ALS4000 Media Audio Controller SPEC

Date: May,28th,1998

Features :

- High performance PCI Digital Audio Subsystem Controller.
- Sound Blaster/Pro/16 Emulation
- MIDI port with input 32 bytes and output 16 bytes FIFO
- Game port interface with fully compatibility with Microsoft SideWinder
- Full-duplex DMA operation
- PCI power management interface
- Hardware power down mode for notebook application.
- Build-In FM compatible synthesizer.

Applications :

- Windows and MPC level 2 compatible audio subsystem.
- PC games.
- Computer based audio reproduction
- Audio on-line tutorial
- Voice annotation or voice E-mail interface.
- Voice recognition or voice command controller.
- Text to speech.
- Karaoke/music sound box.
- MIDI controller.

ALS4000 Pin Descriptions:

100 pin QFP package

PCI Bus Interface Signal : 49 pins

Pin Name	Type	Pin No.	Description	Characteristic Definition
AD0	I/O	23	PCI Address/Data bit 0	6mA TTL compatible CMOS IO (Vt=1.7V)
AD1	I/O	22	PCI Address/Data bit 1	6mA TTL compatible CMOS IO (Vt=1.7V)
AD2	I/O	21	PCI Address/Data bit 2	6mA TTL compatible CMOS IO (Vt=1.7V)
AD3	I/O	20	PCI Address/Data bit 3	6mA TTL compatible CMOS IO (Vt=1.7V)
AD4	I/O	19	PCI Address/Data bit 4	6mA TTL compatible CMOS IO (Vt=1.7V)
AD5	I/O	17	PCI Address/Data bit 5	6mA TTL compatible CMOS IO (Vt=1.7V)
AD6	I/O	16	PCI Address/Data bit 6	6mA TTL compatible CMOS IO (Vt=1.7V)
AD7	I/O	15	PCI Address/Data bit 7	6mA TTL compatible CMOS IO (Vt=1.7V)
AD8	I/O	13	PCI Address/Data bit 8	6mA TTL compatible CMOS IO (Vt=1.7V)
AD9	I/O	11	PCI Address/Data bit 9	6mA TTL compatible CMOS IO (Vt=1.7V)
AD10	I/O	10	PCI Address/Data bit 10	6mA TTL compatible CMOS IO (Vt=1.7V)
AD11	I/O	9	PCI Address/Data bit 11	6mA TTL compatible CMOS IO (Vt=1.7V)
AD12	I/O	8	PCI Address/Data bit 12	6mA TTL compatible CMOS IO (Vt=1.7V)
AD13	I/O	6	PCI Address/Data bit 13	6mA TTL compatible CMOS IO (Vt=1.7V)
AD14	I/O	5	PCI Address/Data bit 14	6mA TTL compatible CMOS IO (Vt=1.7V)
AD15	I/O	4	PCI Address/Data bit 15	6mA TTL compatible CMOS IO (Vt=1.7V)
AD16	I/O	92	PCI Address/Data bit 16	6mA TTL compatible CMOS IO (Vt=1.7V)
AD17	I/O	91	PCI Address/Data bit 17	6mA TTL compatible CMOS IO (Vt=1.7V)
AD18	I/O	90	PCI Address/Data bit 18	6mA TTL compatible CMOS IO (Vt=1.7V)
AD19	I/O	89	PCI Address/Data bit 19	6mA TTL compatible CMOS IO (Vt=1.7V)
AD20	I/O	88	PCI Address/Data bit 20	6mA TTL compatible CMOS IO (Vt=1.7V)
AD21	I/O	86	PCI Address/Data bit 21	6mA TTL compatible CMOS IO (Vt=1.7V)
AD22	I/O	85	PCI Address/Data bit 22	6mA TTL compatible CMOS IO (Vt=1.7V)
AD23	I/O	84	PCI Address/Data bit 23	6mA TTL compatible CMOS IO (Vt=1.7V)
AD24	I/O	80	PCI Address/Data bit 24	6mA TTL compatible CMOS IO (Vt=1.7V)
AD25	I/O	79	PCI Address/Data bit 25	6mA TTL compatible CMOS IO (Vt=1.7V)
AD26	I/O	78	PCI Address/Data bit 26	6mA TTL compatible CMOS IO (Vt=1.7V)
AD27	I/O	77	PCI Address/Data bit 27	6mA TTL compatible CMOS IO (Vt=1.7V)
AD28	I/O	76	PCI Address/Data bit 28	6mA TTL compatible CMOS IO (Vt=1.7V)
AD29	I/O	74	PCI Address/Data bit 29	6mA TTL compatible CMOS IO (Vt=1.7V)

AD30	I/O	73	PCI Address/Data bit 30	6mA TTL compatible CMOS IO (Vt=1.7V)
AD31	I/O	72	PCI Address/Data bit 31	6mA TTL compatible CMOS IO (Vt=1.7V)
C/BE0	I/O	14	Command/Byte enable bit 0	6mA TTL compatible CMOS IO (Vt=1.7V)
C/BE1	I/O	3	Command/Byte enable bit 1	6mA TTL compatible CMOS IO (Vt=1.7V)
C/BE2	I/O	93	Command/Byte enable bit 2	6mA TTL compatible CMOS IO (Vt=1.7V)
C/BE3	I/O	82	Command/Byte enable bit 3	6mA TTL compatible CMOS IO (Vt=1.7V)
FRAME#	I/O	95	PCI Cycle Frame	8mA TTL compatible CMOS IO (Vt=1.7V)
IRDY#	I/O	96	PCI Initiator Ready	8mA TTL compatible CMOS IO (Vt=1.7V)
TRDY#	I/O	98	PCI Target Ready	8mA TTL compatible CMOS IO (Vt=1.7V)
STOP#	I/O	100	PCI Stop	8mA TTL compatible CMOS IO (Vt=1.7V)
IDSEL	I	83	PCI Initialization Device Select	TTL compatible CMOS input (Vt=1.7V)
DEVSEL#	I/O	99	PCI Device Select	8mA TTL compatible CMOS IO (Vt=1.7V)
CLK	I	67	PCI System clock(33MHz)	TTL compatible CMOS input
RST#	I	65	PCI System Reset	Schmitt triggered CMOS input (1.4V-2.2V)
PAR	O	2	PCI Parity	6mA TTL compatible CMOS output
REQ#	O	70	PCI Request	6mA TTL compatible CMOS output
GNT#	I	69	PCI Grant	TTL compatible CMOS input (Vt=1.7V)
INTA#	O	64	PCI Interrupt Request A	6mA open-drain CMOS output
PME#	O	71	Power management event	6mA open-drain CMOS output

Analog Signal: 21 pins

Pin Name	Type	Pin No.	Description	Characteristic Definition
MIC	I	39	microphone input	analog input (3.0V dynamic rang, 0.5V-3.5V)
LINE-L	I	33	external line input left	analog input (3.0V dynamic rang, 0.5V-3.5V)
LINE-R	I	46	external line input right	analog input (3.0V dynamic rang, 0.5V-3.5V)
CD-L	I	32	external CD audio input left	analog input (3.0V dynamic rang, 0.5V-3.5V)
CD-R	I	47	external CD audio input right	analog input (3.0V dynamic rang, 0.5V-3.5V)
MUSIC-L	I	35	external music input left	analog input (3.0V dynamic rang, 0.5V-3.5V)
MUSIC-R	I	44	external music input right	analog input (3.0V dynamic rang, 0.5V-3.5V)
MONO-I	I	51	PC speaker input	analog input (3.0V dynamic rang, 0.5V-3.5V)
MONO-O	O	40	PC speaker output	analog output (2mA maximum output current)
ALEFT	O	34	audio mixer output left	analog output (2mA maximum output current)
ARIGHT	O	45	audio mixer output right	analog output (2mA maximum output current)
VREF1	O	41	2.0V voltage output 1	analog reference voltage output
VREF2	O	38	2.0V voltage output 2	analog reference voltage output
ADL-O	O	30	A/D ANTI-ALIAS filter left	analog filter output pad
ADR-O	O	49	A/D ANTI-ALIAS filter right	analog filter output pad
ADL-I	I	29	left A/D filter loop back	analog input (3.0V dynamic rang, 0.5V-3.5V)
ADR-I	I	50	right A/D filter loop back	analog input (3.0V dynamic rang, 0.5V-3.5V)
DACL	I/O	31	SB D/A low-pass smooth filter left	analog filter IO pad
DACR	I/O	48	SB D/A low-pass smooth filter right	analog filter IO pad
SYNCL-F	O	36	FM D/A low-pass smooth filter left	analog filter output pad
SYNCR-F	O	43	FM D/A low-pass smooth filter right	analog filter output pad

Game Port/MIDI Interface Signal : 10 pins

Pin Name	Type	Pin No.	Description	Characteristic Definition
MIDIIN	I	63	MIDI serial input	Schmitt triggered CMOS input with 10KΩ pull up (1.7V-2.7V)
MIDIOUT	O	62	MIDI serial output	8mA CMOS output
GD0	I/O	61	Game Port A timer X	8mA Schmitt triggered CMOS IO (2.0V-3.0V)
GD1	I/O	60	Game Port A timer Y	8mA Schmitt triggered CMOS IO (2.0V-3.0V)
GD2	I/O	59	Game Port B timer X	8mA Schmitt triggered CMOS IO (2.0V-3.0V)
GD3	I/O	57	Game Port B timer Y	8mA Schmitt triggered CMOS IO (2.0V-3.0V)
GD4	I	56	Game Port A button A	Schmitt triggered CMOS input with 10KΩ pull up (2.0V-3.0V)
GD5	I	55	Game Port A button B	Schmitt triggered CMOS input with 10KΩ pull up (2.0V-3.0V)
GD6	I	54	Game Port B button A	Schmitt triggered CMOS input with 10KΩ pull up (2.0V-3.0V)
GD7	I	53	Game Port B button B	Schmitt triggered CMOS input with 10KΩ pull up (2.0V-3.0V)

Miscellaneous Signal : 3 pins

Pin Name	Type	Pin No.	Description	Characteristic Definition
XTALI	I	25	Crystal or oscillator input (14.318M)	crystal / oscillator input pad
XTALO	O	26	Crystal output	crystal output pad
PWRDN_	I	27	H/W power down	Schmitt triggered CMOS input with 50KΩ pull up (1.4V-2.2V)

Note: if PWRDN_ pin is activated (low), ALS4000 is in the minimal power consumption and will not respond to any PCI transactions even PCI configuration access.

Power/Ground : 17 pins

Pin Name	Type	Pin No.	Description	Characteristic Definition
VDD1	I	7	Digital power 5V	
VDD2	I	18	Digital power 5V	
VDD3	I	66	Digital power 5V	
VDD4	I	75	Digital power 5V	
VDD5	I	87	Digital power 5V	
VDD6	I	97	Digital power 5V	
AVDD1	I	28	Analog power 5V	
AVDD2	I	42	Analog power 5V	
GND1	I	1	Digital Ground	
GND2	I	12	Digital Ground	
GND3	I	24	Digital Ground	
GND4	I	58	Digital Ground	
GND5	I	68	Digital Ground	
GND6	I	81	Digital Ground	
GND7	I	94	Digital Ground	
AGND1	I	37	Analog Ground	
AGND2	I	52	Analog Ground	

Part I : Specification for SB Core Logic

I/O Register Address:

ESP (enhanced sound processor):

BASE+6h	W	ESP-RESET-PORT
BASE+6h	R	CR1E-ACK-PORT
BASE+Ah	R	ESP-READ-DATA
BASE+Ch	W	ESP-COMMAND/DATA
BASE+Ch	R	ESP-WR-STATUS
BASE+Eh	R	ESP-RD-STATUS8
BASE+Fh	R	ESP-RD-STATUS16

BASE.9~4 = GCRA9.9~4;

ESP-RD-STATUS = ESP-RD-STATUS8 or ESP-RD-STATUS16

Mixer & Control:

BASE+4h	R/W	MIXER-INDEX
BASE+5h	R/W	MIXER-DATA

FM (OPL3 compatible) synthesizer :

BASE+0-3h	R/W	OPL3/4 address:0-3
BASE+8,9h	R/W	OPL3/4 address:0,1
ADLIBBASE+0-3h	R/W	OPL3/4 address:0-3

ADLIBBASE.9~3 = GCRA8.9~3;

Gameport :

GAMEBASE+0-1h	R	GAME-READ
GAMEBASE+0-1h	W	GAME-WRITE

GAMEBASE.9~3 = GCRA8.25~19;

MPU401:

MPU401BASE+0h	R	MIDI INPUT FIFO
MPU401BASE+0h	W	MIDI OUTPUT FIFO
MPU401BASE+1h	R	MIDI-STATUS
MPU401BASE+1h	W	MIDI-COMMAND

MPU401BASE.9~3 = GCRA9.25~19;

Game Port Definition:

ALS4000 usually drive GD0~3 to low. When Game-port write command is active, GD0~3 is tri-state until level on GD0~3 is high again. ALS4000 implement a FLIP-FLOP for GD0~3 respectively, called FF0~3. FF0~3 are default 1 after system reset. Any Game write operation will set FF0~3 . When level on GD0~3 reach logic high level, the corresponding FLIP-FLOP will be clear automatically. This implementation work fine with analog joystick and digital game pad with any detection procedure. The following are mapping between SD0~7 and GD0~7:

SD7	GD7
SD6	GD6
SD5	GD5
SD4	GD4
SD3	FF3(set/reset controlled by GD3)
SD2	FF2(set/reset controlled by GD2)
SD1	FF1(set/reset controlled by GD1)
SD0	FF0(set/reset controlled by GD0)

GAMEPORT = GAMEBASE+1h

FM synthesizer :

Refer to “ALS120 FM synthesizer specification“

ESP Register Definition:

ESP-RESET-PORT:
Write only

Bit 0	0	normal
	1	reset ESP
Bit 7..1	X	reserved

ESP_RESET should do the following things:

a. Reset ESP to no operation status and clear ESP busy flag.
b. Flush primary output FIFO.
c. Reset any flag that may affect command execution.
d. Reset data latched by SB D/A to middle range.
e. Reset sample frequency to 44.1KHz.
f. Reset DMA block length to 0x7ff.

ESP-RD-STATUS
Read only

Bit 7	0	no data on ESP-READ-DATA
	1	data available at ESP-READ-DATA port
Bit 6..0	X	reserved (same as ESP-READ-DATA bit 6..0)

Read ESP-RD-STATUS8 will clear interrupt generated by the ESP for **non-BX type DMA**.
Read ESP-RD-STATUS16 will clear interrupt for **BX type DMA**.
After CPU read the data from ESP-READ-DATA port, bit 7 of this read status port will reset to 0 (no data) until the next read data is available and bit 7 set to 1 (data available).

ESP-READ-DATA
Read only

Bit 7..0	X	the data return by ESP

ESP-COMMAND/DATA
Write only

Bit 7..0	X	the command or data to ESP

ESP-WR-STATUS:
Read only

Bit 7	0	ESP is available for next command/data
	1	ESP is busy
Bit 6..0	X	reserved (same as ESP-READ-DATA bit 6..0)

After CPU write the command/data to the ESP-COMMAND/DATA port, bit 7 of this write status will set to 1 (busy) until the ESP processed the written command/data and waiting for the next command/data by reset bit 7 to 0 (not busy). Any acknowledge byte must be readback before any new command is issued. ESP will be set busy if any DMA operation is started and will be set not busy if command port is read twice.

Interrupt Acknowledge Procedure:

1. IO read(ESP-RD-STATUS8), clear non BX type DMA IRQ
2. IO read(ESP-RD-STATUS16), clear BX type DMA IRQ .
3. IO read(MIDI-DATA), clear MPU401 MIDI interrupt
4. IO read(CR1E-ACK-PORT),clear CR1E type DMA IRQ

MPU-401 MIDI Register Definition:

MIDI-STATUS
Read only

Bit 7	0	MIDI input data is available
	1	no MIDI input
Bit 6	0	ready for MIDI data output or new MIDI command
	1	MIDIOUT FIFO full or MIDIOUT FIFO not empty when MIDI_RESET
Bit 5-0		reserved

MIDI-COMMAND
Write only

Bit 7..0	command to MIDI controller

In **pass-thru** mode,

ENTER_UART 03Fh
Return ack byte (0FEh) in **midi-data**, generate an interrupt if switch to UART mode successfully.
Reading data port will clear the interrupt signal.

MIDI_RESET 0FFh
Return ack byte (0FEh) in **midi-data**, generate an interrupt, stay in **pass-thru** mode.

In UART mode,

MIDI_RESET 0FFh
Flush MIDIIN FIFO, wait until MIDIOUT FIFO/MIDI RAM empty, go to **pass-thru** mode .

MIDI-DATA
Read/write

Read
Bit 7..0 MIDI data input in UART mode or acknowledge byte

Write
Bit 7..0 MIDI data output in UART mode

ALS4000 Mixer Register Definition:

MIXER-INDEX
Read/write

Bit 7 0 mixer
1 control

Bit 6..0 index

MIXER-DATA
Read/write

Bit 7..0 mixer or control data register by **mixer-index**

MX00-MX7F mixer data register 00-7F base on **mixer-index** value
CR00-CR3F control data register 00-3F base on **mixer-index** value with bit 7 and 6 = 1.
SBCONFIGn = MX80-MXBF

Sound Blaster Pro:

MX00 mixer reset
Write only

Any write to this port will reset MX00-MX7F to default value. ESP_RESET() does not affect any of the mixer register.

MX02 master volume
Ghost register

Write
Bit 7 X
Bit 6 X
Bit 5 X
Bit 4 X
Bit 3 MX30.7 & MX31.7
Bit 2 MX30.6 & MX31.6
Bit 1 MX30.5 & MX31.5
Bit 0 MX30.4 & MX31.4

Bit 3 MX30.3

Bit 3	MX31.3

Read

Bit 7	MX30.7
Bit 6	MX30.6
Bit 5	MX30.5
Bit 4	MX30.4
Bit 3	MX31.7
Bit 2	MX31.6
Bit 1	MX31.5
Bit 0	MX31.4

Bit 7..4	reserved
Bit 3..0	0 to 15, master volume left and right in 3 dB step

0	-45 dB (off)
15	0 dB (maximum volume)

MX04 digital audio left/right volume
Ghost register

Write

Bit 7	MX32.7
Bit 6	MX32.6
Bit 5	MX32.5
Bit 4	MX32.4
Bit 3	MX33.7
Bit 2	MX33.6
Bit 1	MX33.5
Bit 0	MX33.4

Bit 7	MX32.3
Bit 3	MX33.3

Read

Bit 7	MX32.7
Bit 6	MX32.6
Bit 5	MX32.5
Bit 4	MX32.4
Bit 3	MX33.7
Bit 2	MX33.6
Bit 1	MX33.5
Bit 0	MX33.4

Bit 7..4	0 to 15, digital audio left volume in 3 dB step
Bit 3..0	0 to 15, digital audio right volume in 3 dB step

0	-45 dB (off)
15	0 dB (maximum volume)

MX06 music volume
Ghost register

Write

Bit 7	X
Bit 6	X
Bit 5	X
Bit 4	X
Bit 3	MX34.7 & MX35.7
Bit 2	MX34.6 & MX35.6

Bit 1	MX34.5 & MX35.5
Bit 0	MX34.4 & MX35.4
Bit 3	MX34.3
Bit 3	MX35.3
Read	
Bit 7	MX34.7
Bit 6	MX34.6
Bit 5	MX34.5
Bit 4	MX34.4
Bit 3	MX35.7
Bit 2	MX35.6
Bit 1	MX35.5
Bit 0	MX35.4
Bit 7..4	reserved
Bit 3..0	0 to 15, music volume left and right in 3 dB step
0	-45 dB (off)
15	0 dB (maximum volume)

MX08 CD-audio volume
Ghost register

Write	
Bit 7	X
Bit 6	X
Bit 5	X
Bit 4	X
Bit 3	MX36.7 & MX37.7
Bit 2	MX36.6 & MX37.6
Bit 1	MX36.5 & MX37.5
Bit 0	MX36.4 & MX37.4
Bit 3	MX36.3
Bit 3	MX37.3
Read	
Bit 7	MX36.7
Bit 6	MX36.6
Bit 5	MX36.5
Bit 4	MX36.4
Bit 3	MX37.7
Bit 2	MX37.6
Bit 1	MX37.5
Bit 0	MX37.4
Bit 7..4	reserved
Bit 3..0	0 to 15, CD-audio volume left and right in 3 dB step
0	-45 dB (off)
15	0 dB (maximum volume)

MX0A microphone volume
Ghost register

Write	
Bit 7..3	X
Bit 2	MX3A.7

Bit 1	MX3A.6
Bit 0	MX3A.5

Bit 2	MX3A.4
Bit 1	MX3A.3

Read

Bit 7..3	X
Bit 2	MX3A.7
Bit 1	MX3A.6
Bit 0	MX3A.5

Bit 7..3	reserved
Bit 2..0	0 to 7, microphone volume in 6 dB step

0	-42 dB (off)
7	0 dB (maximum 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 Sound Blaster 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 (MX3D & MX3E = 1)
0	1	CD-audio (MX3D = 04h, MX3E = 02h)
1	0	microphone (MX3D & MX3E = 1)
1	1	external line-in (MX3D = 10h, MX3E = 08h)

Write to input source will update MX3D and MX3E input mixer left/right control register and 2 dummy bits. Read from input source register will return 2 dummy bits.

MX0E digital audio output control
Default 00h

Bit 7,6	reserved
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 Sound Blaster Pro compatibility.

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

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

Stereo switch is not used in ESP command BXh and CXh.

MX22 master left/right volume
Ghost register

Write
Bit 7 MX30.7
Bit 6 MX30.6
Bit 5 MX30.5
Bit 4 MX30.4
Bit 3 MX31.7
Bit 2 MX31.6
Bit 1 MX31.5
Bit 0 MX31.4

Bit 7 MX30.3
Bit 3 MX31.3

Read
Bit 7 MX30.7
Bit 6 MX30.6
Bit 5 MX30.5
Bit 4 MX30.4
Bit 3 MX31.7
Bit 2 MX31.6
Bit 1 MX31.5
Bit 0 MX31.4

Bit 7..4 0 to 15, master left volume in 3 dB step
Bit 3..0 0 to 15, master right volume in 3 dB step

0 -45 dB (off)
15 0 dB (maximum volume)

MX24 digital audio left/right volume
Same as MX04

MX26 music left/right volume
Ghost register

Write
Bit 7 MX34.7
Bit 6 MX34.6
Bit 5 MX34.5
Bit 4 MX34.4
Bit 3 MX35.7
Bit 2 MX35.6
Bit 1 MX35.5
Bit 0 MX35.4

Bit 7 MX34.3
Bit 3 MX35.3

Read
Bit 7 MX34.7
Bit 6 MX34.6
Bit 5 MX34.5
Bit 4 MX34.4
Bit 3 MX35.7

Bit 2	MX35.6
Bit 1	MX35.5
Bit 0	MX35.4

Bit 7..4	0 to 15, music left volume in 3 dB step
Bit 3..0	0 to 15, music right volume in 3 dB step

0	-45 dB (off)
15	0 dB (maximum volume)

MX28 CD-audio left/right volume
Ghost register

Write

Bit 7	MX36.7
Bit 6	MX36.6
Bit 5	MX36.5
Bit 4	MX36.4
Bit 3	MX37.7
Bit 2	MX37.6
Bit 1	MX37.5
Bit 0	MX37.4

Bit 7	MX36.3
Bit 3	MX37.3

Read

Bit 7	MX36.7
Bit 6	MX36.6
Bit 5	MX36.5
Bit 4	MX36.4
Bit 3	MX37.7
Bit 2	MX37.6
Bit 1	MX37.5
Bit 0	MX37.4

Bit 7..4	0 to 15, CD-audio left volume in 3 dB step
Bit 3..0	0 to 15, CD-audio right volume in 3 dB step

0	-45 dB (off)
15	0 dB (maximum volume)

MX2E external line left/right volume
Ghost register

Write

Bit 7	MX38.7
Bit 6	MX38.6
Bit 5	MX38.5
Bit 4	MX38.4
Bit 3	MX39.7
Bit 2	MX39.6
Bit 1	MX39.5
Bit 0	MX39.4

Bit 7	MX38.3
Bit 3	MX39.3

Read

Bit 7	MX38.7

Bit 6	MX38.6
Bit 5	MX38.5
Bit 4	MX38.4
Bit 3	MX39.7
Bit 2	MX39.6
Bit 1	MX39.5
Bit 0	MX39.4

Bit 7..4	0 to 15, external line left volume in 3 dB step
Bit 3..0	0 to 15, external line right volume in 3 dB step

0	-45 dB (off)
15	0 dB (maximum volume)

Sound Blaster 16 (Physical Register):

MX30	master left volume
Read/write	
Default 90h	

Bit 7..3	0 to 31, master left volume in 1.5 dB step
Bit 2..0	reserved

0	-46 dB (off)
31	0 dB (maximum volume)

MX31	master right
Read/write	
Default 90h	

Bit 7..3	0 to 31, master right volume in 1.5 dB step
Bit 2..0	reserved

0	-46 dB (off)
31	0 dB (maximum volume)

MX32	digital audio left volume
Read/write	
Default 90h	

Bit 7..3	0 to 31, digital audio left volume in 1.5 dB step
Bit 2..0	reserved

0	-46 dB (off)
31	0 dB (maximum volume)

MX33	digital audio right volume
Read/write	
Default 90h	

Bit 7..3	0 to 31, digital audio right volume in 1.5 dB step
Bit 2..0	reserved

0	-46 dB (off)
31	0 dB (maximum volume)

MX34	music left volume
Read/write	
Default 90h	

Bit 7..3	0 to 31, music left volume in 1.5 dB step
Bit 2..0	reserved
0	-46 dB (off)
31	0 dB (maximum volume)

MX35 music right volume
Read/write
Default 90h

Bit 7..3	0 to 31, music right volume in 1.5 dB step
Bit 2..0	reserved
0	-46 dB (off)
31	0 dB (maximum volume)

MX36 CD-audio left volume
Read/write
Default 00h

Bit 7..3	0 to 31, CD-audio left volume in 1.5 dB step
Bit 2..0	reserved
0	-46 dB (off)
31	0 dB (maximum volume)

MX37 CD-audio right volume
Read/write
Default 00h

Bit 7..3	0 to 31, CD-audio right volume in 1.5 dB step
Bit 2..0	reserved
0	-46 dB (off)
31	0 dB (maximum volume)

MX38 external line left volume
Read/write
Default 00h

Bit 7..3	0 to 31, external line left volume in 1.5 dB step
Bit 2..0	reserved
0	-46 dB (off)
31	0 dB (maximum volume)

MX39 external line right volume
Read/write
Default 00h

Bit 7..3	0 to 31, external line right volume in 1.5 dB step
Bit 2..0	reserved
0	-46 dB (off)
31	0 dB (maximum volume)

MX3A microphone volume
Read/write
Default 00h

Bit 7..3	0 to 31, microphone volume in 1.5 dB step
Bit 2..0	reserved

0	-46 dB (off)
31	0 dB (maximum volume)

MX3B PC speaker/mono input volume
Read/write
Default 00h

Bit 7,6	0 to 3, mono input volume in 6 dB step
Bit 5..0	reserved

0	-18 dB (off)
3	0 dB (maximum volume)

MX3C output mixer control 1
Read/write
Default 1Fh

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
Read/write
Default 15h

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
Read/write
Default 0Bh

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
- 00 gain = 1
- 01 gain = 2
- 10 gain = 4
- 11 gain = 4

Bit 5..0 reserved

MX40 input right mixer gain
Default 00h
Read/write
Bit 7,6 input right mixer gain control
- 00 gain = 1
- 01 gain = 2
- 10 gain = 4
- 11 gain = 4

Bit 5..0 reserved

MX41 output left mixer gain
Read/write
Default 40h
Bit 7,6 output left mixer gain
- 00 gain = 1
- 01 gain = 2
- 10 gain = 4
- 11 gain = 4

Bit 5..0 reserved

MX42 output right mixer gain
Read/write
Default 40h
Bit 7,6 output right mixer gain
- 00 gain = 1
- 01 gain = 2
- 10 gain = 4
- 11 gain = 4

Bit 5..0 reserved

MX43 microphone automatic gain control (AGC)
Default 00h
Dummy read/write register
Bit 7..1 reserved
Bit 0 dummy read/write bit AGC enable

MX44 treble left control
Default 80h
Dummy read/write register
Bit 7..4 dummy read/write bit
Bit 3..0 reserved

MX45 treble right control
Default 80h
Dummy read/write register
Bit 7..4 dummy read/write bit
Bit 3..0 reserved

MX46 bass left control
Default 80h
Dummy read/write register
Bit 7..4 dummy read/write bit
Bit 3..0 reserved

MX47 bass right control
Default 80h
Dummy read/write register
Bit 7..4 dummy read/write bit
Bit 3..0 reserved

MX4B mono input/output control
Read/write
Default 80h
Bit 7, 6 mono recording control
- 00 recording left channel only
- 01 recording right channel only
- 10 recording (left channel / 2 + right channel / 2)
- 11 recording (left channel / 2 + right channel / 2)

Bit 5 mono output enable
- 0 mute
- 1 enable output buffer

Bit 4..0 reserved

Mono output = (left line out / 2) + (right line out / 2)

Mono output is the analog sum of left and right channel of the output mixer. It is after output gain control, before 3D processor.
Mono output should be attenuated by 6dB to prevent clipping. Bit 5 is the output mute control.

MX4C output mixer control 2
Read/write
Default 1Fh

Bit 7~5 Reserved
Bit 4 music left enable
Bit 3 music right enable
Bit 2 digital audio D/A left enable
Bit 1 digital audio D/A right enable
Bit 0 mono input enable
- 0 mute
- 1 enable audio into output mixer

MX4D microphone input gain control
Read/write
Default 02h Power up value changeable
Bit 7 Output mixer to input mixer control
- 0 Disable
- 1 Enable (Output mixer L/R → Input mixer L/R)

Bit 6~2 Reserved
Bit 1~0 Microphone Gain Control

Bit 1	Bit 0	Gain
0	0	1 (0dB)
0	1	12
1	0	16
1	1	20

MX4E analog reference voltage control register
Read/write
Power up value changeable.
Default 23h
Bit 7..6 reserved
Bit 5..4 current control
Bit 3..0 reference voltage control

MX4F
Read/write
Default 20h
Power up value changeable
Bit 7 CODEC test mode control
1 Test mode (Primary FIFO → FM D/A, SB A/D → SB D/A
Enable internal Music resistor for test)
0 normal mode
Bit 6 CP Test mode control
1 CP test mode (Enable SBAD/SBDA clock and power)
0 Normal mode
Bit 5 SB D/A 3dB freq control bit (HLDA_d)
1 20KHz
0 8KHz
Bit 4 reserved
Bit3..1 analog test bit
1 disable
0 enable
Bit3 DABLOCK.INTERP (MUINT_d)
Bit2 ADBLOCK.SH (MUSH_d)
Bit1 ADBLOCK.LPF (MULPF_d)
Bit0 reserved

MX50 3-D sound effect control register
Default A4h
Read/write
Power up value changeable.
Bit 7 normal sound output control(for testing only)
0 mute normal output
1 connect normal output

Bit 6 3-D sound output control
0 mute 3-D sound output
1 connect 3-D sound output

Bit 5..4 3-D block lpf 3dB freq control
00 1.25KHz
01 2.5KHz
10 3.75KHz
11 5.0KHz

Bit 3 reserved

Bit 2..0 3-D sound vs normal sound ratio control
000 1/4
001 2/4
010 3/4
011 4/4
100 5/4
101 6/4
110 7/4
111 8/4

MX51 3-D sound BBD time delay control register
default 14h
read/write
Power up value changeable.
Bit 7..6 accuracy of readback pattern from 3-D delay-line(for testing)
- 00 12 bits
- 01 11 bits
- 10 10 bits
- 11 9 bits

Bit 5 3-D delay-line resource select (for testing)
- 0 from 3-D front ADC
- 1 from input pattern register

Bit 4 3-D sound block power control
- 0 enable power
- 1 cut off power

Bit 3..0 BBD time delay control

delay time = 1.142ms * (1 + bit 3..0)

bit 3..0	delay time
0000	1.142ms
0001	2.284ms
.	.
.	.
.	.
1111	18.272ms

MX52 Auto-Mute 3D control and VP, VN control
Read/write
Power up value changeable.
Default 80h
Bit 7 Auto-Mute select
- 0 Enable
- 1 Disable

Bit 6 Detecting period select
- 0 60 ms
- 1 120 ms

Bit 5,4 Auto-Mute ON→ OFF input bit select
- 00 Bit 11~9
- 01 Bit 11~8
- 10 Bit 11~7
- 11 Bit 11~6

Bit 3,2 Auto-Mute OFF→ON input bit select
- 00 Bit 11~9
- 01 Bit 11~8
- 10 Bit 11~7
- 11 Bit 11~6

Bit 1,0 Vref-P/Vref-N control
- 00 1.68V
- 01 2.11V
- 10 2.53V
- 11 2.95V

MX53 Analog block control register
Read/write
Default 00h
Bit 7 AD Zero input control

0 Disable
1 Enable (Disconnect SH/LPF from AD)
Bit 6 LPF clock control
0 Clock running
1 Clock stopped
Bit 5 SH Zero input control
0 Disable
1 Enable (Disconnect LPF from SH)
Bit 4 SB AD/DA clock select
0 FXTALB + DFF
1 FXTALB
Bit 3 CKLP1/CKLP2 exchange control
0 CLKP1↔CKINT2, CLKP2↔CKINT1
1 CLKP1↔CKINT1, CLKP2↔CKINT2
Bit 2 SH clock control
0 Clock running
1 Clock stopped
Bit 1,0 Reserved

MX54 3-D delay-line pattern register (for testing)
Default 00h
Read
If subsequent access odd times (i.e, 1st, 3rd, 5th...)
Bit 7..0 3-D delay-line output pattern high byte
If subsequent access even times (i.e, 2nd, 4th, 6th...)
Bit 7..4 3-D delay-line output pattern low byte (4 bits only)
Stuff LSB with "0" according to MX51.7..6.
Bit 3..0 reserved

Write
If subsequent access odd times (i.e, 1st, 3rd, 5th...)
Bit 7..0 input pattern high byte to 3-D delay-line
If subsequent access even times (i.e, 2nd, 4th, 6th...)
Bit 7..4 input pattern low byte to 3-D delay-line (4 bits only)
Bit 3..0 reserved
After writing a whole pattern (12 bits), a pulse would be generated to shift the delay-line.
After each accessing to pattern register, the internal H/L byte pointer will toggle automatically.
After power on reset, the internal H/L byte pointer points to high byte.

Sound Blaster 16 Configuration Register:

MIXER.80 Sound Blaster interrupt setup (Dummy register for compatibility)
Read
Write if CR0.7=1
Default 00h

Bit 7..5 reserved (read as 1)
Bit 4 IRQ11
Bit 3 IRQ10
Bit 2 IRQ7
Bit 1 IRQ5
Bit 0 IRQ9

0 : disable interrupt
1 : enable interrupt

Only 1 bit can be set at any time by software.

MIXER.81 Sound Blaster DMA setup

Read
Write if CR0.7=1
Default 00h

Bit 7..4	reserved, read as 0
Bit 3	DMA3 for SB DMA command
Bit 2	reserved, read as 0
Bit 1	DMA1 for SB DMA command
Bit 0	DMA0 for SB DMA command

0: disable DMA
1: enable DMA

Only 1 bit can be set at any time by software.

Change this register will set GCR99.26~24 with the correct value.

MIXER.82 interrupt status
Read only
Default 00h

Bit 7,6	reserved
Bit 5	CR1E type DMA interrupt request
Bit 4,3	reserved
Bit 2	MPU-401 MIDI interrupt request
Bit 1	BX type command DMA interrupt request
Bit 0	non-BX type command DMA interrupt request

0 : no interrupt
1 : interrupt triggered

Control Register Definition:

CR0 Sound Blaster configuration
Default 06h
Read/write

Bit 7	MX80, MX81 write protection
Bit 6..3	reserved, read as 0
Bit 2	DMA mode control for 90h command
Bit 1	continuous DMA mode control
Bit 0	reserved

Bit 7	0	MX80 and MX81 can not be changed
	1	MX80 and MX81 can be changed

Bit 2	0	IRQ controlled 90h command (SB16 version)
	1	FIFO controlled 90h command (SBPRO version)

BIT 1	0	IRQ controlled continuous DMA mode
	1	FIFO controlled continuous DMA mode

CR2 MISC.control
Default 00h
Read/write

Bit 7..6	Reserved	
Bit 5	0	use crystal as ALS4000 system clock source
	1	use external input clock as ALS4000 system clock source and cut off

internal crystal oscillation circuit.
Bit 4..0 reserved, read as 0

CR3 configuration
Default 03h
Bit 7 r/w digital loop back test enable
Bit 6..2 reserved
Bit 1,0 chip version number (read as 11)

Bit 1	Bit 0	
0	0	ALS007SP
0	1	ALS100
1	0	ALS007/WTA2000
1	1	ALS200/ALS110//ALS4000/ALS120

CR3.7	0	secondary PCM FIFO recording from A/D
	1	secondary PCM FIFO recording from the primary PCM FIFO

The output of primary PCM FIFO can directly feed back as the input to the secondary PCM FIFO instead of the Sound Blaster A/D converter. This implement PCM FIFO digital loop back test. This mode is disabled at default. The operation of secondary PCM FIFO is fully controlled by CR1E register. CR3.7 just selects the input source of secondary PCM FIFO.

CR17 FIFO status
Default 00h
Bit 7-3 read only
Bit 2-0 read/write

Bit 7	1	secondary PCM FIFO overrun flag
Bit 6	X	reserved
Bit 5	1	primary PCM FIFO underrun flag
Bit 4	1	MIDI output FIFO full flag
Bit 3	1	MIDI input FIFO overrun flag
Bit 2	1	flush MIDI in and out FIFO
Bit 1	1	flush primary and secondary PCM FIFO
Bit 0	1	active XRST_ to reset internal OPL3

During full duplex running mode, if the secondary PCM FIFO is full when new sample arrives, bit 7 will be set.
Any time during DMA playback, if the primary FIFO is empty when new sample is needed, bit 5 will be set.
Any time midi output FIFO is full, bit 4 will be set.
Any time a new midi input data is received and the midi input FIFO is full, bit 3 will be set.
Reading this register will clear all flags(ie. Bit 7-3).
Bit 2-0 are sticky, that is, hardware will not clear them automatically and it is needed for software to clear the bits.

CR18 ESP major version number
Read/write
Default 03h

CR19 ESP minor version number
Read/write
Default 02h

CR1A MPU401 MIDI UART mode control
CR1A.7-5 R/W
CR1A.4-0 R
Default 0Xh

Bit 7	0	internal midiin from external midiin pad
	1	internal midiin from internal midiout
Bit 6	0	internal midiout to external midiout pad
	1	external midiin pad to external midiout pad
Bit 5	0	regular midi clock
	1	fast midi clock(14.318MHz)
Bit 4	X	reserved
Bit 3	0	MPU401 midi at pass-thru mode
	1	MPU401 midi at UART mode
Bit 2..0	X	reserved

When midi is in **pass-thru** mode, midi clock is stopped. Internal midiin is tied high and external midiout is connected to external midiin pad.

In MPU-401 UART mode, the internal midiin input, the external midiout output and the midi clock are controlled by CR1A.7-5 bits.

In midi loop back mode, when midiin FIFO is full, midi clock will be stop until midiin is not full. This will avoid midi FIFO overrun.

CR1C secondary recording PCM FIFO low byte block length

CR1D secondary recording PCM FIFO high byte block length

Block length = # of 8 or16 bit sample -1

CR1E secondary recording PCM FIFO control
Default 00h

Bit 7	0	stop secondary FIFO DMA
	1	run secondary FIFO DMA
Bit 6	0	normal
	1	pause secondary FIFO DMA
Bit 5	0	mono sample
	1	stereo sample
Bit 4	0	unsigned (0--80h--0ffh or 0--8000h--0ffffh)
	1	signed (80h--0--7fh or 8000h--0--7fffh)
Bit 3	X	reserved
Bit 2	0	16 bit sample
	1	8 bit sample
Bit 1	X	reserved
Bit 0	X	reserved

Only CR1E can control the secondary PCM FIFO. To avoid conflict, when both PCM FIFO will be active, these two FIFO should run under the same sample frequency. The secondary PCM FIFO is for recording only. SB AD and input mixer power line controlled by CR1E.7.

CR3A MISC control register
Read/write
Default 00h

Bit 7~3		reserved
Bit 2		FIFO CRC check control
	0	Normal
	1	Clear CRC-32 shift register contents

Bit 1		reserved
Bit 0	0	disable SB16 E3 command
	1	enable SB16 E3 command

CR3B CRC-32 Byte 0
Default : 00h
Bit 7~0 CRC-32 bit 7~0

CR3C CRC-32 Byte 1
Default : 00h
Bit 7~0 CRC-32 bit 15~8

CR3D CRC-32 Byte 2
Default : 00h
Bit 7~0 CRC-32 bit 23~16

CR3E CRC-32 Byte 3
Default : 00h
Bit 7~0 CRC-32 bit 31~24

CRC-32 for FIFO check :

ALS4000 build-in CRC-32 check circuit for speed up testing. It can help verification in development stage. To start CRC check, toggle CR3A.2 1 time. The clock of CRC-32 shift register is the clock of SB primary FIFO. When CPU read CR3B~3E,ALS4000 return CRC-32 shift register content to CPU.

Test Mode Access:

Access GCR90 will enter the test mode you set.

Power Management:

Try the best power management for both active and inactive stage of ALS4000 for notebook application.

Pay special attention to data bus and address bus inside ALS4000 because this may be the very heavy power consumption source.

Use internal block chip select signal to be the block power control signal as much as possible.

~~The pull up resistors on any power up configuration pin must be disconnected from VCC after reset.~~

FIFO Control Attention:

Whenever new continuous DMA command for 8/16 bit wave playback is received, SB16 ESP should always flush primary PCM FIFO.

ALS4000 should has a primary FIFO r/w counter reset indicator, system reset or ESP RESET should reset this indicator. When current DMA operation is 8 bit stereo or 16 bit mono, if DMA transfer times is not even, this indicator should indicate bit 0 of FIFO r/w counter must be reset by ESP when new DMA command is received. When current DMA operation is 16 bit stereo, if DMA transfer times is not multiple times of 4, this indicator should indicate that bit 0 and bit 1 of FIFO r/w counter must be reset by ESP when new DMA command is received.

Any time when new DMA command is received, this indicator must be reset after primary FIFO r/w counter control.

MIDIOUT FIFO Size : 16 bytes
MIDIIN FIFO Size : 32 bytes

Appendix A : ESP command set

* Undocumented
$ Sound Blaster Pro only
# Sound Blaster 16 only

0Xh reserved
1Xh set audio output mode for 4:1 ADPCM 2 to 8 bit playback
2Xh reserved
3Xh reserved
4Xh set sample rate and continuos/special DMA block length
5Xh reserved
6Xh reserved
7Xh set audio output for all ADPCM, 8 bit playback
8Xh output silence
9Xh 8 bit special DMA mode playback
AXh reserved
BXh 16 bit DMA audio output
CXh 8 bit DMA audio output
DXh control DMA and speaker
EXh ESP version and diagnostic test
FXh test IRQ and ESP ROM

All 8-bit ESP command is unsigned PCM except CXh command.

All 8-bit ESP command is MONO except command 14h,1Ch,9Xh and CXh

8 Bit And 4:1 ADPCM 2 To 8 Bit Output
1Xh

Bit 3 0 direct/normal DMA mode
1 continuous DMA mode

Bit 2 0 direct mode
1 DMA mode

Bit 1 0 8 bit data
1 4:1 ADPCM 2 to 8 bit mode

Bit 0 0 normal
1 the first ADPCM block with reference byte

10h 8 bit direct mode output

a. ESP_WRITE(10h)
b. ESP_WRITE(single-sample)
c. Wait for next sample time, go to a.

14h 8 bit normal DMA output

a. ESP_WRITE(14h)
b. ESP_WRITE(length.low)
c. ESP_WRITE(length.high)

Length = # of byte transfer - 1

ESP will generate an interrupt after the specified size of data transferred.

16h 4:1 ADPCM 2 to 8 bit normal DMA output

a. ESP_WRITE(16h)

b. ESP_WRITE(length.low)
c. ESP_WRITE(length.high)

Length = # of byte transfer - 1

ESP will generate an interrupt after the specified size of data transferred.

17h first 4:1 ADPCM 2 to 8 bit normal DMA output

a. ESP_WRITE(17h)
b. ESP_WRITE(length.low)
c. ESP_WRITE(length.high)

Length = # of byte transfer - 1

ESP will generate an interrupt after the specified size of data transferred.

1Ch 8 bit continuous DMA output

ESP_WRITE(1Ch)

ESP will generate an interrupt for every specified block size transferred.

1Eh 4:1 ADPCM 2 to 8 bit continuous DMA output

ESP_WRITE(1Eh)

ESP will generate an interrupt for every specified block size transferred.

1Fh first 4:1 ADPCM 2 to 8 bit continuous DMA output

ESP_WRITE(1Fh)

ESP will generate an interrupt for every specified block size transferred.

Set Sample Rate And Continuous/Special DMA Block Length
4Xh

40h set sample rate time constant

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

Channel = 1 for mono or BX,CX type command, 2 for stereo

a. ESP_WRITE(40h)
b. ESP_WRITE(time constant)

#41h set sample rate

a. ESP_WRITE(41h)
b. ESP_WRITE(frequency.high)
c. ESP_WRITE(frequency.low)

Sampling frequency = 4 KHz TO 48 KHz, either mono or stereo

48h set block length for continuous and special DMA

a. EPS_WRITE(48h)
b. ESP_WRITE(length.low)
c. ESP_WRITE(length.high)

Length = # of byte transfer - 1

ESP will generate an interrupt after transferred the block of data.

After reset, the default block size is 2048. Length = 07FFh

All ADPCM 8 Bit Output
7Xh

Bit 3	0	normal DMA mode
	1	continuous DMA mode

Bit 2	Bit 1	
0	0	reserved
0	1	4:1 ADPCM 2 to 8 bit mode
1	0	2:1 ADPCM 4 to 8 bit mode
1	1	3:1 ADPCM 2.6 to 8 bit mode

Bit 0	0	normal
	1	the first ADPCM block with reference byte

*72h 4:1 ADPCM 2 to 8 bit normal DMA output

a. ESP_WRITE(72h)
b. ESP_WRITE(length.low)
c. ESP_WRITE(length.high)

Length = # of byte transfer - 1

ESP will generate an interrupt after the specified size of data transferred.

*73h first 4:1 ADPCM 2 to 8 bit normal DMA output

a. ESP_WRITE(73h)
b. ESP_WRITE(length.low)
c. ESP_WRITE(length.high)

Length = # of byte transfer - 1

ESP will generate an interrupt after the specified size of data transferred.

74h 2:1 ADPCM 4 to 8 bit normal DMA output

a. ESP_WRITE(74h)
b. ESP_WRITE(length.low)
c. ESP_WRITE(length.high)

Length = # of byte transfer - 1

ESP will generate an interrupt after the specified size of data transferred.

75h first 2:1 ADPCM 4 to 8 bit normal DMA output

a. ESP_WRITE(75h)
b. ESP_WRITE(length.low)
c. ESP_WRITE(length.high)

Length = # of byte transfer - 1

ESP will generate an interrupt after the specified size of data transferred.

76h 3:1 ADPCM 2.6 to 8 bit normal DMA output

a. ESP_WRITE(76h)
b. ESP_WRITE(length.low)
c. ESP_WRITE(length.high)

Length = # of byte transfer - 1

ESP will generate an interrupt after the specified size of data transferred.

77h first 3:1 ADPCM 2.6 to 8 bit normal DMA output

a. ESP_WRITE(77h)
b. ESP_WRITE(length.low)
c. ESP_WRITE(length.high)

Length = # of byte transfer - 1

ESP will generate an interrupt after the specified size of data transferred.

*7Ah 4:1 ADPCM 2 to 8 bit continuous DMA output

ESP_WRITE(7Ah)

ESP will generate an interrupt for every specified block size transferred.

*7Bh first 4:1 ADPCM 2 to 8 bit continuous DMA output

ESP_WRITE(7Bh)

ESP will generate an interrupt for every specified block size transferred.

7Ch 2:1 ADPCM 4 to 8 bit continuous DMA output

ESP_WRITE(7Ch)

ESP will generate an interrupt for every specified block size transferred.

7Dh first 2:1 ADPCM 4 to 8 bit continuous DMA output

ESP_WRITE(7Dh)

ESP will generate an interrupt for every specified block size transferred.

7Eh 3:1 ADPCM 2.6 to 8 bit continuous DMA output

ESP_WRITE(7Eh)

ESP will generate an interrupt for every specified block size transferred.

7Fh first 3:1 ADPCM 2.6 to 8 bit continuous DMA output

ESP_WRITE(7Fh)

ESP will generate an interrupt for every specified block size transferred.

Output Silence

8Xh

Bit 3..0 X reserved

80h continuous mode silence audio for a duration

a. ESP_WRITE(80h)
b. ESP_WRITE(duration.low)
c. ESP_WRITE(duration.high)

Duration = # of silence sample period - 1

After each specified duration elapses, ESP will generate an interrupt. During silence period, ESP out 0x80 to PCM D/A.

8 Bit Special DMA Playback
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 RESET-ESP() to end the continuous special DMA, all other parameters remains the same after RESET-ESP( ).

Special DMA playback:
MONO MX0E.1 = 0
Stereo MX0E.1 = 1

Default MONO after system reset or mixer reset or ESP reset

Bit 3	1	audio input
Bit 3	0	audio output
Bit 2	X	reserved
Bit 1	X	reserved
Bit 0	0	continuous DMA
Bit 0	1	non-continuous DMA

$90h 8 bit continuous special DMA output

ESP_WRITE(90h)

ESP will generate an interrupt for every specified block size transferred.

ALS4000 use CR0.2 to control the DMA running mode of 90h command. When CR0.2 is 1, ESP will continue DMA transfer no matter whether the interrupt is acknowledged, this is SBPRO 90 command. When CR0.2 is 0, ESP will continue DMA transfer after the interrupt is acknowledged, this is SB16 90h command.

$91h 8 bit non-continuous special DMA output

ESP_WRITE(91h)

ESP will generate an interrupt after the specified size of data transferred.

For non-continuous special DMA output, every time when ESP receives the 91h command, ESP will use the block length that most recently set by command 48h to begin the special DMA transfer. So if every block of DMA data is the same size, software needs only set block length one time by using command 48h. When each block is transferred over, software needs only send 91h command to ESP, the DMA transfer will then continue with previous block length.

16 Bit DMA Audio Output

#BXh

Bit 3	0	audio output
	1	audio input
Bit 2	0	non-continuous DMA
	1	continuous DMA
Bit 1	X	reserved
Bit 0	X	reserved

FIFO will reset when ESP receives any digital audio I/O command.
FIFO is always on.

a. ESP_WRITE(BXh)
b. ESP_WRITE(mode)
c. ESP_WRITE(length.low)
d. ESP_WRITE(length.high)

Mode

Bit 7..6	X	reserved
Bit 5	0	MONO
	1	stereo
Bit 4	0	unsigned (0--8000h--FFFFh)
	1	signed (8000h--0--7FFFh)
Bit 3..0	X	reserved

Length = # of 16 bit sample -1

ESP will generate an interrupt after the specified size of data transferred (if non-continuous) or every block(if continuous).

8 Bit DMA Audio Output
#CXh

Bit 3	0	audio output
	1	audio input
Bit 2	0	non-continuous DMA
	1	continuous DMA
Bit 1	X	reserved
Bit 0	X	reserved

FIFO will reset when ESP receives any digital audio I/O command.
FIFO is always on.

a. ESP_WRITE(CXh)
b. ESP_WRITE(mode)
c. ESP_WRITE(length.low)
d. ESP_WRITE(length.high)

Mode

Bit 7..6	X	reserved
Bit 5	0	MONO
	1	stereo
Bit 4	0	unsigned (0--80h--FFh)
	1	signed (80h--0--7Fh)
Bit 3..0	X	reserved

Length = # of 8 bit sample -1

ESP will generate an interrupt after the specified size of data transferred (if non-continuous) or every block(if continuous).

Control DMA And Digital Audio
DXh

D0h pause non BX type DMA transfer

ESP_WRITE(D0h)

The DMA request is stopped after this command. Internal FIFO will continue until the FIFO is empty (playback) or full (record). The DMA request will resume after command D4h or any of new DMA command is issued.

D1h turn digital audio on

ESP_WRITE(D1h)

Set digital audio status flag for D8h command.

D3h turn digital audio off

ESP_WRITE(D3h)

Reset digital audio status flag for D8h command.
Audio status flag is reset after system reset or ESP_RESET().

D4h resume non BX type DMA transfer

ESP_WRITE(D4h)

The DMA request that is suspended by the command D0h is enable again. The internal FIFO is working as usual in pause or resume DMA mode.

#D5h pause BX type DMA mode transfer

ESP_WRITE(D5h)

The DMA request is stopped after this command. Internal FIFO will continue until the FIFO is empty (playback) or full (record). The DMA request will resume after command D6h or any of new DMA command is issued

#D6h resume BX type DMA mode transfer

ESP_WRITE(D6h)

The DMA request that is suspended by the command D5h is enable again. The internal FIFO is working as usual in pause or resume DMA mode. This command is no use to non-BX type command DMA transfer.

D8h get digital audio status

a. ESP_WRITE(D8h)
b. ESP_READ(status)
c. Status = 00h (digital audio off) or FFh (digital audio on)

#D9h exit current BX type continuous DMA transfer

ESP_WRITE(D9h)

Causes the 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 ESP. ESP-RESET() or any of new DMA command

should reset this flag.

DAh exit current non-BX type continuous DMA transfer

ESP_WRITE(DAh)

Causes the 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 ESP. ESP-RESET() or any of new DMA command should reset this flag.

ESP Version And Diagnostic
EXh

*E0h read/write diagnostic test

a. ESP_WRITE(E0h)
b. ESP_WRITE(test-data)
c. ESP_READ(result)
d. If result = bit invert of test-data, ESP is working

E1h get ESP version number

a. ESP_WRITE(E1h)
b. ESP_READ(major.version)
c. ESP_READ(minor.version)

Sound Blaster Pro default version 3.02
Sound Blaster 16 default version 4.02

ALS4000 default **3.02**

Major version = CR18
Minor version = CR19

*E2 DMA testing

Computes The Subroutine's Starting Address for Digital Sound
Playback or Recording According to The Dedicated Algorithm.
Sends Back The Result To System via DMA Method(Two Bytes Transferred).

a. ESP_WRITE(E2h)
b. ESP_WRITE(first byte b1),return r1 via DMA
c. ESP_WRITE(E2h)
d. ESP_WRITE(second byte b2),return r2 via DMA

Formula for r1:
r1 = (b1.7..5 + 1) * 0x40 + b1.3 * 0x10 + b1.0 * 0x02 - b1 (this is 8 bit unsigned operation)
Formula for r2:
1. r2_h = 0xe0 + b2.4 * 0x20 - b2.6 * 0x80
2. r2_h = r2_h + b1 & 0xe0 - b2 & 0xf0
3. r2_l = 0x05 + b2.1 * 0x04 - b2 & 0x0f
4. r2 = r2_h & 0xf0 + r2_l & 0x0f + b1.0 * 0x01 - b1 & 0x1e
5. r2 = r2 + b1.3 * 0x10
(this is 8 bit unsigned operation)

*E3 copyright message

a. ESP_WRITE(E3h)
b. ESP_READ(message)

Message :
TEXT "COPYRIGHT (C) CREATIVE TECHNOLOGY LTD, 1992.",0h)
HEX 43 4F 50 59 52 49 47 48 54 20 28 43 29 20
43 52 45 41 54 49 56 45 20 54 45 43 48 4E 4F 4C 4F 47 59 20
4C 54 44 2C 20 31 39 39 32 2E 00
This command is valid only when CR3A.0 = 1.

*E4 send test byte for command E8h

a. ESP_WRITE(E4h)
b. ESP_WRITE(test-data)

*E8 read diagnostic byte

a. ESP_WRITE(E4h)
b. ESP_WRITE(test-data)
c. ESP_WRITE (E8h)
c. ESP_READ(result)
d. If result = test-data, ESP is working

Testing
FXh

*F2h generate an interrupt for test

ESP_WRITE(F2h)

ESP will generate an interrupt immediately after this command.

*F8 read back dedicated byte 00
a. ESP_WRITE(F8h)
b. ESP_READ(result)
c. If result = 00,ESP is working

Appendix B : PCM Data Format

Digitized Sound Data Format And Order:

Length	Format	Min value	Mid value	Max value
8 Bit	Unsigned	00h	80h	FFh
8 Bit	Signed	80h	00h	7Fh
16 Bit	Unsigned	0000h	8000h	FFFFh
16 Bit	Signed	8000h	0000h	7FFFh

PCM Sample Order:

- **8 Bit Mono**

Byte No.	N	N+1	N+2	N+3	N+4
Sample	PCM 0	PCM 1	PCM 2	PCM 3	PCM 4

- **8 Bit Stereo With 1Xh And 9Xh Type Command**

Byte No.	2N	2N+1	2N+2	2N+3	2N+4
Sample	PCM0.R	PCM0.L	PCM1.R	PCM1.L	PCM2.R

- **8 Bit Stereo With Cxh Type Command**

Byte No.	2N	2N+1	2N+2	2N+3	2N+4
Sample	PCM0.L	PCM0.R	PCM1.L	PCM1.R	PCM2.L

- **16 Bit Mono**

Byte No.	2N	2N+1	2N+2	2N+3	2N+4
Sample	PCM0.LOW	PCM0.HIGH	PCM1.LOW	PCM1.HIGH	PCM2.LOW

- 16 Bit Stereo

Byte No.	4N	4N+1	4N+2	4N+3	4N+4
Sample	PCM0.L.Low	PCM0.L.High	PCM0.R.Low	PCM0.R.High	PCM1.L.Low

Appendix C : ESP/MPU401 Programming Guide

ESP Procedure:

ESP Read Data Procedure:

ESP_READ(return_byte)

1. If I/O read(ESP-RD-STATUS) bit 7 = 1 goto 2 else 1
2. Return_byte = I/O read(ESP-READ-DATA)

ESP Write Procedure:

ESP_WRITE(command/data)

1. If I/O read(ESP-WR-STATUS) bit7 = 0 goto 2 else 1
2. I/O write(ESP-COMMAND/DATA, command/data)

ESP Reset Procedure:

ESP_RESET( )

1. I/O write(ESP-RESET-PORT, 1), and wait 3 us
2. I/O write(ESP-RESET-PORT, 0)
3. ESP_READ(status), if status = AAh goto 4 else 3
4. End of reset

You can use ESP_RESET() procedure to immediately terminate the special DMA transfer.

Digitized Sound Transfer Method:

1. Direct Mode

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

Output:

a. ESP_WRITE(10h)
b. ESP_WRITE(next 8 bit mono PCM)
c. Wait until next sample time, goto a.

2. DMA Mode

2.1 Normal DMA Mode

ESP is programmed to make one transfer with a specified block size. At the end of transfer, the ESP will generate an interrupt and wait for next command.

2.2 Continuous DMA Mode

ESP is programmed to make continuous transfer to CODEC. After each transfer of a specified block size, 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.

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

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

2.3 Special DMA Mode

Once ESP is in the special DMA mode, it will not accept any further commands or data until the DMA mode is terminated by
Either

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

Or

2. For continuous special DMA mode, a ESP_RESET() is needed to exit special DMA mode.

The ESP_RESET() will only stop special DMA transfer, all other parameters remain the same.

I/O Transfer Rate Setup:

Either
1. Sound Blaster Pro (time constant)
Time constant = 256d - (1,000,000d / (channel * sampling rate))
Channel = 1 for mono or BX,CX type DMA command, 2 for stereo
a. ESP_WRITE(40h)
b. ESP_WRITE(time constant)

Or
2. Sound Blaster 16 (sampling frequency)
a. ESP_WRITE(41h)
b. ESP_WRITE(frequency.high)
c. ESP_WRITE(frequency.low)

Sampling frequency = 4 KHz to 48 KHz, either mono or stereo

MPU-401 MIDI Programming:

1. MPU-401 pass-thru mode
After reset, midi is in **pass-thru** mode. MIDIIN data is directly connected to MIDIOUT data.
Any write to **midi-data** will be ignored. MIDI-COMMAND will support either MIDI_RESET (0FFh) or ENTER_UART (03Fh).
Both command will return with interrupt and acknowledge byte (0FEh).

2. MPU-401 UART mode
After enter UART mode, all write to **midi-data** will go to MIDIOUT FIFO, any MIDIIN data will go to MIDIIN FIFO.

MIDI-STATUS bit 7 = MIDIIN FIFO empty.
MIDI-STATUS bit 6 = MIDIOUT FIFO full + MIDIOUT FIFO not empty when MIDI_RESET
MIDI INTERRUPT = MIDIIN FIFO not empty & not I/O read(MIDI-DATA)

To end the UART mode, send MIDI_RESET (0FFh) to **midi-command** port. MIDIIN FIFO will be flushed, any data in MIDIOUT FIFO will be output. When both FIFO are empty, MPU-401 will enter **pass-thru** mode .

Part II : Specification for PCI Interface

Note for Interrupt Service Routine(ISR):
Software developer should pay more attention to decide which request to be served.

For TSR :
1.Mask INTA# output. (Set GCR8C.15) ,check IRQ-STATUS and acknowledge IRQs.
2.Call corresponding **ISR**s (higher-priority ISR first)
3.If not own interrupt request, jump to old ISR.
4.Enable INTA# output capability again.(Clear GCR8C.15) and return to main program

Legacy function control emulation for DMAC

For DMA emulation or distributed DMA, refer to ***Appendix A***. Pay more attention to the two PCM FIFO control.

INTA# edge trigger mode:
If GCR8C.27=1, the acknowledge of INTA# acts as edge trigger. There are three internal flags which indicate SB, CR1E and MPU401 type interrupt request respectively. The corresponding flag should be set to one when internal interrupt request signal toggles from low to high, then INTA# should be asserted. ALS4000 should snoop PCI interrupt acknowledge cycle. If PCI interrupt acknowledge cycle with the correct vector occurs on the bus, one flag will be cleared (If more than one flag are set, clear the flag of SBIRQ first, CR1E second, MPU401 the last). INTA# should sustain low unless all the three flags are cleared. Note that these actions will **not** affect the interrupt status register.
If GCR8C.27=0, the acknowledge of INTA# acts just as normal level trigger.

PCI interrupt acknowledge cycle:
PCIIRQn=PCI configuration register 3Ch

- Address phase
  - AD[31:0] : not valid
  - C/BE#[3:0] : 0000
- Data phase
  - AD[31:8] : not valid
  - AD[7:3] : M_IVBA (GCRA7.7~3) if PCIIRQn bit 3=0
    S_IVBA (GCRA7.23~19) if PCIIRQn bit 3=1
  - AD[2:0] : PCIIRQn bit 2~0
  - C/BE#[3:0] : 1110

Snooping on PIC IO write operation:

Address	Function	Condition
20h	Write Bit4 to M_ICWS (GCRA7.15)	Always
21h	No Action	M_ICWS=0
	Write Bit7~3 to M_IVBA (GCRA7.7~3) and then clear M_ICWS	M_ICWS=1
A0h	Write Bit4 to S_ICWS (GCRA7.31)	Always
A1h	No Action	S_ICWS=0
	Write Bit7~3 to S_IVBA (GCRA7.23~19) and then clear S_ICWS	S_ICWS=1

PCI Power Management Interface and Event:

ALS4000 is capable of D0, D2 and D3 power management states.
When in D2 or $D3_{hot}$ state, ALS4000 must not respond to PCI transactions targeting its I/O spaces. It must always be ready to accept a PCI configuration access when in D2 or $D3_{hot}$ state.
Whenever PCI RST# is asserted, ALS4000 will return to D0 (uninitialized state).
While device has been opened, it must be put into D0 state before starting playing. Software can put the device into any state it wants. The only exception is, during digital wave input/output, if ESP is processing data to/from PCM FIFO, and no “PAUSE” command has been issued, the device must stay in D0 state, any attempt to write GCRA6 and PMCSR will be terminated normally but ignored during this period.

Warm (soft) reset
If ALS4000 has been put into $D3_{hot}$ state, an attempt to change the power state to D0 will cause a warm reset. An internal reset signal will be generated just as PCI RST# be asserted and the device will return to D0 uninitialized state. Note that all of the PCI signal drivers must remain disabled for the duration of the $D3_{hot}$ to D0 uninitialized state transition.

D0
Power is on. Device is operating.

D1
Not implemented. Any attempt to put the device to this state will be discarded and the power state should remain the original state.

D2
Power down internal FM synthesizer, FM D/A, SB D/A, gameport block. All the information of mixer and global control register should remain. Internal **Audio Inactivity Timer** started. When **Audio Inactivity Timer** expires, **PME#** should be asserted.

$D3_{hot}$
ALS4000 is drawing minimal power. The only power consumption is for audio activity detection circuit, internal registers and for PCI configuration read/write. If audio activity is detected, **PME#** should be asserted to wake up from sleeping. Only legal state transition is to D0. Any other attempt to change the power state is illegal and has no effect.

$D3_{cold}$
Power is removed.

Audio Activity Detection Circuit
The circuit is used to detect if there is any valid analog input on the pad. That is, if MIC is plugged into the phone-jack, or Line-in (or CD-audio) starts playing, system should be told to wake up the audio device. It is always running, even in $D3_{hot}$ state.

Audio Inactivity Timer
This timer starts counting only when ALS4000 is transitioned to D2 state. The timer default value is 30 seconds. In D2, each time when the timer expires, **PME#** should be asserted, and the counter restarts. Note that if audio activity is detected, the timer should restart counting from zero.

ALS4000 IO Space : 128 bytes

Address	Byte 3	Byte 2	Byte 1	Byte 0
IOBASE+00h	AC97-ACCESS(R/W)			
IOBASE+04h	IRQ-STATUS(R/W)	AC97-STATUS(R)	AC97-READ(R)	
IOBASE+08h	GCR-DATA (R/W)			
IOBASE+0Ch	reserved	SB-MPU-IRQ(R/W)	reserved	GCR-INDEX(R/W)
IOBASE+10h	ADLIB address 3*	ADLIB address 2*	ADLIB address 1*	ADLIB address 0*
IOBASE+14h	reserved	ESP-RST-PORT* / CR1E-ACK-PORT*	MIXER-DATA*	MIXER-INDEX*
IOBASE+18h	reserved	ESP-RD-DATA*	OPL-ADDR1(229)*	OPL-ADDR0(228)*
IOBASE+1Ch	ESP-RD-STATUS16*	ESP-RD-STATUS8*	reserved	ESP-CMD/DATA* / ESP-WR-STATUS*
IOBASE+20h	reserved	reserved	GAMEPORT(201)*	GAMEPORT(200)*
IOBASE+24h~2Fh	reserved			
IOBASE+30h	reserved	reserved	MIDI-STATUS(R)* / COMMAND(W)*	MIDI-DATA*
IOBASE+34h~3Fh	reserved			
IOBASE+40h~7Fh	DDMA address group			

IOBASE = Base address defined in configuration space
ALS4000 support DW/Word/Byte on GCR-DATA access.

Note: Only **BYTE** access is supported on those IO addresses that followed with an asterisk (*).

Hardware should translate and forward the IO cycles to SB core. If WORD or DWORD access occurs, HW should deal with the condition gracefully, i.e., the access cycle should be terminated normally. If write, the written data should be discarded. If read, the data driven out by ALS4000 is meaningless.

SB-MPU-IRQ Default : 00h IOBASE + 0Eh

Bit	Type	Function
7	R/W	SB DMA interrupt request (CR1E, BX or non-BX type)
6	R/W	CR1E type interrupt request
5		reserved
4	R/W	MPU MIDI interrupt request
3:0		reserved

❶Write the register with "1" will clear the corresponding IRQ flag.
❷Write the register with "0" will keep the corresponding IRQ flag.
❸Acknowledge IRQ in PCI block will **not** effect the "interrupt logic" in SB core.
❹Acknowledge IRQ in SB core will clear IRQ flag in PCI block.

GCR-DATA Default : XXXXXXXXh IOBASE + 08-0Bh

Bit	Type	Function
31:0	R/W	Global control register data

GCR-Index Default : XXh IOBASE + 0Ch

Bit	Type	Function
7:0	R/W	Global control register index

ALS4000 Global Control Register Array :

Index Port : GCR-Index
Data Port : GCR-DATA

GCR8C Default 000X0000h Miscellaneous control

Bit	Type	Function
31:28		Reserved
27	R/W	INTA# trigger mode 0: level trigger(default) 1: edge trigger
26:20		Reserved
19:16	R	Chip revision number
15	R/W	INTA# mask control 1 Enable IRQ output 0 Disable IRQ output (Drive INTA# to inactive state)
14:0		Reserved

❶Chip revision number will increase by 1 when chip tape-out.

GCR90 Default : XXXX00XXh Test mode register

Bit	Type	Function
31:16		reserved
15:8	R/W	SB test mode register (Normal = 00h)
7:0		reserved

GCR91 Default : 00000000h DMA 0 starting address

Bit	Type	Function
31:24		Reserved
23:0	R/W	DMA channel 0 starting address SA0[23..0]

❶Stuff SA0[31..24] with 0 when implemented.

GCR92 Default : 00000000h DMA 0 mode register and base byte count

Bit	Type	Function
31:22		Reserved
21	R/W	Address increment/decrement select. 0 : increment 1 : decrement
20	R/W	Auto-initialization enable 0 : Disable 1 : Enable
19:18	R/W	Transfer mode 01 : Write 10 : Read Others : Reserved
17:16		Reserved

15:0	R/W	DMA channel 0 base byte count BBC0[15..0]

❶Write transfer ALS4000 → Memory
Read transfer ALS4000 ← Memory

GCR93 Default : 00000000h DMA 1 starting address

Bit	Type	Function
31:24		Reserved
23:0	R/W	DMA channel 1 starting address SA1[23..0]

❶Stuff SA1[31..24] with 0 when implemented.

GCR94 Default : 00000000h DMA 1 mode register and base byte count

Bit	Type	Function
31:22		Reserved
21	R/W	Address increment/decrement select. 0 : increment 1 : decrement
20	R/W	Auto-initialization enable 0 : Disable 1 : Enable
19:18	R/W	Transfer mode 01 : Write 10 : Read
17:16		Reserved
15:0	R/W	DMA channel 1 base byte count BBC1[15..0]

❶Write transfer ALS4000 → Memory
Read transfer ALS4000 ← Memory

GCR95 Default : 00000000h DMA 3 starting address

Bit	Type	Function
31:24		Reserved
23:0	R/W	DMA channel 3 starting address SA3[23..0]

❶Stuff SA[31..24] with 0 when implemented.

GCR96 Default : 00000000h DMA 3 mode register and base byte count

Bit	Type	Function
31:22		Reserved
21	R/W	Address increment/decrement select. 0 : increment 1 : decrement
20	R/W	Auto-initialization enable 0 : Disable 1 : Enable
19:18	R/W	Transfer mode 01 : Write 10 : Read
17:16		Reserved
15:0	R/W	DMA channel 3 base byte count BBC3[15..0]

❶Write transfer ALS4000 → Memory
Read transfer ALS4000 ← Memory

GCR99 Default B4000000h DMA emulation control

Bit	Type	Function
31	R	DMA 3 mask 1 : mask DMA operation (Default : 1)
30		Reserved
29	R	DMA 1 mask 1 : mask DMA operation (Default : 1)
28	R	DMA 0 mask 1 : mask DMA operation (Default : 1)
27		Reserved
26:24	R	SB DMA channel select, decoded from MX81
23		Reserved
22	R/W	Legacy-DMA IO timing 0: Subtractive 1: Positive decode
21	R/W	PCM buffer clear control 0: normal 1: clear
20	R/W	REQ# generated when PCM buffer is 0: half empty 1: not full (one layer or more empty)
19:17		Reserved
16	R	DMA channel group enable 0 : Enable 1 : Disable
15:4	R/W	DDMA base address DDMABA[15..4]
3		Reserved
2:0	R/W	DMA emulation mode select Bit 2~0= 000 no any DMA emulation function 001 DDMA enable

			010 Legacy-DMA read enable 011 WB-DMA (write back DMA register) enable 100 Burst mode for playback enable others reserved

❶When GCR99.2~0=000 or 100, internal bus master is independent with DMA group enable and DMA mask bit, it will always work if bus master enable bit is set in PCI configuration space.
❷When DMA group enable bit (GCR99.16) is set, it will disable DMA 0,1,3 emulation. When this bit is clear, DMA emulation is controlled by mask bit (GCR99.31,29,28).
❸If auto-initialized mode is disabled, corresponding bit of GCR99.31,29,28 will be set when TC0=1.

TC0=1 SBDMA=000 set GCR99.28
SBDMA=001 set GCR99.29
SBDMA=011 set GCR99.31

❹SB DMA 000 DMA 0
001 DMA 1
011 DMA 3
100 No DMA select
others Reserved

❷If GCR99.2~0=100, PCM buffer for playback will be used, that is, ALS4000 will perform burst memory read cycle. Else, ALS4000 will request bus to perform memory read to get one byte each time when internal DRQx is activated.
❸If GCR99.2~0=011, ALS4000 will request bus to perform IO write to update DMAC register each time after one byte has been transfered.
❹It is hardware's responsibility to detect legacy-DMA I/O timing and set or reset GCR99.22 correctly. Write this bit with "1" will clear this bit.

GCRA0 Default : 00XXXXXXh Primary PCM FIFO current address

Bit	Type	Function
31:24		reserved (read as 0)
23:0	R	Current address CA0[23..0] for primary FIFO

GCRA1 Default : XXXXXXXXh Primary PCM FIFO status and current byte count

Bit	Type	Function
31	R	FFLP status in DMA emulation block
30:18		Reserved
17	R	Request status, set to 1 if a request for primary FIFO is pending.
16	R	TC0 status, 1=TC0 is reached. Read and clear it.
15:0	R	Primary FIFO current byte count CBC0[15..0]

GCRA2 Default : 00000000h PCI start address for secondary PCM FIFO

Bit	Type	Function
31:24		reserved
23:0	R/W	PCI starting address SSA[23..0] for secondary FIFO

❶Stuff SSA[31..24] with 0 when implemented.

GCRA3 Default : 00000000h Secondary PCM FIFO base byte count

Bit	Type	Function
31:16		Reserved
15:0	R/W	Secondary base byte count SBBC[15..0]

❶Only auto-init and address increase mode supported while recording.

GCRA4 Default : 00XXXXXXh Secondary PCM FIFO current address

Bit	Type	Function
31:24		reserved (read as 0)
23:0	R	PCI current address CA1[23..0] for secondary FIFO

GCRA5 Default : XXXXXXXXh Secondary PCM FIFO status and current byte count

Bit	Type	Function
31:18		Reserved
17	R	Request status, set to 1 if a request for secondary FIFO is pending.
16	R	TC1 status, 1=TC1 is reached. Read and clear it.
15:0	R	Secondary FIFO current byte count CBC1[15..0]

GCRA6 Default : 00000000h Power Management Control

Bit	Type	Function
31	R/W	Individual module power line controlled by 0: PowerState 1: PWRDN0~5
30	R/W	PWRDN0 0: normal 1: power down internal FM synthesizer
29	R/W	PWRDN1 0: normal 1: power down FM D/A (DA2PWR)
28	R/W	PWRDN2 0: normal 1: power down SB D/A
27	R/W	PWRDN3 0: normal 1: power down MPU401 block
26	R/W	PWRDN4 0: normal 1: power down gameport block
25	R/W	PWRDN5 0: normal 1: power down analog block (PWR_d)
24:16		Reserved
15	R/W	Audio activity detection circuit power (APWR) 0: normal 1: power down
14	R/W	Line-in activity detection mask 0: mask 1: normal
13	R/W	CD-audio activity detection mask 0: mask 1: normal
12	R/W	MIC activity detection mask 0: mask 1: normal
11	R/W	Audio Inactivity Timer control 0: stop counting 1: normal
10:7		reserved
6	R/W	PME# routing to INTA# control 0: normal 1: enable PME# routing
5:4	R/W	Audio activity detection input gain 00 4X 01 6X 10 8X 11 10X
3:2	R/W	**Audio Inactivity Timer** duration 00 30 seconds 01 60 seconds 10 90 seconds 11 512 clocks (14.318 MHz) for testing
1:0		reserved

❶GCRA6.31 decides the way that the individual function module should be power down. If GCRA6.31=1, software must switch each block power line through PWRDN0~5 while power management state should be changed. If GCRA6.31=0, each block power line is determined by PowerState in PMCSR as described below:

D0 all the power line turn on
D2 cut power of FM synthesizer, FM D/A, SB D/A and gameport
$D3_{hot}$ all the power line turn off except that of audio activity detection and PCI configuration register

❷PCI clock can be stopped in $D3_{hot}$ state. External crystal clock is always on except in $D3_{cold}$ state..
❸Audio activity detection circuit should implement a counter which increases by 1 (with 14.318 MHz clock) if input analog level shift over a certain range (it is up to analog designer). A timer, has a period of 1ms, is needed to clear the counter. When the counter counts to a certain value (it is up to digital designer), a flag should be set to indicate that the audio is active.
❹If GCRA6.6=1, routing PME# to INTA#. That is, if PME# should be asserted, ALS4000 will generate an interrupt at the same time. Clear PME# will acknowledge this interrupt.

GCRA7 Default : 00000000h PIC access storage

Bit	Type	Function
31	R	S_ICWS: slave initialization command word select
30:24		reserved
23:19	R	S_IVBA: slave interrupt vector base address
18:16		reserved
15	R	M_ICWS: master initialization command word select
14:8		reserved
7:3	R	M_IVBA: master interrupt vector base address
2:0		reserved

❶Content of this register will remain even if system reset is asserted.

GCRA8 Default 00000000h Legacy function configuration 1

Bit	Type	Function
31:26		Reserved
25:19	R/W	GAMEBASE[9..3]
18:17		Reserved
16	R/W	Legacy Game-Port activate 0 : Disable 1 : Enable
15:10		Reserved
9:3	R/W	ADLIBBASE[9..3]
2:1		Reserved
0	R/W	Legacy ADLIB activate 0 : Disable 1 : Enable

GCRA9 Default 00000000h Legacy function configuration 2

Bit	Type	Function
31:26		Reserved
25:19	R/W	MPUBASE[9..3]
18:17		Reserved
16	R/W	Legacy MPU401 activate 0 : Disable 1 : Enable
15:10		Reserved
9:4	R/W	SBBASE[9..4]
3:1		Reserved
0	R/W	Legacy SB activate 0 : Disable 1 : Enable

❶SB IO length 16 bytes. If GCRA9.0=1, ALS4000 will claim legacy SB IO cycles. Note that if GCRA9.0=0, DMA emulation and DDMA will be disabled.
❷ADLIB IO length 4 bytes. If GCRA8.0=1, ALS4000 will claim legacy ADLIB IO cycles.
❸MPU401 IO length 2 bytes. If GCRA9.16=1, ALS4000 will claim legacy MPU401 IO cycles.
❹GAMEPORT IO length 2 bytes. If GCRA8.16=1, ALS4000 will claim legacy GAMEPORT IO cycles.
❺If legacy function disabled, ALS4000 can function only through the remapped IO accessing and will not claim legacy IO cycles.

GCRFF Default : 00000000h SCRATCH

Bit	Type	Function
31:0	R/W	dummy read/write for software switch

PCI Configuration Registers :

Vendor ID register :

Index : 00-01h Read only
Bit 15~8 40h (Vendor ID high byte)
Bit 7~0 05h (Vendor ID low byte)

Device ID register :

Index : 02-03h Read only
Bit 15~8 01h (Device ID high byte)
Bit 7~0 30h (Device ID low byte)

Command register :

Default 00h
Index : 04-05h R/W
Bit 2~0 0 Disable
1 Enable
Bit 15~3 reserved, read as 0
Bit 2 Bus Master enable
Bit 1 reserved, read as 0
Bit 0 IO access enable

Write PCI configuration register indexed 04h will reset ALS4000 bus master to avoid redundant action when warm boot (<CLT><ALT><DEL>) occurs.

Status register :

Index : 06-07h Read only
Default 0210h
Read :
Bit 15~11 Reserved ,read as 0
Bit 10,9 DEVSEL# Timing

Read as 01 (Medium Timing)
Bit 8~5 Reserved, read as 0
Bit 4 New capabilities indicator
Read as 1 (presence of new capability -- PCI power management)
Bit 3~0 Reserved, read as 0

Class code register :
Index : 09~0Bh Read only
Read as 040100h (Multimedia Audio Device)

Latency Timer Register :
Index: 0Dh Dummy R/W
Default : 00h
Bit 7~4 No. of PCI clock that ALS4000 retain ownership of bus.
Bit 3~0 Reserved , read as 0

Base Address Register 0 :
Index : 10~13h R/W
Default : 0001h
Bit 0 Read as 1 (IO base address indicator)
Bit 6~1 Reserved, read as 0
Bit 31~7 IOBASE bit 31~7 (128-bytes IO space)

Subsystem Vendor ID (SSVID) :
Index: 2C~2Dh Writable when SSWE=1.
Default : 4005h
Bit 15~0 Subsystem Vendor ID

Subsystem ID (SSID) :
Index: 2E~2Fh Writable when SSWE=1.
Default : 0130h
Bit 15~0 Subsystem ID

Capabilities Pointer (Cap_Ptr) :
Index : 34h Read only
Default : DCh
This register provides an offset into the Configuration Space for the location of the first item in Capabilities linked list.

Interrupt Line Register :
Index : 3Ch R/W
Default : 00h
Bit 7~0 : IRQ number

Interrupt Pin Register :
Index : 3Dh Read only
Default : 01h (INTA#)

Self-defined (SSWE) :
Index : D8h R/W
Default : 00h
Bit 7: SSVID and SSID write enable control (SSWE)
0 write disable
1 write enable
Bit 6~0: reserved

PCI Power Management register block :
Index : DC~E3h

Capability ID (Cap_ID) :
Index : DCh Read only
Default : 01h (identifies the presence of PCI power management)

Next Item Pointer (Next_Item_Ptr) :
Index : DDh Read only
Default : 00h (there are no additional items in the Capabilities list)

PMC (Power Management Capabilities) :
Index : DE~DFh Read only
Default : 6422h
Bit 15 : 0 **PME#** can not be asserted from $\mathbf{D3_{cold}}$
Bit 14 : 1 **PME#** can be asserted from $\mathbf{D3_{hot}}$
Bit 13 : 1 **PME#** can be asserted from **D2**

Bit 12 : 0 **PME#** can not be asserted from **D1**
Bit 11 : 0 **PME#** can not be asserted from **D0**
Bit 10 : 1 support **D2** power management state
Bit 9 : 0 not support **D1** power management state
Bit 8~6 : reserved, read as 0
Bit 5 : 1 Device Specific Initialization (DSI) required
Bit 4 : 0 auxiliary power source not required
Bit 3 : 0 PCI clock is not required for **PME#** operation
Bit 2~0 : 010 version number (Revision 1.0a of the *PCI Power Management Interface Specification*)

PMCSR (Power Management Control/Status) :

Index : E0~E1h
Default : 00h
Bit 15 (PME_Status) R/W
Write a "1" to this bit will clear it.
0 : Normal
1 : the function would assert **PME#**
Bit 14~13 (Data_Scale) Read only
Bit 12~9 (Data_Select) Read only
Bit 8 (PME_En) R/W
0 : **PME#** assertion is disabled
1 : **PME#** assertion is enabled
Bit 7~2 Read only
reserved
Bit 1~0 (PowerState) R/W
00 **D0**
01 reserved (not support **D1**)
10 **D2**
11 **$D3_{hot}$**

PMCSR_BSE (PMCSR PCI to PCI Bridge Support) :

Index : E2h Read only
Default : 00h

Data :

Index : E3h Read only
Default : 00h

Pay more attention to PME# circuit design. Note that when VCC is removed, PME# must be tri-state.

Appendix A : DMA Emulation

DMA emulation by BUS Master function :

ALS4000 implement 3 types of DMA emulation scheme for SBDMA only when legacy SB function is enabled (GCRA9.0=0): Legacy DMA type 1, Legacy DMA type 2 and Distributed DMA (DDMA). ALS4000 emulate the assigned channel (0,1,3) only.

It is not necessary to emulate the secondary DMA channel for recording or full duplex.

Internal Registers/Flags :

RETRY: The flag decide whether read status from 8237 or not.(Effective in Legacy-DMA)
- 1 Enabled read (Default)
- 0 Disable read

FFLP: 1_bit Flip-Flop function as low/high byte pointer for DMA IO register (00~07h) (Effective in Legacy-DMA)
Write 0x0c will clear FFLP to 0.FFLP is default 0 after reset.
Any access to 00h~07h will toggle it's value.

CA0: 24-bit address counter of DMA emulation for primary PCM FIFO

CBC0: 16-bit byte counter of DMA emulation for primary PCM FIFO

CA1: 24-bit address counter secondary PCM FIFO

CBC1: 16-bit byte counter secondary PCM FIFO

DMA emulation and data buffer operation design suggestion :

❶CBC0 decrease by 1 each time when 1-byte transfer to SB core is complete.

❷CA0 will decrease/increase by 1 each time when internal DRQx is activated.

❸Bus master write a DWORD to system memory in one memory write cycle if **CA1 equals to multiple times of 4 and CBC1 is greater than 3 and BL is greater than 3 bytes** while recording. In this case, CA1 will increase by 4 and CBC1 will decrease by 4 each time. Else, bus master will only write one byte to system memory each time.

❹Secondary PCM FIFO size is 32 bytes (4 bytes each layer, totle 8 layers). ALS4000 will request the bus when one layer (4 bytes) is filled. Burst transfer is not supported while recording.

❺There is a buffer in PCI block for burst mode playback, 32 layers, 4 bytes each layer. This buffer is used only when GCR99.2~0=100. When to assert REQ# and perform burst transfer is decided by threshold set (GCR99.20) and internal DRQx. This buffer acts as FIFO. When TC0 is reached or interrupt should be asserted or GCR99.21 is set, read/write pointer of the buffer will be reset, and address counter for bus master will load from CA0.

❻Bus master always assert byte enable signal with "0000" while playback even burst mode is disabled. Internal FIFO (primary PCM FIFO) will choice the useful byte according to CA0.

❼Primary internal PCM FIFO size is 16 bytes.

❽It is recommended to set SA, SSA, BBC, SBBC and BL equal to multiple times of 4.

DMA Emulation Scheme for two Legacy-DMA modes and DDMA mode :

- **Legacy-DMA Mode 1 (claim DMAC read cycle):**

ALS4000 decode the following IO command:

Address	Command	Function	Enabled
00h	IO write	FFLP=0: Write to GCR91.7~0 FFLP=1: Write to GCR91.15~8	GCRA9.0=1
00h	IO read	FFLP=0: read from CA0.7~0 FFLP=1: read from CA0.15~8	SBDMA=000 & GCRA9.0=1
01h	IO write	FFLP=0: Write to GCR92.7~0 FFLP=1: Write to GCR92.15~8	GCRA9.0=1
01h	IO read	FFLP=0: read from CBC0.7~0 FFLP=1: read from CBC0.15~8	SBDMA=000 & GCRA9.0=1
87h	IO write	write to GCR91.23~16	GCRA9.0=1
02h	IO write	FFLP=0: Write to GCR93.7~0 FFLP=1: Write to GCR93.15~8	GCRA9.0=1
02h	IO read	FFLP=0: read from CA0.7~0 FFLP=1: read from CA0.15~8	SBDMA=001 & GCRA9.0=1
03h	IO write	FFLP=0: Write to GCR94.7~0	GCRA9.0=1

		FFLP=1: Write to GCR94.15~8	
03h	IO read	FFLP=0: read from CBC0.7~0 FFLP=1: read from CBC0.15~8	SBDMA=001 & GCRA9.0=1
83h	IO write	write to GCR93.23~16	GCRA9.0=1
06h	IO write	FFLP=0: Write to GCR95.7~0 FFLP=1: Write to GCR95.15~8	GCRA9.0=1
06h	IO read	FFLP=0: read from CA0.7~0 FFLP=1: read from CA0.15~8	SBDMA=011 & GCRA9.0=1
07h	IO write	FFLP=0: Write to GCR96.7~0 FFLP=1: Write to GCR96.15~8	GCRA9.0=1
07h	IO read	FFLP=0: read from CBC0.7~0 FFLP=1: read from CBC0.15~8	SBDMA=011 & GCRA9.0=1
82h	IO write	write to GCR95.23~16	GCRA9.0=1
08h	IO write	Write bit 2 to GCR99.16	GCRA9.0=1
08h	IO read	Return the combined status	**$$$**
0Ah	IO write	Bit 1, Bit 0 0 0 Write bit 2 to GCR99.28 0 1 Write bit 2 to GCR99.29 1 0 No action 1 1 Write bit 2 to GCR99.31	GCRA9.0=1
0Bh	IO write	Bit 1, Bit 0 0 0 Write bit 5~2 to GCR92 bit 21~18 0 1 Write bit 5~2 to GCR94 bit 21~18 1 0 No action 1 1 Write bit 5~2 to GCR96 bit 21~18	GCRA9.0=1
0Ch	IO write	Clear FFLP to 0	GCRA9.0=1
0Dh	IO write	Clear FFLP,DMA group enable(GCR99.16), DMA status and set mask bit (GCR99.31,29,28)	GCRA9.0=1
0Eh	IO write	Clear mask bit (GCR99.31,29,28)	GCRA9.0=1
0Fh	IO write	Write all mask bit(Write bit 3,1,0 to GCR99.31,29,28)	GCRA9.0=1

For DMA register access, ALS4000 take different action for read and write operation :

Write Operation "Snoop" it, fetch data and save to corresponding registers.

Read Operation Controlled by GCR99.2~0. When Legacy-DMA read is enabled, ALS4000 will claim the read command.
For DMA status (08h) read operation, ALS4000 will take action depending on RETRY. If RETRY=1, ALS4000 issue PCI retry to terminate the cycle and request the bus. When granting the bus, ALS4000 issue 8237 status read cycle to get the status. After getting the status, ALS4000 combined it with current status of the emulated DMA channel and clear RETRY.
The constructed data is :

SBDMA =000 Replace 8237-Status bit 4,0 with internal status
SBDMA =001 Replace 8237-Status bit 5,1 with internal status
SBDMA =011 Replace 8237-Status bit 7,3 with internal status

If RETRY=0, ALS4000 return the status. After status is read, RETRY is set again.
For current address/byte counter (00h,01h,02h,03h,06h,07h) read, ALS4000 claim the command for the DMA channel emulated only.
For I/O read from 87h/82h/83h, it's not necessary to claim these IO command.

- **Legacy-DMA Mode 2 (write back DMAC current address and byte counter register):**

For DMAC register write operation, ALS4000 acts just the same as Legacy-DMA mode 1, snoops and fetches data and stores in the corresponding register.
ALS4000 will always ignore DMAC register read cycle.

If bus master for DMAC register update is enabled, ALS4000 should perform four IO write cycles each time after one byte has been read from memory, as follow:

	Address	**Data**

1st~2nd	00h (if SBDMA=0) 02h (if SBDMA=1) 06h (if SBDMA=3)	CA0.7~0 (if FFLP=0) CA0.15~8 (if FFLP=1)
3rd~4th	01h (if SBDMA=0) 03h (if SBDMA=1) 07h (if SBDMA=3)	CBC0.7~0 (if FFLP=0) CBC0.15~8 (if FFLP=1)

Note: ❶REQ# will keep low unless all the five transactions are completed or terminated by retry.
❷After each write cycle has completed, toggle FFLP's value.

- **DDMA Mode :**

ALS4000 implemented DDMA register for system chipset access. One channel for primary PCM FIFO only.

The DDMA address group for SB type DMA is calculated as follow :

DDMABASE[31..16]=0
DDMABASE[15..4]=GCR99.15~4

DDMABASE[3..0] map to the following operation.

DDMABASE[3..0]	Type	Function
00h	W	SBDMA=000 Write to GCR91.7~0 SBDMA=001 Write to GCR93.7~0 SBDMA=011 Write to GCR95.7~0
00h	R	Read from CA0.7~0
01h	W	SBDMA=000 Write to GCR91.15~8 SBDMA=001 Write to GCR93.15~8 SBDMA=011 Write to GCR95.15~8
01h	R	Read from CA0.15~8
02h	R/W	SBDMA=000 Access GCR91.23~16 SBDMA=001 Access GCR93.23~16 SBDMA=011 Access GCR95.23~16
03h		Reserved
04h	W	SBDMA=000 Write to GCR92.7~0 SBDMA=001 Write to GCR94.7~0 SBDMA=011 Write to GCR96.7~0
04h	R	Read from CBC0.7~0
05h	W	SBDMA=000 Write to GCR92.15~8 SBDMA=001 Write to GCR94.15~8 SBDMA=011 Write to GCR96.15~8
05h	R	Read from CBC0.15~8
06h~07h		Reserved
08h	W	Write bit 2 to GCR99.16
08h	R	Read internal DMA status
09h	W	Terminate the cycle only
0Ah		Reserved
0Bh	W	SBDMA=000 Write bit5~2 to GCR92.21~18 SBDMA=001 Write bit5~2 to GCR94.21~18 SBDMA=011 Write bit5~2 to GCR96.21~18
0Ch		Reserved
0Dh	W	Clear FFLP,DMA group enable(GCR99.16),DMA status and set mask bit (GCR99.31,29,28)
0E		Reserved
0F	W	SBDMA=000 Write bit 0 to GCR99.28 SBDMA=001 Write bit 0 to GCR99.29 SBDMA=011 Write bit 0 to GCR99.31

For DMA/software compatibility issue, CA0/CBC0 load starting value according the condition as follow in spite of legacy SB function is enabled or not:

Source	Enabled condition
GCR91 GCR92	(SBDMA=0)&[(write GCR99.26~24) | (IO write GCR91,GCR92.15~0) | (TC0=1 in auto-initialization mode)]
GCR93 GCR94	(SBDMA=1)&[(write GCR99.26~24) | (IO write GCR93,GCR94.15~0) | (TC0=1 in auto-initialization mode)]
GCR95 GCR96	(SBDMA=3)&[(write GCR99.26~24) | (IO write GCR95,GCR96.15~0) | (TC0=1 in auto-initialization mode)]

Note : TC0=1 when CBC0 reach FFFFh.

PCI playback method when legacy SB function disabled:

If legacy SB function is disabled, DMA emulation is disabled too. Software can setup starting address and byte counter for primary PCM FIFO bus master by accessing GCR91~GCR96 and get status through GCRA0~GCRA1. It is recommanded that software should set SA, BBC and BL equal to multiple times of 4 and enable burst mode to improve performance.

Secondary PCM FIFO bus master control:

In spite of that legacy SB function is enabled or disabled, software can setup starting address, byte counter and work mode for secondary PCM FIFO bus master and get status through GCRA2~GCRA5.

For software compatibility issue, CA1/CBC1 load starting value according the condition as follow :

Source: GCRA2/GCRA3
Condition: (IO write to GCRA2, GCRA3.15~0) | (TC1=1)
Note: TC1=1 when CBC1 reach FFFFh.

Note: When PWRDN_ is low, PCI clock can be stopped. All PCI output drivers must be disabled. It is suggested that block power switch control signal should be combined with PWRDN_, except analog power switch control (PWR_d). Audio activity detection power (APWR) should be cut too. When PWRDN_ pin goes from low to high, an internal reset signal should be generated. In any case, crystal oscillator (14.318MHz) will always work.

Attention for test mode: If set GCR90.14=1, and set sample rate=48KHz, internal PCM FIFO will run at a fixed rate=48*8 (KHz) to accelerate procedure.