be transferred - 1

SB ESP will generate an interrupt after the specified block length of data has been transferred.

**4.2.1.8 SB ADPCM Playback Only Command 7xh**

**4.2.1.8.1 Command 72–77h**

Function: SB ADPCM normal DMA playback

Length = number of bytes to be transferred - 1

These commands are not supported, the implemented SB ESP engine will behave like its normal operation except silent the output and do not perform the bus mastering operation.

**4.2.1.8.2 Command 7A–7Fh**

Function: SB ADPCM continuous DMA playback

These commands are not supported, the implemented SB ESP engine will behave like its normal operation except silent the output and do not perform the bus mastering operation.

**4.2.1.9 Output Silence Command 8xh**

**4.2.1.9.1 Command 80h**

Function: Silence Digital Audio for a Duration

Duration = number of sample rate count - 1

After the specified duration elapses, SB ESP will generate an interrupt. During silence period, SB ESP output middle range data to CODEC (8000h).

**4.2.1.10 Special DMA Playback/Recording Command 9xh**

All special DMA mode use command 48h to set the transfer block size. The non-continuous special DMA mode will interrupt the CPU at the end of the transfer block and wait for new command. Use SB ESP_Reset( ) to end the continuous special DMA, all other parameters remain the same after the SB ESP_Reset( ).

SB ESP uses SBMX0E.1 to do mono/stereo playback.

SBMX0E.1 = 0 for mono playback

1 for stereo playback

MX0E.1 default value is 0 after power up reset or mixer reset

SB ESP uses command A0h/A8h to set mono/stereo recording state.

Mono recording after SB ESP command A0h

Stereo recording after SB ESP command A8h

Default recording state after power up reset or ESP_Reset() is mono.

**4.2.1.10.1 Command Bit Definition**

Bit 3

0 Special DMA Playback

1 Special DMA Recording

Bit 2 X reserved

Bit 1 X reserved

Bit 0

0 Continuous DMA

1 Non-Continuous DMA

**4.2.1.10.2 COMMAND 90H**

Function: 8-bit continuous special DMA playback

SB ESP will generate an interrupt after each block length of data has been transferred.

**4.2.1.10.3 Command 91h**

Function: 8-bit non-continuous special DMA playback

SB ESP will generate an interrupt after the specified block length of data has been transferred.

For non-continuous special DMA playback, every time when SB ESP receives the 91h command, SB ESP will use the block length that most recently set by command 48h to begin the special DMA transfer.

**4.2.1.10.4 Command 98h**

Function: 8-bit continuous special DMA recording

SB ESP will generate an interrupt after the specified block length of data has been transferred.

**4.2.1.10.5 Command 99h**

Function: 8-bit non-continuous special DMA recording

SB ESP will generate an interrupt after the specified block length of data has been transferred.

**4.2.1.11 16-Bit Audio Playback/Recording Command Bxh**

**4.2.1.11.1 Command Bit Definition**

Bit 3

0 Playback

1 Recording

Bit 2

0 Non-Continuous DMA

1 Continuous DMA

Bit 1 X Reserved

Bit 0 X Reserved

**4.2.1.11.2 Command B0h/B1h/B2h/B3h/B4h/B5h/B6h/B7h**

Function: 16-bit audio playback

Mode

Bit 7..6 X reserved

Bit 5

0 mono

1 stereo

Bit 4

0 unsigned PCM (0–8000h–FFFFh)

1 signed PCM (8000h–0–7FFFh)

Bit 3..0 X reserved

Length = number of words to be transferred -1

For non-continuous DMA, SB ESP will generate an interrupt and terminate the DMA transfer after the specified length of data has been transferred.

For continuous DMA, SB ESP will generate an interrupt after each specified block length of data has been transferred.

**4.2.1.11.3 Command B8h/B9h/BAh/BBh/BCh/BDh/BEh/BFh**

Function: 16-bit audio recording

Mode

Bit 7..6 X reserved

Bit 5

0 mono

1 stereo

Bit 4

0 unsigned PCM (0–8000h–FFFFh)

1 signed PCM (8000h–0–7FFFh)

Bit 3..0 X reserved

Length = number of words to be transferred -1

For non-continuous DMA, SB ESP will generate an interrupt and terminate the DMA transfer after the specified length of data has been transferred.

For continuous DMA, SB ESP will generate an interrupt after each specified block length of data has been transferred.

#### 4.2.1.12 8-Bit Audio Playback/Recording Command Cxh

##### 4.2.1.12.1 Command Bit Definition

Bit 3

0 Playback

1 Recording

Bit 2

0 Non-Continuous DMA

1 Continuous DMA

Bit 1 X Reserved

Bit 0 X Reserved

##### 4.2.1.12.2 Command C0h/C1h/C2h/C3h/C4h/C5h/C6h/C7h

Function: 8-bit audio playback

Procedure:

1. SB ESP_Write(C0h/C1h/C2h/C3h/C4h/C5h/C6h/C7h)
2. SB ESP_Write(Mode)
3. SB ESP_Write(Length.low)
4. SB ESP_Write(Length.high)

Mode

Bit 7..6 X reserved

Bit 5

0 mono

1 stereo

Bit 4

0 unsigned PCM (0–80h–FFh)

1 signed PCM (80h–0–7Fh)

Bit 3..0 X reserved

Length = number of bytes to be transferred -1

For non-continuous DMA, SB ESP will generate an interrupt and terminate the DMA transfer after the specified length of data has been transferred.

For continuous DMA, SB ESP will generate an interrupt after each specified block length of data has been transferred.

##### 4.2.1.12.3 Command C8h/C9h/CAh/CBh/CCh/CDh/CEh/CFh

Function: 8-bit audio recording

Mode

Bit 7..6 X reserved

Bit 5

0 mono

1 stereo

Bit 4

0 unsigned PCM (0–80h–FFh)

1 signed PCM (80h–0–7FFh)

Bit 3..0 X reserved

Length = number of words to be transferred -1

For non-continuous DMA, SB ESP will generate an interrupt and terminate the DMA transfer after the specified length of data has been transferred.

For continuous DMA, SB ESP will generate an interrupt after each specified block length of data has been transferred.

#### 4.2.1.13 8/16-Bit Audio Dma Operation Control Command Dxh

##### 4.2.1.13.1 Command D0h

Function: Pause Non-Bx Type Command DMA Transfer

The DMA request is stopped after this command. Internal FIFO will continue running until the FIFO is empty (playback) or full (record). The DMA request will resume after command D4h or any of new DMA command is captured by SB ESP. This command is no use to Bx type command DMA transfer.

##### 4.2.1.13.2 Command D1h

Function: Set Digital Audio On Status Flag for D8 Command

##### 4.2.1.13.3 Command D3h

Function: Set Digital Audio Off Status Flag for D8 Command

Digital audio status flag is off after system reset or SB ESP_Reset().

##### 4.2.1.13.4 Command D4h

Function: Resume Non-Bx Type Command DMA Transfer

The DMA request that is suspended by the command D0h is enabled again. The internal FIFO is working as usual. This command is no use to Bx type command DMA transfer.

##### 4.2.1.13.5 Command D5h

Function: Pause Bx Type Command DMA Transfer

The DMA request is stopped after this command. Internal FIFO will continue running until the FIFO is empty (playback) or full (record). The DMA request will resume after command D6h or any of new DMA command is captured by SB ESP. This command is no use to non-Bx type command DMA transfer.

##### 4.2.1.13.6 Command D6h

Function: Resume Bx Type Command DMA Transfer

The DMA request that is suspended by the command D5h is enabled again. The internal FIFO is working as usual. This command is no use to non-Bx type command DMA transfer.

##### 4.2.1.13.7 Command D8h

Function: Get Digital Audio Status Flag

##### 4.2.1.13.8 Command D9h

Function: Exit Current Bx Type Command Continuous DMA Transfer

Causes the SB ESP to finish the current block, then cease transferring. Use this command while the DMA is transferring the last block of audio data from/to the SB ESP. SB ESP-Reset() or any of new DMA command should reset this flag.

##### 4.2.1.13.9 Command DAh

Function: Exit Current Non-Bx Type Command Continuous DMA Transfer

Causes the SB ESP to finish the current block, then cease transferring. Use this command while the DMA is transferring the last block of audio data from/to the SB ESP. SB ESP-Reset() or any of new DMA command should reset this flag.

#### 4.2.1.14 SB ESP Version and Diagnostic Command Exh

##### 4.2.1.14.1 Command E0h

Function: SB ESP Read/Write Diagnostic Test

##### 4.2.1.14.2 Command E1h

Function: Get SB ESP Version Number

SB PRO Version 3.02

SB16 Version 4.02

Major Version = ASR5 (Default 04h)

Minor Version = ASR6 (Default 02h)

##### 4.2.1.14.3 Command E2h

Function: DMA testing

## 4.0 Functional Description (Continued)

E2h command is used to compute the subroutine's starting address for digital sound playback or recording according to the dedicated algorithm. The resultant byte should be sent back to system memory via legacy DMA operation method (two bytes transferred in all). After E2h is received, ESP will be set busy if E2Status (source from wave engine) is set active.

**4.2.1.14.4 Command E4h**

Function: Send Test Byte for Command E8h

**4.2.1.14.5 Command E8h**

Function: Get Test Byte Sent by Command E4h

**4.2.1.15 SB ESP Testing Command FXh**

**4.2.1.15.1 Command F2h**

Function: Generate an interrupt for test

SB ESP will generate an interrupt immediately after this command.

**4.2.1.15.2 Command F8h**

Function: SB ESP data read test

**4.2.1.16 Digitized Sound Data Format and Order**

**4.2.1.16.1 PCM Sound Data Format**

| Length | Format | Max. Value | | Min. Value | | Mid. Value |
|---|---|---|---|---|---|---|
| 8 Bit | Unsigned FFh | | 0h | | 80h | |
| 8 Bit | Signed | 7Fh | | 80h | | 0h |
| 16 Bit | Unsigned FFFFh | | 0h | | 8000h | |
| 16 Bit | Signed | 7FFFh | | 8000h | | 0h |

**4.2.1.16.2 PCM Sound Data Order**

**4.2.1.16.2.1 8-Bit Mono**

| | | | | |
|---|---|---|---|---|
| N | N+1 | N+2 | N+3 | N+4 |
| PCM 0 | PCM 1 | PCM 2 | PCM 3 | PCM 4 |

**4.2.1.16.2.2 8-Bit Stereo for 1xh and 9xh Type Command**

| | | | | |
|---|---|---|---|---|
| 2N | 2N+1 | 2(N+1)+0 | 2(N+1)+1 | 2(N+2)+0 |
| PCM 0.R | PCM 0.L | PCM 1.R | PCM 1.L | PCM 2.R |

**4.2.1.16.2.3 8-BIT STEREO FOR CXH TYPE COMMAND**

| | | | | |
|---|---|---|---|---|
| 2N | 2N+1 | 2(N+1)+0 | 2(N+1)+1 | 2(N+2)+0 |
| PCM 0.L | PCM 0.R | PCM 1.L | PCM 1.R | PCM 2.L |

**4.2.1.16.2.4 16-Bit Mono**

| | | | | |
|---|---|---|---|---|
| 2N | 2N+1 | 2(N+1)+0 | 2(N+1)+1 | 2(N+2)+0 |
| LOW | HIGH | LOW | HIGH | LOW |
| PCM 0 | PCM 0 | PCM 1 | PCM 1 | PCM 2 |

**4.2.1.16.2.5 16-Bit Stereo**

| | | | | |
|---|---|---|---|---|
| 4N | 4N+1 | 4N+2 | 4N+3 | 4(N+1)+0 |
| LOW | HIGH | LOW | HIGH | LOW |
| PCM 0.L | PCM 0.L | PCM 0.R | PCM 0.R | PCM 1.L |

**4.2.1.17 Legacy Sound Blaster Mixer Register**

The legacy SB mixer register are accessed through indirect addressing method, where the SBR4 stores the index of addressed register and SBR5 stores the register content to be read/write. In the following description, MXxx represents a mixer register whose index is xx.

## 4.0 Functional Description (Continued)

All of the legacy sound blaster mixer register except MX80, MX81 and MX82 are implemented virtually using the opl3 emulation RAM (512 bytes). Please refer to section 2.3 to get detailed description of the virtual implementation method.

The implemented MX80, MX81 are all read/write registers. Contents of MX80 and MX81 must be consistent with header 1 interrupt line configuration register. MX80 and MX81 can be accessed through AudioBase based port. MX82 is a part of the Audio interrupt status register.

#### 4.2.1.18 Implementation Note For Legacy Sound Blaster Mixer

#### 4.2.1.18.1

**To ease our design and maintain the game compatibility on mixer volume set, writing to the legacy sound blaster mixer register will set a dirty flag, and this flag will inform the software which one of the AC97 mixer should be updated and then update it using AudioBase based programming port.**

When any one of the SBPRO mixer registers (mixer index from 00h–2Fh) has been updated, a SBPRO volume update dirty flag will be set at Audio Status Register ASR0.

When any one of the SB16 mixer registers (mixer index from 30h–4Fh) has been updated, a SB16 volume update dirty flag will be set at Audio Status Register ASR0.

Any time when legacy sound blaster mixer is mixer-reset, a mixer soft-reset dirty flag will be set at Audio Status Register ASR0. All the SB16 / SBPRO mixer register (index from 02h to 4fh) are implemented using part of the opl3 emulation RAM(512bytes).

Any of the unused RAM cell (mixer index above 48h) can be used as the scratch registers for specific purpose and that's user defined.

The AC97 internal mixer registers are accessed via another AudioBase based port. (See 4. AC 97)

#### 4.2.1.19 SBPRO Mixer Register

**MX00 Mixer Reset**

Write only

Any write to this port will reset MX00-MX4f to default value. SB ESP_Reset() does not affect any of the mixer register. When this port is written, a SB mixer soft-reset dirty flag will be set in ASR0.

**MX02 Master Volume**

Default 808h

Write

Bit 7..4 reserved

Bit 3..0 16 level master volume for left and right channel

Read

Bit 7..4 master volume for left channel

Bit 3..0 master volume for right channel

**MX04 Digital Audio Volume**

Default 88h

Write

Bit 7..4 16 level left digital audio volume

Bit 3..0 16 level right digital audio volume

Read

Bit 7..4 left digital audio volume

Bit 3..0 right digital audio volume

**MX06 Music Volume**

Default 88h

Write

Bit 7..4 reserved

Bit 3..0 16 level music volume for left and right channel

Read

Bit 7..4 left music volume

Bit 3..0 right music volume

**MX08 CD-Audio Volume**

Default 00h

Write

Bit 7..4 reserved

Bit 3..0 16 level CD-Audio volume for left and right channel

Read

Bit 7..4 left CD-Audio volume

Bit 3..0 right CD-Audio volume

**MX0A Microphone Volume**

Default 00h

Write

Bit 7..3 reserved

Bit 2..0 8 level microphone volume

Read

Bit 7..3 reserved

Bit 2..0 microphone volume

**MX0C Digital Audio Input Control**

Default 00h

Dummy read/write register

Bit 7,6 reserved

Bit 5 input filter enable

Bit 4 reserved

Bit 3 input filter high/low

Bit 2,1 input source

Bit 0 reserved

Input filter enable and input filter high/low are dummy read/write bits for SB PRO compatibility.

Input filter enable: 0 - input low-pass filter on, 1 - off

Input filter high/low: 0 - low filter (3.2 kHz low pass), 1 - high filter (8.8 kHz low pass)

Input source

| Bit 2 | Bit 1 | |
|---|---|---|
| 0 | 0 | Microphone |
| 0 | 1 | CD-Audio |
| 1 | 0 | Microphone |
| 1 | 1 | External Line-In |

**MX0E Digital Audio Output Control**

Default 00h

Bit 7,6 reserved

## 4.0 Functional Description (Continued)

Bit 5 output filter enable
Bit 4..2 reserved
Bit 1 stereo switch
Bit 0 reserved

Output filter enable is dummy read/write bit for SB PRO compatibility.

Output filter enable: 0 - output low-pass filter on, 1 - off

Stereo switch: 1 - stereo output, 0 - mono output

Stereo switch is no use for SB ESP command Bxh and Cxh.

**MX22 Master Volume**

Default 88h

Write

Bit 7..4 16 level master left volume
Bit 3..0 16 level master right volume

Read

Bit 7..4 master left volume
Bit 3..0 master right volume

**MX24 Digital Audio Volume**

Default 88h

Write

Bit 7..4 16 level left digital audio volume
Bit 3..0 16 level right digital audio volume

Read

Bit 7..4 left digital audio volume
Bit 3..0 right digital audio volume

**MX26 Music Volume**

Default 88h

Write

Bit 7..4 16 level music left volume
Bit 3..0 16 level music right volume

Read

Bit 7..4 music left volume
Bit 3..0 music right volume

**MX28 CD-Audio Volume**

Default 00h

Write

Bit 7..4 16 level CD-Audio left volume
Bit 3..0 16 level CD-Audio right volume

Read

Bit 7..4 CD-Audio left volume
Bit 3..0 CD-Audio right volume

**MX2E External Line Volume**

Default 00h

Write

Bit 7..4 16 level external line-in left volume
Bit 3..0 16 level external line-in right volume

Read

Bit 7..4 external line-in left volume
Bit 3..0 external line-in right volume

**4.2.1.20 SB16 Mixer Register**

**MX30 Master Left Volume**

Default 90h

Read/Write

Bit 7..3 32 level master left volume
Bit 2..0 reserved
0 mute
31 0 dB (maximum volume)

**MX31 Master Right Volume**

Default 90h

Read/Write

Bit 7..3 32 level master right volume
Bit 2..0 reserved
0 mute
31 0 dB (maximum volume)

**MX32 Digital Audio Left Volume**

Default 90h

Read/Write

Bit 7..3 32 level digital audio left volume
Bit 2..0 reserved
0 mute
31 0 dB (maximum volume)

**MX33 Digital Audio Right Volume**

Default 90h

Read/Write

Bit 7..3 32 level digital audio right volume
Bit 2..0 reserved
0 mute
31 0 dB (maximum volume)

**MX34 Music Left Volume**

Default 90h

Read/Write

Bit 7..3 32 level music left volume
Bit 2..0 reserved
0 mute
31 0 dB (maximum volume)

**MX35 Music Right Volume**

Default 90h

Read/Write

Bit 7..3 32 level music right volume
Bit 2..0 reserved
0 mute
31 0 dB (maximum volume)

**MX36 CD-Audio Left Volume**

Default 00h

Read/Write

Bit 7..3 32 level CD-Audio left volume
Bit 2..0 reserved
0 mute
31 0 dB (maximum volume)

**MX37 CD-Audio Right Volume**

Default 00h

Read/Write

Bit 7..3 32 level CD-Audio right volume
Bit 2..0 reserved
0 mute
31 0 dB (maximum volume)

**MX38 External Line Left Volume**

Default 00h

## 4.0 Functional Description (Continued)

Read/Write
Bit 7..3 32 level external line left volume
Bit 2..0 reserved
0 mute
31 0 dB (maximum volume)

**MX39 External Line Right Volume**

Default 00h
Read/Write
Bit 7..3 32 level external line right volume
Bit 2..0 reserved
0 mute
31 0 dB (maximum volume)

**MX3A Microphone Volume**

Default 00h
Read/Write
Bit 7..3 32 level microphone volume
Bit 2..0 reserved
0 mute
31 0 dB (maximum volume)

**MX3B PC Speaker/Mono Input Volume**

Default 00h
Read/Write
Bit 7,6 4 level PC Speaker/Mono input volume
Bit 5..0 reserved
0 mute
3 0 dB (maximum volume)

**MX3C Output Mixer Control**

Default 1Fh
Read/Write
Bit 7..5 reserved
Bit 4 external line left enable
Bit 3 external line right enable
Bit 2 CD-Audio left enable
Bit 1 CD-Audio right enable
Bit 0 microphone enable
0 mute
1 enable audio output

**MX3D Input Mixer Left Control**

Default 15h
Read/Write
Bit 7 reserved
Bit 6 music left enable
Bit 5 dummy read/write bit
Bit 4 external line left enable
Bit 3 dummy read/write bit
Bit 2 CD-Audio left enable
Bit 1 dummy read/write bit
Bit 0 microphone enable
0 mute
1 enable audio input

**MX3E Input Mixer Right Control**

Default 0Bh
Read/Write
Bit 7 reserved
Bit 6 dummy read/write bit
Bit 5 music right enable
Bit 4 dummy read/write bit
Bit 3 external line right enable
Bit 2 dummy read/write bit
Bit 1 CD-Audio right enable
Bit 0 microphone enable
0 mute
1 enable audio input

**MX3F Input Left Mixer Gain**

Default 00h
Read/Write

| | | |
|---|---|---|
| Bit 7,6 | | input left mixer gain control |
| Bit 5..0 | | reserved |
| Bit 7 | Bit 6 | |
| 0 | 0 | gain = 1 |
| 0 | 1 | gain = 2 |
| 1 | 0 | gain = 4 |
| 1 | 1 | gain = 8 |

**MX40 Input Right Mixer Gain**

Default 00h
Read/Write
Bit 7,6 input right mixer gain control
Bit 5..0 reserved

| Bit 7 | Bit 6 | |
|---|---|---|
| 0 | 0 | gain = 1 |
| 0 | 1 | gain = 2 |
| 1 | 0 | gain = 4 |
| 1 | 1 | gain = 8 |

**MX41 Output Left Mixer Gain**

Default 00h
Read/Write
Bit 7,6 4 level output left mixer gain
Bit 5..0 reserved

| Bit 7 | Bit 6 | |
|---|---|---|
| 0 | 0 | gain = 1 |
| 0 | 1 | gain = 2 |
| 1 | 0 | gain = 4 |
| 1 | 1 | gain = 8 |

**MX42 Output Right Mixer Gain**

Default 00h
Read/Write
Bit 7,6 4 level output right mixer gain
Bit 5..0 reserved

| Bit 7 | Bit 6 | |
|---|---|---|
| 0 | 0 | gain = 1 |
| 0 | 1 | gain = 2 |
| 1 | 0 | gain = 4 |
| 1 | 1 | gain = 8 |

**MX43 Microphone Automatic Gain Control (AGC)**

Default 00h

Read/Write

| | | |
|---|---|---|
| Bit 7..1 | | reserved |
| Bit 0 | 0 | AGC disable |
| | 1 | AGC enable |

**MX44 Treble Left Control**

Default 80h

Read/Write

Bit 7..4 16 level treble left control

Bit 3..0 reserved

**MX45 Treble Right Control**

Default 80h

Read/Write

Bit 7..4 16 level treble right control

Bit 3..0 reserved

**MX46 Bass Left Control**

Default 80h

Read/Write

Bit 7..4 16 level bass left control

Bit 3..0 reserved

**MX47 Bass Right Control**

Default 80h

Read/Write

Bit 7..4 16 level bass right control

Bit 3..0 reserved

#### 4.2.1.21 SB16 Configuration/Status Register

**MX80 Sound Blaster Interrupt Setup Register**

Default 00h

Read/Write

| | | |
|---|---|---|
| Bit 7..6 | | reserved (read as 1) |
| Bit 5 | 1 | IRQ12 is used as legacy SB interrupt line |
| Bit 4 | 1 | IRQ11 is used as legacy SB interrupt line |
| Bit 3 | 1 | IRQ10 is used as legacy SB interrupt line |
| Bit 2 | 1 | IRQ7 is used as legacy SB interrupt line |
| Bit 1 | 1 | IRQ5 is used as legacy SB interrupt line |
| Bit 0 | 1 | IRQ9 is used as legacy SB interrupt line |

Only 1 bit of this register can be set 1 at any one time

Content of this register must be consistent with header 1 interrupt line configuration register.

**MX81 Sound Blaster DMA Channel Setup Register**

Default 00h

Read/Write

| | | |
|---|---|---|
| Bit 7..2 | | reserved (read as 0) |
| Bit 1 | 1 | legacy DMA channel 1 is used as legacy SB DMA channel |
| Bit 0 | 1 | legacy DMA channel 0 is used as legacy SB DMA channel |

Only 1 bit of this register can be set 1 at any one time

Content of this register must be consistent with CR1 on SB DMA channel setup.

**MX82 Sound Blaster Interrupt Status Register**

Default 00h

Read Only

| | | |
|---|---|---|
| Bit 7..3 | | reserved |
| Bit 2 | 1 | MPU-401 MIDI interrupt request |
| Bit 1 | 1 | Sound Blaster Bx type command DMA interrupt request |
| Bit 0 | 1 | Sound Blaster Non-Bx type command DMA interrupt request |

0: no interrupt

1: interrupt triggered

#### 4.2.1.22 Interface With Wave Engine

**4.2.1.22.1** Interfaced Register

All registers that physically implemented in wave engine are addressed using the 3-bit wide address line and 8-bit wide unidirectional data line initiated by Sound Blaster ESP Emulation Engine. These registers are considered to be the peripheral registers of implemented SB ESP engine.

**4.2.1.22.2** SB ESP DMA Control Register

**4.2.1.22.2.1** SBDMPD Direct Mode Playback Data Register

SB ESP Address: 6h

Size: 8 bits

Type: Write Only For SB ESP

Read Only For Wave Engine

Default: 80h

Bit 7..0 Direct Mode Playback Data

Any time after this register has ever been written, a DirectModePlay internal flag will be set by wave engine. This flag will be cleared when ESPRESET is active or LegacyDMARun is active. When this flag is active, the unsigned 8 bit PCM data will be output to AC-97 Codec.

**4.2.1.22.2.2** SBE2R DMA Testing Byte Register

SB ESP Address: 7h

Size: 8 bits

Type: Write Only For SB ESP

Read Only For Wave Engine

Default: 00h

Bit 7..0 testing byte that will be transferred to system location specified by DMAR0–3.

After this register has ever been written, E2Status (source from wave engine) will be set high. E2Status will be cleared after the testing byte has been sent to the system location.

**4.2.1.22.2.3** SBDMAC SB DMA Current Block Length Low Byte

SB ESP Address: 0h

Size: 8 bits

Type: Write Only For SB ESP

Read Only For Wave Engine

## 4.0 Functional Description (Continued)

Default: FFh

Bit 7..0 Low Byte of SB DMA Current Block Length Remained - 1

**4.2.1.22.2.4** SBDMAC SB DMA Current Block Length High Byte

SB ESP Address: 1h

Size: 8 bits

Type: Write Only For SB ESP

Read Only For Wave Engine

Default: 07h

Bit 7..0 High Byte of SB DMA Current Block Length Remained - 1

**4.2.1.22.2.5** SBDMAL SB DMA Base Block Length Low Byte

SB ESP Address: 2h

Size: 8 bits

Type: Write Only For SB ESP

Read Only For Wave Engine

Default: FFh

Bit 7..0 Low Byte of SB DMA Base Block Length - 1

**4.2.1.22.2.6** SBDMAL SB DMA Base Block Length High Byte

SB ESP Address: 3h

Size: 8 bits

Type: Write Only For SB ESP

Read Only For Wave Engine

Default: 07h

Bit 7..0 High Byte of SB DMA Base Block Length - 1

SBDMAL, SBDMAC are byte count when SBCTRL Bit 7 is 0.

SBDMAL, SBDMAC are word count when SBCTRL Bit 7 is 1.

When SBDMAC changed from 0 to FFFFh, a SBBLOVER interrupt signal should be generated and sent to SB ESP engine. Also the SBDMAC is reload from SBDMAL.

**4.2.1.22.2.7** SBCTRL SB DMA Running Mode and Data Format Register

SB ESP Address: 4h

Size: 8 bits

Type: Write Only For SB ESP

Read Only For Wave Engine

Default: 00h

Bit 7

0: 8 bit data format

1: 16 bit data format

Bit 6

0: mono

1: stereo

Bit 5

0: unsigned data format

1: signed data format

Bit 4

0: playback

1: recording

Bit 3

0: stop running after current block length expired

1: continue running after current block length expired

Bit 2..0 X legacy channel working mode

000 stop

001 run

010 silent audio

011 reserved

100 silent DMA

101 pause

110 reserved

111 direct mode play

### 4.3 SERIAL INTERFACE

#### 4.3.1 AC-97 Interface

#### 4.3.2 I2S Interface

#### 4.3.3 S/PDIF

### 4.4 POWER MANAGEMENT

Term:

| | |
|---|---|
| Header | standard PCI configuration space standard header type 0 |
| HIFW | host interface write |
| HRST | H/W Reset on PCI bus |
| SRST | S/W Reset |
| D0 | Audio Device Run: DC97, AC97 and PCI INTF at full power on state |
| D1 | Audio Device Pause: DC97 clock off |
| D2 | Audio Device Close: DC97 clock off, Inactivity Timer running, AC97 clock is about to be turned off |
| D3hot | Audio Device Clock Off: DC97 clock off, AC97 clock off, PCI clock keep running |
| D3cold | Audio Device Power Off: $V_{CC}$ is taken off. |

#### 4.4.1 Power Management for D0–D3 State

When at D0 state, DC97, AC97 and PCI INTF are running normally. After power on reset, audio device is at D0 state.

Once enter D1 state, if DC_PM_EN_ (PM_CFG bit[1]) = 0, clock of DC97 will be shut off and kept staying low after audio engine enters Pause or Stop state, i.e., Audio_clk will be shut off when Audio_idle flag is set.

Once enter D2 state, DC97 clock will be turned off as the same way as in D1 state. And also, the Inactivity Timer begins to count immediately.

When the counter expired (around 30 seconds later after enter D2 state), if Timer_PME_EN (PM_CFG bit[7]) = 1, a $\overline{\text{PME}}$ will be issued; otherwise, no $\overline{\text{PME}}$ issued but PM_ST will be set to 11b (D3) by chip.

Once enter D3 state, if AC_PM_EN_ (PM_CFG bit) = 0, the bit clock of AC97 will be shut off after a H/W power down command sequence through AC-link generated by DC97.

A one shot signal called ac97pm_pulse (generated by PM logic) is used to inform DC97 to generate those power down sequences.

#### 4.4.2 D0–D3 State Transition Table

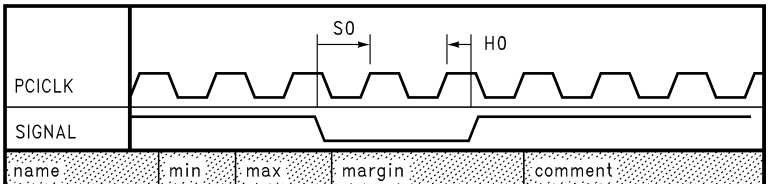
| Present State | Next State | State Change Way |
|---|---|---|
| D3cold | D0 | HRST |
| D3hot | D0 | SRST |

| Present State | Next State | State Change Way |
|---|---|---|
| D0 | D1 | HIFW |
| D2 or D1 | D0 | HIFW |

| Present State | Next State | State Change Way |
|---|---|---|
| D0 | D2 | HIFW |
| D2 | D3 | HIFW or Inactivity Timer expires |
| D0 | D3 | HIFW |

#### 5.1.2 Timing

The timing descriptions of host interface signals. All signal's transition edges are respected to the rising edge of PCICLK. The setup time and hold time requirements are illustrated in *Figure 4*.




| name | min | max | margin | comment |
|---|---|---|---|---|
| S0 | 3 | 27 | [16.741, 7.259] | Signal setup time |
| H0 | 1 | 27 | [9.372, 16.628] | Signal hold time |

DS100910-5

**FIGURE 4. Setup Time and Hold Time of All Signals**

##### 5.1.2.1 Slave Read/Write Timing

Slave read/write means that host performs I/O read/write operation to audio block while audio block acts as a bus slave. Data transition occurs at the rising edge of PCICLK while both chip select signal and $\overline{\text{DATARDY}}$ are active. They are two-clock wait states from chip select signal and $\overline{\text{DATARDY}}$.

Timing diagram are printed into file: sl_io_rd.ps and sl_io_wr.ps .

##### 5.1.2.2 Master Read/Write

There's one clock wait state of $\overline{\text{IRDY}}$ which means: 1) $\overline{\text{IRDY}}$ can be asserted in 2 clocks after $\overline{\text{FRAME}}$ is asserted; 2) $\overline{\text{IRDY}}$ has to be de-asserted for 1 clock after each of data transferring.

Timing diagram are printed into file: ma_mem_rd1.ps, ma_memrd2.ps, ma_mem_rd3.ps, ma_mem_wr2.ps and ma_mem_wr3.ps.

## Absolute Maximum Ratings (Note 1)

**If Military/Aerospace specified devices are required, please contact the National Semiconductor Sales Office/ Distributors for availability and specifications.**

| | |
|---|---|
| Supply Voltage ($V_{DD}$) | −0.5V to 4.0V |
| Input Voltage ($V_I$) | −0.5V to 5.5V +0.5V |
| Output Voltage ($V_O$) | −0.5V to $V_{DD}$ +0.5V |
| Storage Temperature ($T_{STG}$) | −65˚C to +165˚C |
| Power Dissipation ($P_D$) | 1W |
| Lead Temperature ($T_L$) (Soldering, 10 seconds) | +200˚C |

## 7.2 CAPACITANCE

$T_A$ = 0˚C to 70˚C, $V_{dd}$ = 3.3V ±5%, $V_{SS}$ = 0V (Note 2)

| Symbol | Parameter | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| $C_{IN}$ | Input Pin Capacitance | | | | 10 | pF |
| $C_{ICLK}$ | Clock Input Capacitance | | | | 12 | pF |
| $C_{IO}$ | I/O Pin Capacitance | f = 1 MHz | | | 10 | pF |
| $C_O$ | Output Pin Capacitance | f = 1 MHz | | | 10 | pF |

**Note 1:** "Absolute Maximum Ratings" are those values beyond which the safety of the device cannot be guaranteed. They are not meant to imply that the devices should be operated at these limits. The table of "Electrical Characteristics" specifies conditions of device operation.

**Note 2:** Capacitance limits are guaranteed by simulation but are not tested.

## 7.3 ELECTRICAL CHARACTERISTICS

$T_A$ = 0˚C to 70˚C, $V_{dd}$ = 3.3V ±5%, $V_{SS}$ = 0V

| Symbol | Parameter | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| $I_{CC}$ | $V_{DD}$ Average Supply Current | $V_{DD}$ = 3.3V, $V_{IL}$ = 0.5V, $V_{IH}$ = 2.4V, No Load | 20 | 35 | 50 | mA |
| $V_{IH}$ | Input High Voltage | | 1.65 | | 5.5 | V |
| $V_{IL}$ | Input Low Voltage | | −0.5 | | 1.0 | V |
| **CORE (3.3V Only)** | | | | | | |
| $CV_{cc}$ | Core Supply Voltage | | 3.2 | | 3.4 | V |
| **I/O - 3.3V Signalling Environment** | | | | | | |
| $V_{ih}$ | Input High Voltage | | 1.65 | | $V_{dd}$ | V |
| $V_{il}$ | Input Low Voltage | | | | 1.0 | V |
| $V_{oh}$ | Output High Voltage | | 2.97 | | | V |
| $V_{ol}$ | Output Low Voltage | | | | 0.33 | V |
| **Gameport I/O DC Characteristic (for pull up resistor values of 2.5k to 10kΩ)** | | | | | | |
| Vtranlh | For Schmitt trigger I/O, Input voltage at trigger point (output is $V_{cc/2}$ | When input goes from low to high | | 2.1 | | V |
| Vtranhl | For Schmitt trigger I/O, Input voltage at trigger point (output is $V_{cc/2}$ | When input goes from high to low | | 1.3 | | V |

## 7.4.1 Timing Table

$T_A$ = 0˚C to 70˚C, $V_{dd}$ = 3.3V ±5%, $V_{SS}$ = 0V

| Symbol | Parameter | Min | Max | Units |
|---|---|---|---|---|
| **PCI Clock** | | | | |
| Tcyc | CLK Cycle Time | 30 | | ns |
| Thigh | CLK High Time | 14 | 16 | ns |
| Tlow | CLK Low Time | 14 | 16 | ns |
| Tskew | CLK Skew | −2 | 2 | ns |
| **AC'97 Clock** | | | | |
| Tcyc | ACCLK Cycle Time | 72 | | ns |
| Thigh | ACCLK High Time | 36 | 45 | ns |
| Tlow | ACCLK Low Time | 36 | 45 | ns |

## 7.4.1 Timing Table (Continued)

$T_A$ = 0˚C to 70˚C, $V_{dd}$ = 3.3V ±5%, $V_{SS}$ = 0V

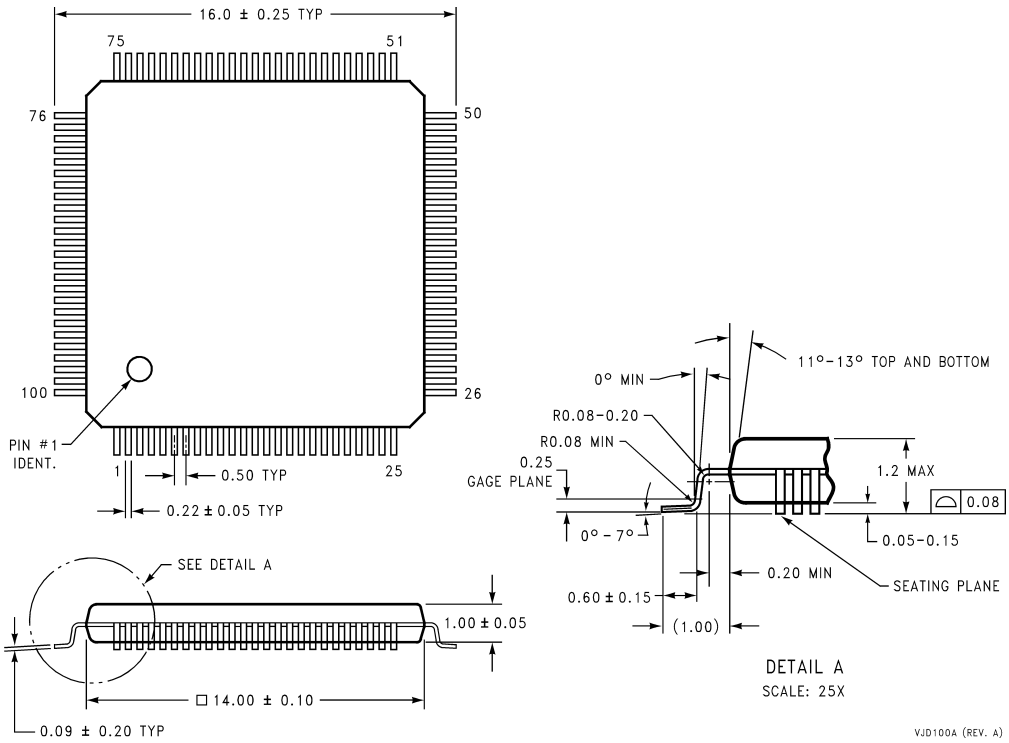
| Symbol | Parameter | Min | Max | Units |
|---|---|---|---|---|
| **PCI Reset** | | | | |
| Trst_low | RST# low time after power stable | 1 | | ms |
| Trst_clk | RST# low time after PCICLK stable | 100 | | µs |
| **AC'97 Reset (Cold and Warm)** | | | | |
| Trst_low | ACRST_low time | 1 | | µs |
| Trst2clk | ACRST_ inactive to ACCCLK starts | 162.8 | | ns |
| Tsync_high | ACSYNC high time | 1 | | µs |
| Tsync2clk | ACSYNC_ inactive to ACCCLK starts | 162.8 | | ns |

## 7.4.2 PCI Signals

PCI Inputs: AD31:0, CBE#3:0, FRAME#, IRDY#, TRDY#, STOP#, DEVSEL#, IDSEL#, PAR, GNT#
PCI Outputs: AD31:0, CBE#3:0, FRAME#, IRDY#, TRDY#, STOP#, DEVSEL#, IDSEL#, PAR, REQ#, INTA#
Point-to-Point Signals: REQ#, GNT#

| Symbol | Parameter | Condition | Min | Max | Units |
|---|---|---|---|---|---|
| Tval_bus | CLK to PCI output Valid Delay_ bussed signals | | 2 | 11 | ns |
| Tval_ptp | CLK to PCI output Valid Delay_ point to point signals | | 2 | 12 | ns |
| Ton | PCI Output float to active | | 2 | | ns |
| Toff | PCI Output active to float | | | 28 | ns |
| AD0:31 bus | PCI Input set up time to CLK_ bussed signals | | 9 | | ns |
| Tsu GNT | PCI Input set up time to CLK_GNT# | | 12 | | ns |
| Tsu REQ | PCI Input set up time to CLK_REQ# | | 12 | | ns |
| Th | PCI Input hold time from CLK | | 0 | | ns |
| **AC'97 Signals** | | | | | |
| Tsetup | Setup from edge of ACCLK | | 10 | | ns |
| Thold | Hold from edge of ACCLK | | 10 | | ns |

## Physical Dimensions inches (millimeters) unless otherwise noted




**100-Pin Thin Plastic Quad Flatpak**
**Order Number LM4560VJD**
**NS Package Number VJD100A**

### LIFE SUPPORT POLICY

NATIONAL'S PRODUCTS ARE NOT AUTHORIZED FOR USE AS CRITICAL COMPONENTS IN LIFE SUPPORT DEVICES OR SYSTEMS WITHOUT THE EXPRESS WRITTEN APPROVAL OF THE PRESIDENT OF NATIONAL SEMICONDUCTOR CORPORATION. As used herein:

1. Life support devices or systems are devices or systems which, (a) are intended for surgical implant into the body, or (b) support or sustain life, and whose failure to perform when properly used in accordance with instructions for use provided in the labeling, can be reasonably expected to result in a significant injury to the user.
2. A critical component in any component of a life support device or system whose failure to perform can be reasonably expected to cause the failure of the life support device or system, or to affect its safety or effectiveness.

**National Semiconductor Corporation**
Americas
Tel: 1-800-272-9959
Fax: 1-800-737-7018
Email: support@nsc.com

www.national.com

**National Semiconductor Europe**
Fax: +49 (0) 1 80-530 85 86
Email: europe.support@nsc.com
Deutsch Tel: +49 (0) 1 80-530 85 85
English Tel: +49 (0) 1 80-532 78 32
Français Tel: +49 (0) 1 80-532 93 58
Italiano Tel: +49 (0) 1 80-534 16 80

**National Semiconductor Asia Pacific Customer Response Group**
Tel: 65-2544466
Fax: 65-2504466
Email: sea.support@nsc.com

**National Semiconductor Japan Ltd.**
Tel: 81-3-5639-7560
Fax: 81-3-5639-7507

## IMPORTANT NOTICE

Texas Instruments Incorporated and its subsidiaries (TI) reserve the right to make corrections, modifications, enhancements, improvements, and other changes to its products and services at any time and to discontinue any product or service without notice. Customers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All products are sold subject to TI's terms and conditions of sale supplied at the time of order acknowledgment.

TI warrants performance of its hardware products to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are used to the extent TI deems necessary to support this warranty. Except where mandated by government requirements, testing of all parameters of each product is not necessarily performed.

TI assumes no liability for applications assistance or customer product design. Customers are responsible for their products and applications using TI components. To minimize the risks associated with customer products and applications, customers should provide adequate design and operating safeguards.

TI does not warrant or represent that any license, either express or implied, is granted under any TI patent right, copyright, mask work right, or other TI intellectual property right relating to any combination, machine, or process in which TI products or services are used. Information published by TI regarding third-party products or services does not constitute a license from TI to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property of the third party, or a license from TI under the patents or other intellectual property of TI.

Reproduction of TI information in TI data books or data sheets is permissible only if reproduction is without alteration and is accompanied by all associated warranties, conditions, limitations, and notices. Reproduction of this information with alteration is an unfair and deceptive business practice. TI is not responsible or liable for such altered documentation. Information of third parties may be subject to additional restrictions.

Resale of TI products or services with statements different from or beyond the parameters stated by TI for that product or service voids all express and any implied warranties for the associated TI product or service and is an unfair and deceptive business practice. TI is not responsible or liable for any such statements.

TI products are not authorized for use in safety-critical applications (such as life support) where a failure of the TI product would reasonably be expected to cause severe personal injury or death, unless officers of the parties have executed an agreement specifically governing such use. Buyers represent that they have all necessary expertise in the safety and regulatory ramifications of their applications, and acknowledge and agree that they are solely responsible for all legal, regulatory and safety-related requirements concerning their products and any use of TI products in such safety-critical applications, notwithstanding any applications-related information or support that may be provided by TI. Further, Buyers must fully indemnify TI and its representatives against any damages arising out of the use of TI products in such safety-critical applications.

TI products are neither designed nor intended for use in military/aerospace applications or environments unless the TI products are specifically designated by TI as military-grade or "enhanced plastic." Only products designated by TI as military-grade meet military specifications. Buyers acknowledge and agree that any such use of TI products which TI has not designated as military-grade is solely at the Buyer's risk, and that they are solely responsible for compliance with all legal and regulatory requirements in connection with such use.

TI products are neither designed nor intended for use in automotive applications or environments unless the specific TI products are designated by TI as compliant with ISO/TS 16949 requirements. Buyers acknowledge and agree that, if they use any non-designated products in automotive applications, TI will not be responsible for any failure to meet such requirements.

Following are URLs where you can obtain information on other Texas Instruments products and application solutions:

| **Products** | | **Applications** | |
|---|---|---|---|
| Audio | www.ti.com/audio | Communications and Telecom | www.ti.com/communications |
| Amplifiers | amplifier.ti.com | Computers and Peripherals | www.ti.com/computers |
| Data Converters | dataconverter.ti.com | Consumer Electronics | www.ti.com/consumer-apps |
| DLP® Products | www.dlp.com | Energy and Lighting | www.ti.com/energy |
| DSP | dsp.ti.com | Industrial | www.ti.com/industrial |
| Clocks and Timers | www.ti.com/clocks | Medical | www.ti.com/medical |
| Interface | interface.ti.com | Security | www.ti.com/security |
| Logic | logic.ti.com | Space, Avionics and Defense | www.ti.com/space-avionics-defense |
| Power Mgmt | power.ti.com | Transportation and Automotive | www.ti.com/automotive |
| Microcontrollers | microcontroller.ti.com | Video and Imaging | www.ti.com/video |
| RFID | www.ti-rfid.com | | |
| OMAP Mobile Processors | www.ti.com/omap | | |
| Wireless Connectivity | www.ti.com/wirelessconnectivity | | |
| | **TI E2E Community Home Page** | | e2e.ti.com |

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265