ESS Technology, Inc.

ES1878
*Audio*Drive®
Data Sheet

GENERAL DESCRIPTION

The ES1878 *Audio*Drive® is a mixed-signal single-chip solution that adds 16-bit stereo sound and FM music synthesis to notebook computers. The ES1878 includes an embedded microprocessor, a 20-voice ESFM™ music synthesizer, 16-bit stereo ADC and DACs, optional ISA Plug and Play (PnP) support, hardware master volume control, DMA control logic with FIFO, ISA bus interface logic, general-purpose I/O, and digital dual game port. The ES1878 also incorporates three serial ports which allow interfacing with external DSP, wavetable, and MIDI (MPU-401 UART Mode compatible). In addition, the ES1878 offers I²S Zoom Video interface and support for the optional ES978 Expansion Audio Mixer in docking stations.

A 4-wire expansion analog bus and 2-wire serial control bus connect the ES1878 with the ES978, allowing the ES1878 to engage the docking station's audio resources when docked.

The ES1878 *Audio*Drive® can record, compress, and play back voice, sound, and music with built-in mixer controls. Using two high-performance DMA channels, the ES1878 supports full-duplex operation for simultaneous record and playback. The ESFM™ synthesizer has extended capabilities within native mode operation providing superior sound and power down capabilities. The ES1878 is register compatible with Yamaha's OPL3™ FM synthesizer.

The ES1878 *Audio*Drive® supports ISA Plug and Play with configuration for 4 logical devices: configuration device, audio plus ESFM™, game port, and MPU-401. The ES1878 also supports an optional non-PnP (BIOS Plug and Play) configuration for the Intel® Mobile Triton chipset in which the PCA-ISA bridge features Subtractive Decode and Positive Decode Modes.

The MPU-401 serial port is for interfacing with an external MIDI device. The digital dual game port supports two joysticks with hardware timing which uses less CPU overhead and improves system performance.

A DSP serial interface in the ES1878 allows an external DSP to take over ADC or DAC resources. A wavetable serial port allows the ES1878 to interface with either the ES689 or ES690 wavetables. I/O address, DMA, and interrupt selection can be controlled through system software or Plug and Play.

Advanced Power Management (APM) features include suspend and resume from disk.

The ES1878 is available in an industry-standard 100-pin Small Quad Flat Pack (SQFP) package.

FEATURE HIGHLIGHTS

- Single, mixed-signal, 16-bit stereo VLSI chip for digital audio
- High-quality, 20-voice ESFM™ music synthesizer; patents pending
- Supports ES978 Expansion Audio Mixer chip
- Full Plug and Play (PnP) capability

Record and Playback Features

- Record, compress, and play back voice, sound, and music
- 16-bit stereo CODEC
- 2 additional DACs for digital audio, music synthesis, and I²S Zoom Video
- Programmable sample rates from 4 kHz to 44.1 kHz for record and playback
- Patented ESPCM® compression
- Full-duplex operation for simultaneous record and playback
- 6-bit (64 step) software volume control
- 2-button or 3-button hardware master volume control for up, down, and mute

Inputs/Outputs

- Stereo inputs for LINE, AUXA and AUXB, and a mono input for MIC
- MIDI serial port compatible with MPU-401 UART Mode
- Serial port interface for external DSP, which optionally controls full-duplex operation
- Supports up to 7 general-purpose inputs (GPIs) and 7 general-purpose outputs (GPOs) that can be slaved with corresponding pins of the Expansion Audio Mixer
- ESS high-performance digital dual game port with hardware timing
- High-performance DMA supports demand transfer and F-type
- Software address mapping, DMA, and IRQ selections for motherboard implementation

- I²S Zoom Video port interface for MPEG audio with sample rates up to 48 kHz
- Wavetable serial port interface for ES689/ES690 to access the music DAC
- PC speaker input/output with volume control
- DSP serial interface

Interfaces to Expansion Audio Mixer (ES978)

- Simple hot-docking interface for ES978 Expansion Audio Mixer
- Two-wire digital status and data communication between ES1878 and ES978 supports register shadowing with worst case latency of approximately 140 microseconds
- Two pairs of on-chip analog differential signals for audio I/O with ES978 Expansion Audio Mixer

Mixer Features

- 6-channel stereo mixer inputs for line, auxiliary A (CD audio), auxiliary B, digital audio (wave files), music synthesizer, I²S Zoom Video port, plus a mono channel mixer input for a microphone
- Programmable 6-bit logarithmic master volume control

Plug and Play (PnP) Features

- On-chip PnP support for audio, joystick port, FM, and MPU-401
- Software address mapping, four DMA and six IRQ selections for motherboard implementation

Power

- Advanced Power Management supports suspend/ resume from disk
- Supports 3.3 or 5.0 V operation

Compatibility

- Supports PC games in Sound Blaster™ and Sound Blaster™ Pro modes
- Microsoft® Windows™ Sound System®

Operating Systems

- Microsoft Windows®95
- Microsoft Windows™ 3.1
- Microsoft Windows for Workgroups™
- Microsoft Windows NT™ 3.51 & 4.0
- IBM® OS/2® Warp™

COPYRIGHTS AND TRADEMARKS

Copyright Notice

Disclaimer

Information supplied by ESS Technology is believed to be accurate and reliable. However, ESS Technology assumes no responsibility for use of this information, nor for any errors contained herein.

Products in this document may be covered by U.S. Patent 4,214,125 and others; other patents pending.

Trademarks

*Audio***Drive**® and **ESPCM**® are registered trademarks of ESS Technology, Inc.

ESFM™ is a trademark of ESS Technology, Inc.

All other brand names and product names mentioned in this document are trademarks or registered trademarks of their respective owners and are used for identification purposes only.

Revision History

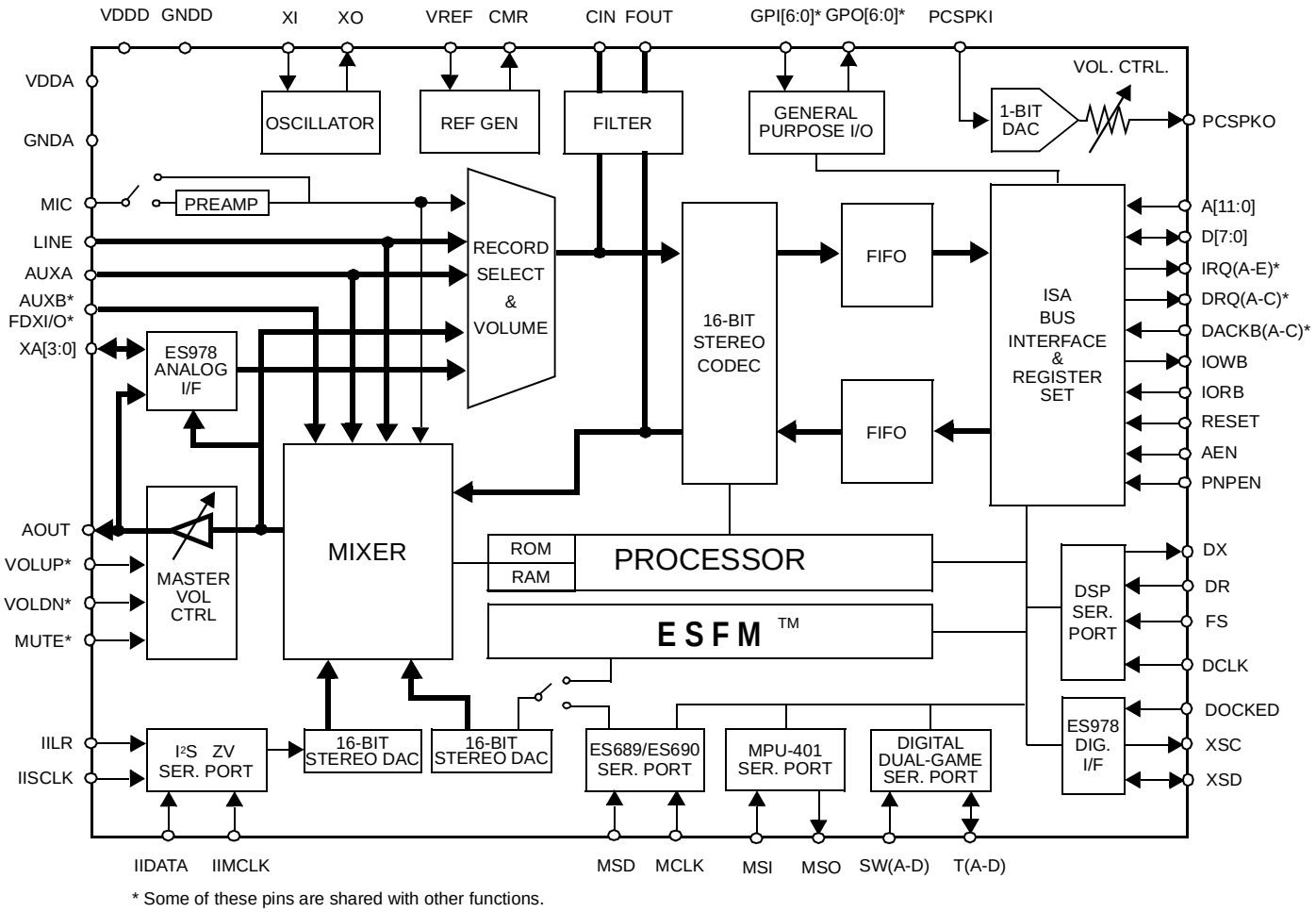
This ES1878 *Audio***Drive**® Data Sheet replaces the ES1878 *Audio***Drive**® Preliminary Design Guide/Data Sheet.

CONTENTS

FIGURES

TABLES

FUNCTIONAL BLOCK DIAGRAM



* Some of these pins are shared with other functions.

Figure 1 ES1878 Functional Block Diagram

MIXER SCHEMATIC BLOCK DIAGRAM

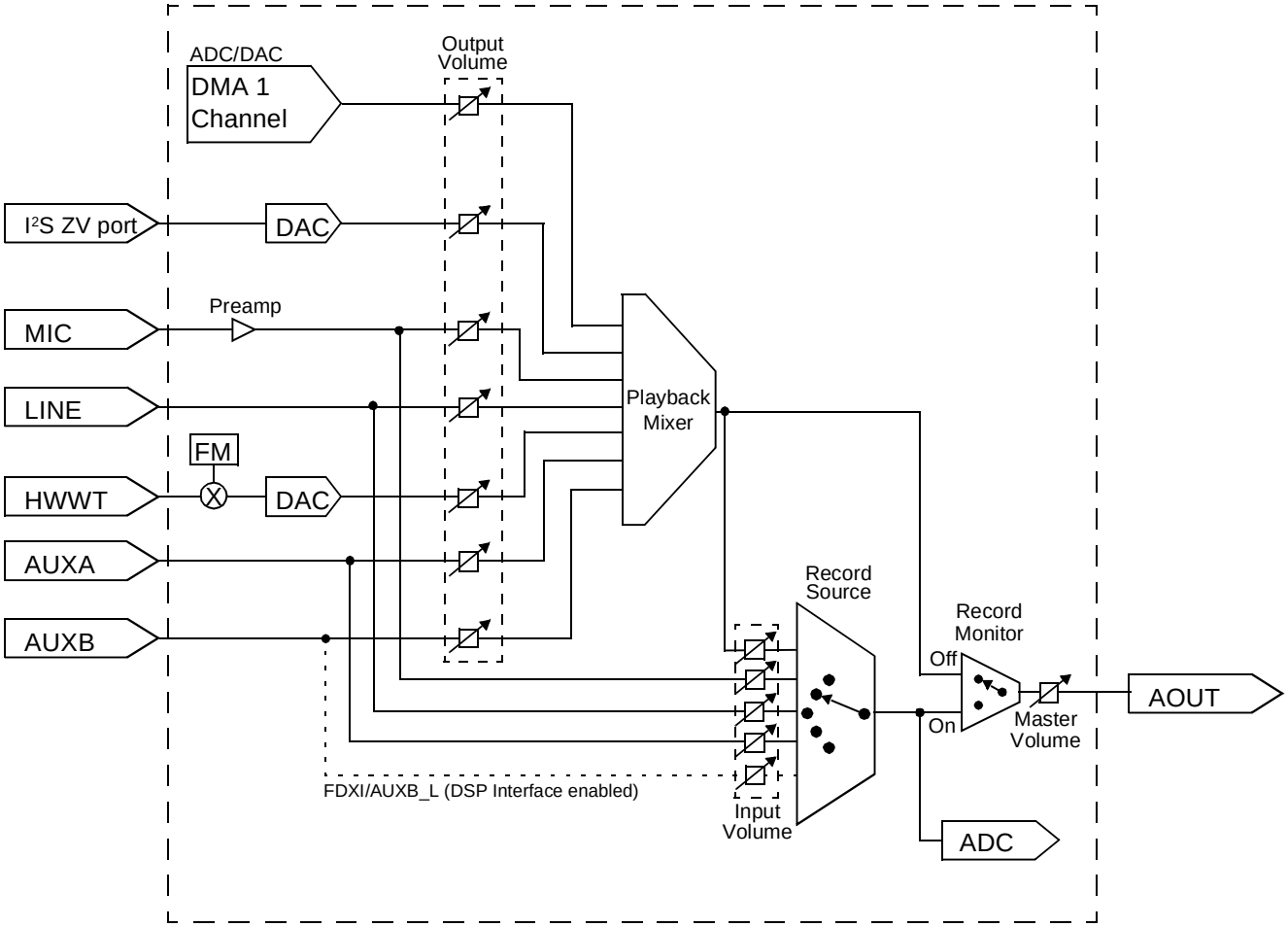


Figure 2 ES1878 Mixer Schematic Block Diagram

Subsystems Description

This section discusses the major subsystems shown in Figure 1.

- **RISC processor** – game-compatible audio functions are performed by an embedded microprocessor.
- **Oscillator** – circuitry to support an external crystal.
- **ROM and RAM** – firmware ROM and data RAM to the embedded processor.
- **FIFO** – RAM for a 256-byte FIFO data buffer.
- **ISA bus interface** – provides interface to ISA bus address, data, and control signals.
- **Digital dual game ports** – ESS high-performance digital switches for two joysticks with hardware timing.
- **ES978 interface** – allows support for Expansion Audio Mixer in the docking station using two-wire control signals and four-wire analog bus.
- **I²S Zoom Video interface** – supports sample rates up to 48 kHz for MPEG audio.
- **MPU-401 serial port** – asynchronous serial port for MIDI devices such as a wavetable synthesizer or a music keyboard input.
- **Wavetable serial port** – serial port connection from the output of an ES689 or ES690 that eliminates the requirements for an external DAC.
- **DSP serial port** – interface for an optional external DSP for control of the CODEC.
- **ESFM™ music synthesizer** – high-quality 20-voice FM synthesizer.
- **Stereo programmable mixer** – seven input stereo mixer. Each input is independent left and right 4-bit.
  - Line In
  - Mic In
  - Aux A (CD)
  - Aux B (or FDXI)
  - Digitized audio (wave files)
  - FM/ES689/ES690 music DAC
  - I²S serial port DAC
- **16-Bit Stereo I²S DAC** – I²S Zoom Video for MPEG audio DAC.
- **16-Bit stereo CODEC** – for audio record and playback CODEC.
- **16-Bit stereo music DAC** – for ESFM™ or external wavetable synthesizer.
- **1-Bit DAC** – for PC speaker digital input.
- **Recording source and input volume control** – input source and volume control for record. The recording source can be selected from one of five choices:
  - Mic
  - Line
  - Aux A (CD)
  - Mixer
  - FDXI pin (mono full-duplex mode, DSP interface enabled)

  In any of the first four cases, the selected recording source may be mixed with audio from the ES978 if the selected source is also enabled in the ES978 and the two chips are docked.
- **Output volume and mute control** – The master volume is controlled by either programmed I/O or volume control switch inputs. The master volume supports 6 bits per channel plus mute. When docked, the ES1878 first transmits the master volume information to the ES978 mixer before it can take effect.

  The 3-pushbutton inputs with internal pull-up devices for up/down/mute can be used to adjust the master volume control.

  The state of these pins is logically AND'ed with the state of the corresponding pins of the ES978 when docked. A software-selectable option allows the mute input to be omitted. The mute input is defined as the state when both up and down inputs are low. By default, this feature is disabled.

  The hardware volume inputs of the ES1878 can be used as general-purpose inputs (see bits 4 and 5 of Vendor-Defined Card-Level register 25h). They cannot be used as volume control inputs.
- **Reference generator** – analog reference voltage generator.
- **PC speaker volume control** – The PC speaker is supported with a 1-bit DAC with volume control. The analog output pin PCSPKO is intended to be externally mixed at the external amplifier. PC speaker audio is not transmitted to the ES978 through the expansion audio interface (XA[3:0]) but is always heard through the portable unit's speakers.

PINOUT DIAGRAM

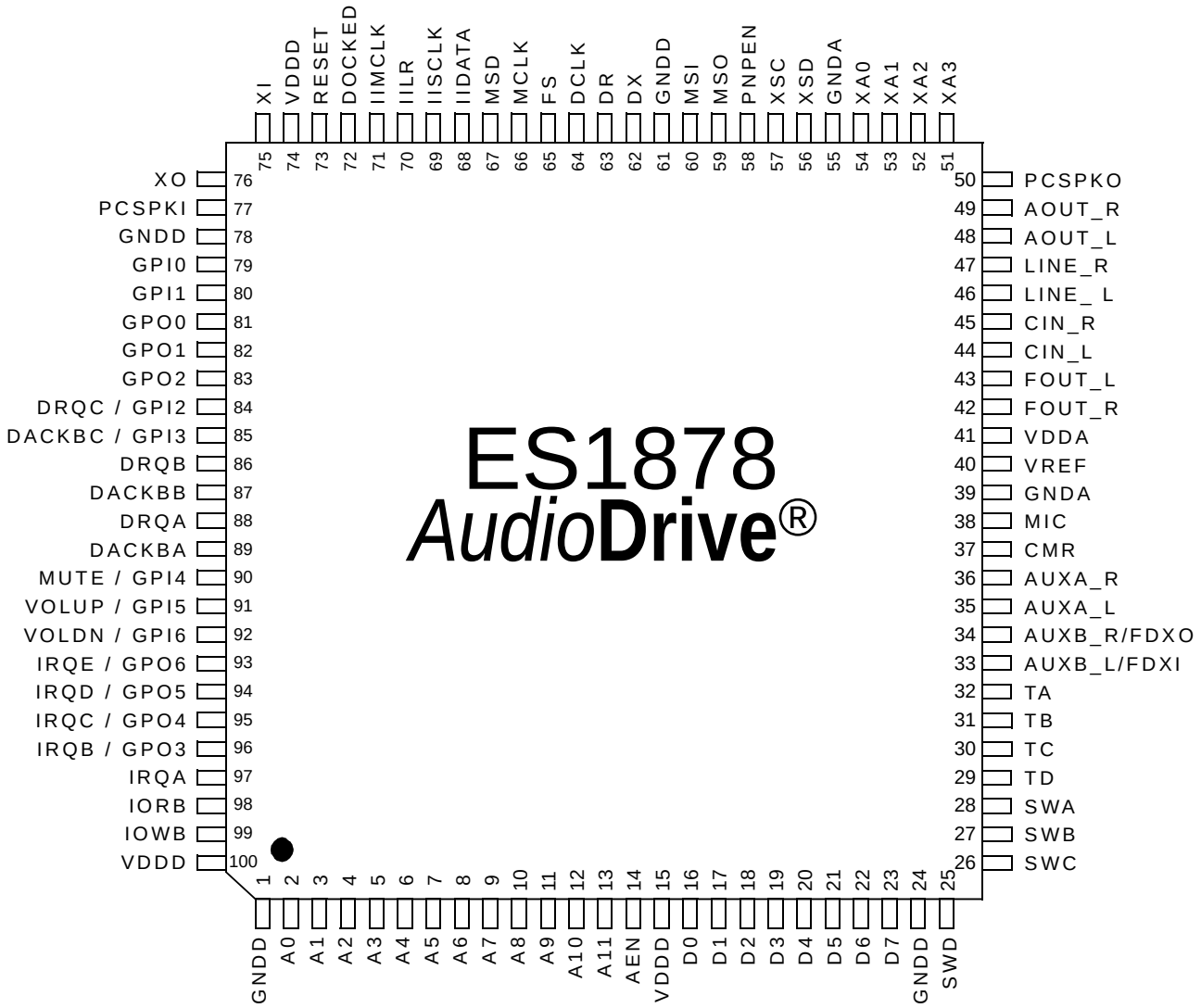


Figure 3 ES1878 Pinout

PIN DESCRIPTIONS

Table 1 Analog Pins

Name	Number	I/O	Description
Power and Ground			
VDDA	41	I (Pwr)	Analog power supply, 4.75 V - 5.25 V.
GNDA	39, 55	I (Pwr)	Analog ground.
Recording Source and Input Volume Control			
LINE_L	46	I	Line input left. 70k ohm pull-up to the CMR pin.
LINE_R	47	I	Line input right. 70k ohm pull-up to CMR.
AUXA_L	35	I	Aux A (CD) input left. 70k ohm pull-up to CMR.
AUXA_R	36	I	Aux A (CD) input right. 70k ohm pull-up to CMR.
AUXB_L	33	I	Multipurpose pin, AUXB_L (Aux B input left) or FDXI. 70k ohm pull-up to CMR.
AUXB_R	34	I	Multipurpose pin, AUXB_R (Aux B input left) or FDXO. 70k ohm pull-up to CMR.
MIC	38	I	Mic input to +26 dB internal preamp. 70k ohm pull-up to CMR.
Output Volume Control and Record Monitor			
AOUT_L	48	O	Analog output left from master volume. This pin can drive a 10k ohm load.
AOUT_R	49	O	Analog output right from master volume. This pin can drive a 10k ohm load.
PCSPKO	50	O	PC speaker analog output.
ES978 Analog Interface			
XA[3:0]	51:54	I/O	Bidirectional differential transmitter/receivers. Expansion audio bus. These are analog signals that are DC-coupled to the corresponding XA[3:0] pins of the ES978.
DSP Interface			
FDXI *	33	I	Multipurpose pin, AUXB_L or FDXI. When used as FDXI with DSP interface, provides a line-level mono input.
FDXO *	34	O	Multipurpose pin, AUXB_R or FDXO. When used as FDXO with DSP interface, is a line-level mono output, capable of driving a 5k ohm load.
Miscellaneous Analog Pins			
CIN_L	44	I	Capacitive coupled input left. The input resistance is about 50k ohms.
CIN_R	45	I	Capacitive coupled input right. The input resistance is about 50k ohms.
FOUT_L	43	O	Filter output left. This pin is normally AC-coupled to CIN_L. The output resistance is about 5k ohms.
FOUT_R	42	O	Filter output right. This pin is normally AC-coupled to CIN_R. The output resistance is about 5k ohms.
CMR	37	I/O	2.25 V buffered common mode reference output.
VREF	40	I/O	2.25 V reference generator.

* These pins are shared with other functions.

Table 2 Digital Pins

Name	Number	I/O	Description
Power and Ground			
VDDD	15, 74, 100	I (Pwr)	Digital power supply (3.0 V - 5.5 V).
GNDD	1, 24, 61, 78	I (Pwr)	Digital ground.
ISA Bus Interface			
A[11:0]	13:2	I	ISA address bus.
AEN	14	I	ISA address valid when active-low, DMA when high.
D[7:0]	23:16	I/O	ISA data bus. 24 mA drivers.
IOWB	99	O	ISA active-low write strobe.
IORB	98	I	ISA active-low read strobe.
IRQA	97	O/Hi Z	ISA interrupt request. 16 mA driver. Connect to IRQ9 if PNPEN = 1.
IRQB *	96	O/Hi Z	Multipurpose pin, IRQB or GPO3. Connect to IRQ5 if PNPEN = 1.
IRQC *	95	O/Hi Z	Multipurpose pin, IRQC or GPO4. Connect to IRQ7 if PNPEN = 1.
IRQD *	94	O/Hi Z	Multipurpose pin, IRQD or GPO5. Connect to IRQ10 if PNPEN = 1.
IRQE *	93	O/Hi Z	Multipurpose pin, IRQE or GPO6. Connect to IRQ11 if PNPEN = 1.
DACKBA	89	I	ISA active-low DMA acknowledge. Connect to DACK0 if PNPEN = 1.
DRQA	88	O/Hi Z	ISA active-high DMA request. Connect to DRQ0 if PNPEN = 1.
DACKBB	87	I	ISA active-low DMA acknowledge. Connect to DACK1 if PNPEN = 1.
DRQB	86	O/Hi Z	ISA active-high DMA request. Connect to DRQ1 if PNPEN = 1.
DACKBC *	85	I	Multipurpose pin, DACKBC or GPI3. If used as DACKBC, is ISA active-low DMA acknowledge. Connect to DACK3 if PNPEN = 1.
DRQC *	84	I/O/ Hi Z	Multipurpose pin. DRQC or GPI2. If used as DRQC, is ISA active-high DMA request. Connect to DRQ3 if PNPEN = 1.
RESET	73	I	ISA active-high reset.
ES978 Digital Interface			
DOCKED	72	I	Input that is active-high when the ES1878 is docked to the ES978. This pin has an internal pull-down to GNDD.
XSC	57	O	Expansion serial bus clock and frame sync. High-impedance when DOCKED = 0.
XSD	56	I/O	Expansion serial bus data I/O. High-impedance when DOCKED = 0.
MPU-401 Serial Port (MIDI) and FIFOs			
MSO	59	O	MIDI serial output.
MSI	60	I	MIDI serial input. MSI has an internal pull-up to VDDD.
Dual Game Port Joystick			
SW(A-D)	28:25	I	Joystick switch inputs. These pins have internal pull-ups to VDDD.
T(A-D)	32:29	I/O	Joystick timers. Use the digital power supply. These pins have internally weak pull-downs to GNDD (> 1M ohms).
I^2S Interface			
IISCLK	69	I	Serial shift clock for I^2S interface. This pin has an internal pull-down to GNDD.
IIDATA	68	I	Serial data for I^2S interface. This pin has an internal pull-down to GNDD.
IILR	70	I	Left/right signal for I^2S interface. This pin has an internal pull-down to GNDD.
IIMCLK	71	I	Reserved for future oversampling clock for I^2S interface (software selectable to set x256, x384 or x512 sample rate). This pin has an internal pull-down to GNDD. The current design using ES1878 does not require IIMCLK.

Table 2 Digital Pins (Continued)

Name	Number	I/O	Description
ES689/ES690 Interface			
MCLK	66	I	Serial clock input from ES689/ES690.
MSD	67	I	Serial data input from ES689/ES690. This pin has an internal pull-down to GNDD.
DSP/CODEC Interface			
FS	65	I	Frame sync input. Software programmable to be active-high or active-low. This pin has an internal pull-down to GNDD.
DCLK	64	I	Serial clock input. This pin has an internal pull-down to GNDD.
DR	63	I	Serial data receive. This pin has an internal pull-down to GNDD.
DX	62	O/Hi Z	Serial data transmit.
External Hardware Volume Control			
MUTE *	90	I	Multipurpose pin, MUTE or GPI4. When used as MUTE, is an active-low mute input. This pin has an internal pull-up tp VDDD.
VOLDN *	92	I	Multipurpose pin, VOLDN or GPI6. When used as VOLDN, is an active-low volume-down input. This pin has an internal pull-up to VDDD.
VOLUP *	91	I	Multipurpose pin, VOLUP or GPI5. When used as VOLUP, is an active-low volume-up input. This pin has an internal pull-up to VDDD.
General-Purpose/PnP I/O Interface			
GPI6 *	92	I	Multipurpose pin, VOLDN or GPI6. If used as GPI6, is general-purpose input 6.
GPI5 *	91	I	Multipurpose pin, VOLUP or GPI5. If used as GPI5, is general-purpose input 5.
GPI4 *	90	I	Multipurpose pin, MUTE or GPI4. If used as GPI4, is general-purpose input 4.
GPI3 *	85	I	Multipurpose pin, DACKBC or PGI3. If used as GPI3, is general-purpose input 3.
GPI2 *	84	I/O/ Hi Z	Multipurpose pin, DRQC or GPI2. If used as GPI2, is general-purpose input 2.
GPI [1:0]	80:79	I	General-purpose inputs 1:0.
GPO6 *	93	O	Multipurpose pin, IRQE or GPO6. If used as GPO6, is general-purpose output 6.
GPO5 *	94	O	Multipurpose pin, IRQD or GPO5. If used as GPO5, is general-purpose output 5.
GPO4 *	95	O	Multipurpose pin, IRQC or GPO4. If used as GPO4, is general-purpose output 4.
GPO3 *	96	O	Multipurpose pin, IRQB or GPO3. If used as GPO3, is general-purpose output 3.
GPO[2:0]	83:81	O	General-purpose outputs 2:0.
Miscellaneous Digital Pins			
XI	75	I	14.31818 MHz clock input, or external crystal.
XO	76	O	Output to external 14.31818 MHz crystal.
PNPEN	58	I	PnP enable. When high, Plug and Play is enabled; when low, Plug and Play is disabled.
PCSPKI	77	I	PC speaker digital input. This pin has an internal pull-down.

* These pins are shared with other functions.

TYPICAL APPLICATION

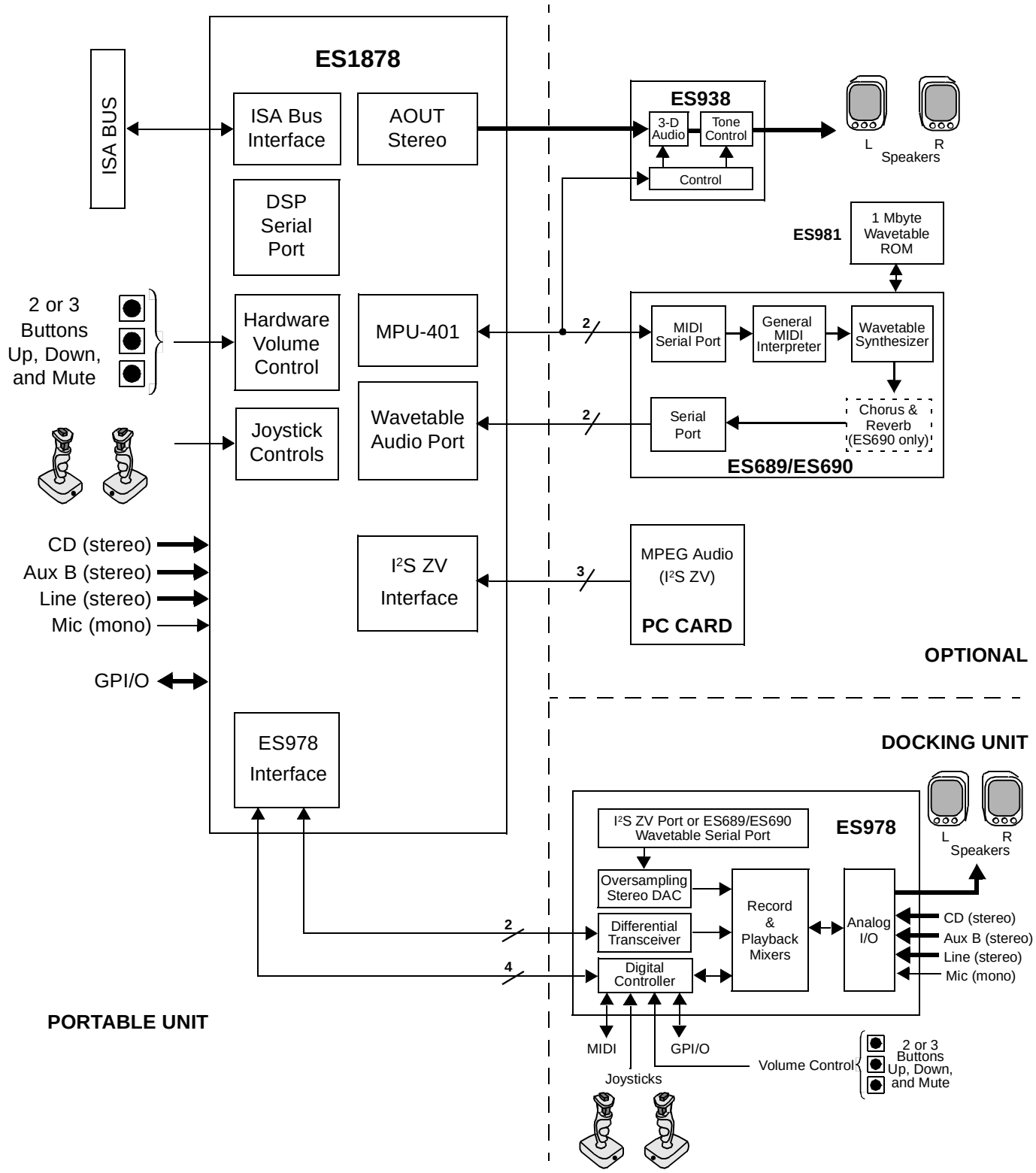


Figure 4 ES1878 Typical Application

ANALOG HARDWARE INTERFACE

Reference Generator

Reference generator pins VREF and CMR are connected through bypass capacitors to analog ground.

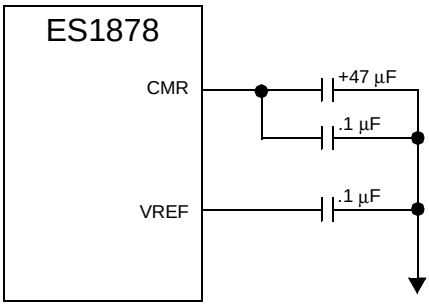


Figure 5 Reference Generator Pin Diagram

Switch-Capacitor Filter

The outputs of the FOUT_L and FOUT_R filters must be AC-coupled to the inputs CIN_L and CIN_R respectively, which provides for DC blocking and an opportunity for low-pass filtering with capacitors to analog ground at these inputs.

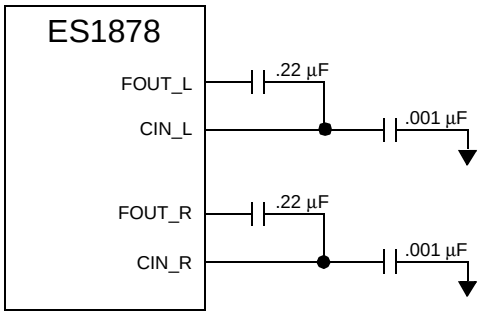


Figure 6 Switch-Capacitor Filter Pin Diagram

Audio Inputs and Outputs

Analog inputs MIC, stereo LINE, stereo AUXA, and stereo AUXB are to be capacitively coupled to their respective input signals. All have pull-up resistors to CMR.

ES1878 analog outputs AOUT_L and AOUT_R are intended to be AC-coupled to an amplifier, volume control potentiometer, or line-level outputs.

DIGITAL HARDWARE INTERFACE

ISA Bus Interface

Table 3 shows the pins used to interface the ES1878 with the ISA bus.

Table 3 ES1878 ISA Bus Interface

Pin	I/O	Description
A[11:0]	I	ISA address bus.
AEN	I	ISA address valid when active low, DMA when high.
D[7:0]	I/O	ISA data bus 24 mA drivers.
IOWB	O	ISA active-low write strobe.
IORB	I	ISA active-low read strobe.
IRQ(A-E)	O/Hi Z	ISA interrupt request. 16 mA driver.
DACKB(A-C)	I	ISA active-low DMA acknowledge.
DRQ(A-C)	O/Hi Z	ISA active-high DMA request.
RESET	I	ISA active-high reset.

DMA Interface

The ES1878 incorporates two DMA channels. There are two sources of DMA requests and two targets for DMA acknowledge:

Audio 1 The first audio DMA channel. This channel is used for Sound Blaster-compatible DMA, and Extended Mode DMA. It can be used for either record or playback. Ideally, this DMA channel should be assigned to ISA DMA channel 1.

Audio 2 The second audio DMA channel. This channel is used for audio playback in full-duplex mode. This channel can be mapped to any of the three 8-bit ISA DMA channels: 0, 1, or 3.

The two DMA sources are mapped to the three DMA pin pairs through PnP registers. Also, the three DMA pin pairs are assigned ISA DMA channel numbers by Vendor-Defined Card-Level registers 23h and 24h.

In order for a DRQ output pin to be *driving* (as opposed to *high-impedance*), two things must occur: 1) The PnP register for the DMA of a given device must match the ISA DMA channel number of the pin and 2) the given device must be activated (that is, bit 0 of PnP register 30h must be high).

For detailed information, see the 'Configuration Description' section.

DRQ Latch Feature

The DRQ latch feature is enabled when bit 7 of Vendor-Defined Card-Level register 29h is high (see Figure 7).

If this feature is enabled, each of the three audio DRQs will be latched high until one of the following occurs:

- A DACK low pulse occurs while DRQ is low or if DRQ goes low due to a DACK pulse.
- A hardware reset occurs.
- 8-16 milliseconds elapse while DRQ is low.

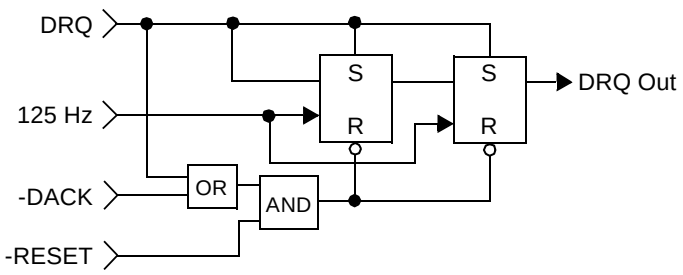


Figure 7 DRQ Latch

Interrupts

Interrupt Sources

Interrupt sources are mapped to any one of the five interrupt output pins via the PnP registers. A given pin can have zero, one, or more interrupts mapped to it. Each PnP pin is assigned to an ISA interrupt channel number by Vendor-Defined Card-Level PnP registers 20h, 21h, and 22h. These registers are automatically loaded from the 8-byte header in the PnP configuration data.

If a given interrupt pin has one or more sources assigned to it, and one or more of those sources is activated (register 30h, bit 0), then the interrupt pin will be active; that is, it will always be driving high or low. Each interrupt also has one or more mask bits that are AND'ed with the interrupt request.

There are four interrupt sources in the ES1878:

- audio 1
- audio 2
- hardware volume
- MPU-401

Audio 1

This interrupt is used for the first DMA channel (Sound Blaster-compatible DMA, Extended Mode DMA, and Extended Mode programmed I/O), as well as Sound Blaster-compatible MIDI receive. Extended register B1h controls use of this interrupt for Extended Mode DMA and programmed I/O. This interrupt request is cleared by a hardware or software reset, or by an I/O read from port Audio_Base+Eh. The interrupt request can be polled by reading from port Audio_Base+Ch. This interrupt is assigned to an interrupt channel by PnP register 70h of LDN #1.

Audio 2

Optional for the second DMA channel. The ES1878 can operate in full-duplex mode using two DMA channels. However, the second DMA channel must have the same sample rate as the first DMA channel. For this reason, it is not necessary to use a separate interrupt for the second DMA channel. This interrupt is masked by bit 6 of Mixer Extension register 7Ah. It can be polled and cleared by reading or writing bit 7 of the same register. This interrupt is assigned to an interrupt channel by PnP register 72h of LDN #1.

Hardware Volume

Hardware volume activity interrupt. This interrupt occurs when one of the three hardware volume controls generates a change in status (an event). Bit 1 of Mixer Extension register 64h is the mask bit for this interrupt. The interrupt request can be polled by reading bit 3 of the same register. The interrupt request is cleared by writing any value to register 66h. This interrupt is assigned to an interrupt channel by PnP register 28h. Typically this interrupt, if used, is shared with an audio interrupt.

MPU-401

This interrupt is generated when a MIDI byte is received. It goes low when a byte is read from the MIDI FIFO and goes high again quickly if there are additional bytes in the FIFO. The interrupt status is the same as the Read-Data-Available status flag in the MPU-401 Status register. This interrupt is masked by bit 6 of Mixer Extension register 64h and is assigned to an interrupt channel in one of two ways: If the MPU-401 is part of the audio device, then PnP register 28h is used to assign the MPU-401 interrupt. If the MPU-401 is its own logical device, it can also be assigned to an interrupt via PnP register 70h of LDN #3. Both of these methods access the same physical register.

Interrupt Status Register (ISR)

Register 6 of the configuration device can be read to quickly find the current state of ES1878 interrupt sources.

Bit 0	Audio 1 interrupt request
Bit 1	Audio 2 interrupt request AND'ed with bit 6 of Mixer Extension register 7Ah
Bit 2	Hardware volume interrupt request AND'd with bit 1 of Mixer Extension register 64h
Bit 3	MPU-401 receive interrupt request AND'ed with bit 6 of Mixer Extension register 64h

Interrupt Mask Register (IMR)

Register 7h of the configuration device can be used to mask any of the seven interrupt sources.

The mask bits can be used to force the interrupt source to be zero, but they do not put the interrupt pin in a high-impedance state. Each bit is AND'ed with the corresponding interrupt source. This register is set to all ones on a hardware reset.

The Interrupt Status Register (ISR) is not affected by the state of the Interrupt Mask Register (IMR). That is, the ISR reflects the status of the interrupt request lines before being masked by the IMR.

The IMR is useful when interrupts are shared. For example, assume that audio 1, audio 2, hardware volume, and MPU-401 all share the same interrupt in Windows™. When returning from Windows to DOS, the hardware volume, MPU-401, and Audio 2 interrupts can be masked by setting the appropriate bits to 0.

A second use of the IMR is within an interrupt handler. The first thing the interrupt handler can do is mask all of the interrupt sources mapped to the interrupt handler. The ISR can then be polled to decide which sources to process. Just before exiting the interrupt handler, the IMR can be restored. If an unprocessed interrupt remains active, it will generate an interrupt request because the interrupt pin was low during the masked period and then went high when the interrupt sources were unmasked. While the interrupts are masked, the individual interrupt sources can change state any number of times without generating a false interrupt request.

Interrupt Edge Generator

The interrupt logic has a feature that makes sharing of interrupts easier. If more than one interrupt source shares an interrupt request pin, the interrupt pin is normally the logical OR of the shared interrupt requests. However, if any one interrupt request goes from high to low, circuitry inside the ES1878 will hold the interrupt request pin low briefly to generate a clock edge if one of the other interrupt sources is also high.

Sharing Interrupts

Plug and Play does not support sharing of interrupts in its resource assignment decision making. If a device wants to share an interrupt with another device that has been assigned an interrupt by PnP, the first device cannot request an interrupt for itself.

A logical device that supports interrupts can be assigned to an interrupt after the PnP sequence is generated by the Windows driver. In this case, the logical device would typically be forced to share an interrupt with the first audio interrupt. For most cases, this is done simply by programming the appropriate PnP register (70h or 72h) for the selected device.

A special case is the hardware volume interrupt. This interrupt source can be assigned to an interrupt through Vendor-Defined Card-Level register 28h, bits 7:4.

A second special case is the MPU-401 interrupt. The MPU-401 device is either part of the audio device or its own logical device. If it is part of the audio device, the interrupt can be assigned by writing to Vendor-Defined Card-Level register 28h, bits 3:0. If the MPU-401 device is its own logical device, it is assigned an interrupt by either Vendor-Defined Card-Level register 28h or LDN#3 register 70h.

PERIPHERAL INTERFACING

I²S Serial Interface

Three input pins, IIDATA, IISCLK, and IILR, are used for a serial interface between an external device and a stereo DAC within the ES1878. A fourth input, IIMCLK, is reserved for future devices that incorporate oversampling and should be left floating or connected to ground. IIDATA, IISCLK, and IILR can be left floating or connected to ground if the serial interface is not used.

Typical applications of the I^2S serial interface are MPEG audio, or digital CD audio.

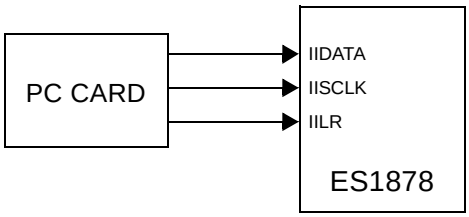


Figure 8 I²S Implementation in ES1878

I²S Serial Interface Format Select

This serial interface supports two different formats: ES689/ ES690 two-wire serial interface, or I²S. When used in the ES689/ES690 format, IIDATA is the serial data and IISCLK is the bit clock. The IILR input is not used and can be left floating or connected to ground. Vendor-Defined Card-Level register 29h bits 5 and 4 select the format (this register is accessed through the configuration device). See Vendor-Defined Card-Level register 29h in the 'Configuration Description' section for more detailed information.

I²S Serial Interface Timing

This section discusses the I²S serial interface signals. The signals when the port is configured for use with an ES689/ ES690 wavetable synthesizer is defined in the Wavetable Interface section.

Three signals (plus one optional) are used for I²S:

IISCLK The bit clock/shift clock. The maximum rate is 6.4 MHz. The minimum number of IISCLK periods per IILR period is 32. Any number greater than or equal to 32 is acceptable.

IILR Sample synchronization signal. The maximum sample rate is 50 kHz.

IIDATA Serial data.

IIMCLK Optional oversampling clock (for future use).

Within the ES1878, IILR and IIDATA are sampled on the rising edge of IISCLK. See Figure 22 and Figure 23 for detailed I²S timing.

MPU-401 Interface

The MPU-401 port can be used for interfacing with either MIDI or with ES938 3-D stereo sound effects signals. Refer to the ES938 Data Sheet for technical details.

MIDI

The ES1878 has a MPU-401 MIDI interface with a 23-byte receive FIFO and an 8-byte transmit FIFO. The output of the transmit FIFO is serialized out the MSO pin and also sent to the ES978 in the expansion unit, where it is serialized out the MSO pin of that chip.

MIDI data can be received from either the MSI pin of the ES1878 or from the MSI pin of the ES978 in the expansion unit. In the unlikely event that MIDI data is received from both sources simultaneously, the data might be corrupted. Data received by the ES978 is transmitted back to the ES1878 in the next upload frame and then placed in the MPU-401 receive FIFO.

Wavetable Interface

The ES1878 contains a synchronous serial interface for connection to a wavetable music synthesizer.

Table 4 Wavetable Interface Pins

Pins	Descriptions
MCLK	Serial clock from external ES689/ES690 music synthesizer (2.75 MHz). Input with pull-down.
MSD	Serial data from external ES689/ES690 music synthesizer. When both MCLK and MSD are active, the stereo DACs normally used by the FM synthesizer are acquired for use by the external ES689/ES690. The normal FM output is blocked. Input with pull-down.

DSP Interface

The ES1878 contains a synchronous serial interface for connection to a DSP serial interface. The typical application for this interface is a speakerphone.

Table 5 DSP Interface Pins

Pins	Descriptions
DCLK	Data clock. The rate can vary, but a typical value is 2.048 MHz (8 kHz x 256). Input with pull-down.
DX	Data transmit. Active output when data is being transmitted serially from the ES1878; otherwise high-impedance. Tri-state output.
DR	Serial data input with pull-down.
FS	Frame sync input. Software programmable to be active-high or active-low. Input with pull-down.

DSP Interface Serial Data Format

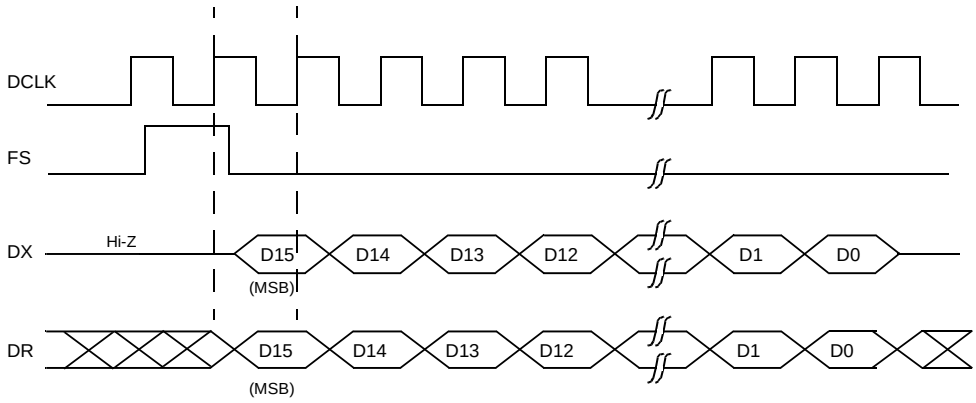


Figure 9 16-Bit Data, Positive Sync Pulse

Game/Joystick Interface

The ES1878 includes 8 pins for a dual joystick port. The digital game port address is decoded for timer pins TA, TB, TC, and TD, and for switch pins SWA, SWB, SWC, and SWD. The MIDI serial input and output also come from the game port connector in most applications.

Four of these eight pins, SW(A-D), are inputs for the switches of the joysticks. The remaining 4 pins, T(A-D), are "one-shot" timers that generate pulses of varying widths, where the width corresponds to the current resistance of one of the joystick potentiometers.

PC Joysticks

Normally, the host processor is responsible for measuring the width of the pulse. The ES1878 can also do this automatically. The host processor can read the measured widths directly rather than having to do the timing itself. This is referred to as a “digital joystick.” Bit 1 of Vendor-Defined Card-Level register 29h determines whether the joystick port is a digital or analog joystick.

Digital Joysticks

For digital joysticks, the host processor first writes any value to the joystick port, and then reads back seven separate values (shown in Table 6).

Table 6 Digital Joystick Read Values

Read #1	Low byte timer A
Read #2	Low byte timer B
Read #3	Low byte timer C
Read #4	Low byte timer D
Read #5	Bits 3:0 – Upper nibble timer A Bits 7:4 – Upper nibble timer B
Read #6	Bits 3:0 – Upper nibble timer C Bits 7:4 – Upper nibble timer D
Read #7	Bit 0 – switch A Bit 1 – switch B Bit 2 – switch C Bit 3 – switch D

The timer values reported range from 0 to FFFh (0-4095). The timer clock is 895 kHz.

When docked, a software programmable bit (bit 0 of Vendor-Defined Card-Level register 29h) causes the joystick connected to the ES978 to replace the one connected to the ES1878 automatically.

Joystick/MIDI External Interface Connector

The joystick portion of the ES1878 reference design is identical to that on a standard PC game control adaptor or game port. The PC-compatible joystick can be connected to a 15-pin D-sub connector. It supports all standard PC joystick compatible software.

If you need to support two joysticks, a joystick conversion cable is required. This cable uses a 15-pin D-sub male connector on one end and two 15-pin D-sub female connectors on the other end. All signals on this cable have direct pin-to-pin connection, except for pins 12 and 15. On the male connector, pins 12 and 15 should be left without connection. On the female connector, pin 15 is internally connected to pin 8, and pin 12 is internally connected to pin 4. The dual joystick port and MIDI port take up only one slot in your PC, leaving room for other cards. The dual joystick/MIDI connector configuration is shown in Figure 10.

The MIDI Serial Interface Adaptor from the Joystick/MIDI Connector is shown in Figure 11.

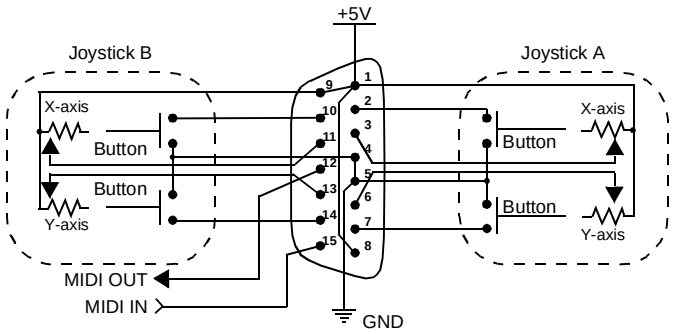


Figure 10 Dual Joystick/MIDI Connector

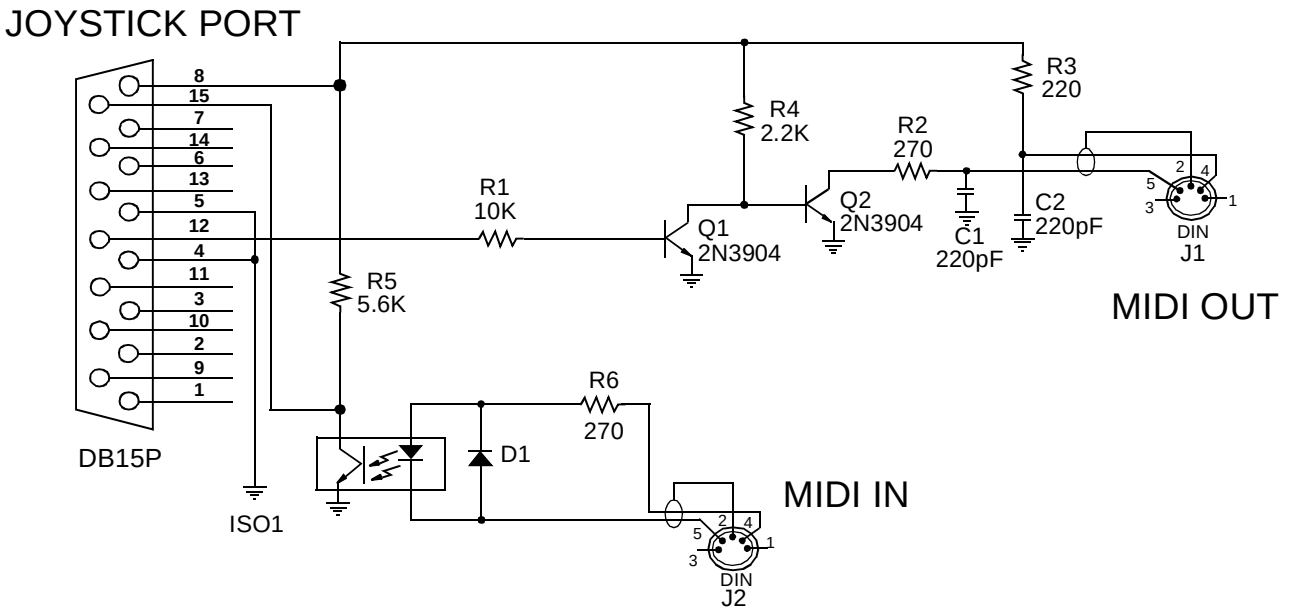


Figure 11 MIDI Serial Interface

ES978 Interface

When docked, the ES1878 is in constant communication with the ES978 in the expansion unit. A half-duplex, bidirectional serial link keeps each chip updated on the status of the other. For example, the Mixer registers located in the ES1878 are transmitted down to the ES978. MIDI data received in the ES978 is transmitted up to the ES1878.

In addition to the digital control link, four analog wires connect the two chips directly. These four wires are configured as a pair of differential audio channels.The ES1878 uses these two audio channels in one of three ways: stereo playback (ES1878 transmits to the ES978), stereo record (the ES978 transmits to the ES1878), and mono full-duplex (one mono channel in each direction).

Docking Status

The ES1878 is either in docked or undocked state. The state is determined by the DOCKED input, which is active high when docked.

In the undocked state, the XSC and XSD pins are driven low. The XA[3:0] pins act as in mode 0 (differential outputs), except they follow the AOUT L/R outputs directly (i.e., after the master volume).

Playback Mode

The ES1878/ES978 design assumes that the active speakers move from the portable to the expansion unit when docked. Except when recording, expansion audio sources are mixed in the expansion unit within the ES978, and played through speakers in the expansion unit. In most cases, speakers within the portable unit are programmed to be automatically muted when docked.

An exception is PC speaker beeps, which are always heard in the portable, even when docked.

Each audio input can be programmed individually on how to respond to a docking situation, in one of the three ways:

- As an analog input, such as mic, which remains in the portable (ES1878) when docked and is muted in the ES978.
- As an analog input, such as line-in, which is disabled in the ES1878 when docked and enabled in the ES978 when docked, i.e. it is muted in the mixers in the ES1878 and ES978.
- As an analog input which is enabled in both the ES1878 and ES978 mixers, and shares a common volume control. (Note: An exception to sharing a common volume control is allowed for the Mappable Volume register 6Ah; see below).

The playback master volume is controlled through software programming or by the up/down/mute switch inputs. The latter method, called hardware volume control, has active-low switch inputs in both the ES1878 and ES978.

Record Mode

In record mode, the expansion audio bus is turned around, and sound data is sent from the expansion unit ES978 chip to the ES1878 in the portable unit. The sound data from the expansion unit can be mixed inside the ES1878 with local sources before recording. Because portable unit sources (for example, FM) can be mixed into the recording, it is not possible to do a record monitor function through the expansion unit speakers (they are automatically muted in record mode). It is possible for record monitor to use the speakers in the portable unit (see bit 5 of Vendor-Defined Card-Level register 2Bh).

The default situation for most applications is to have all speakers muted during recording.

As in previous chips, one of four record sources can be selected: Mic, Line, Aux A, or Mixer. When docked, the ES1878 chip knows whether each resource is present in the portable, the docking station, or both, and acts accordingly.

No ES978 – Differential AOUT Mode

In some applications, there is no ES978 in the expansion unit. In this case, XA[3:0] are used as differential outputs that follow AOUT L/R and are intended to connect to a differential-input power amplifier in the expansion unit. This mode of operation is selected automatically whenever the DOCKED input is zero. When DOCKED is zero, XSC and XSD are high-impedance.

Mono Full-Duplex Mode

In the ES1878, host-based software applications can use full-duplex mode through two 8-bit DMA channels. The restrictions are that both record and playback are monophonic, and that record and playback are synchronous (i.e., at the same sample rate).

The record channel can record from any analog input of either the ES1878 or remote ES978 chips, or any mix of the same, or from the FDXI input to the ES1878 when using the DSP serial port.

Power Management

Power management is controlled by Vendor-Defined Card-Level register 2Dh. In previous ESS *Audio***Drive**® chips, power management was controlled via I/O port Audio_Base+7h. Only bit 5 (FM reset) and bit 7 (suspend request) of I/O port Audio_Base+7h are supported in the ES1878.

Expansion Audio Interface – Digital

Two wires are used to transmit serial data between the ES1878 and ES978. The first signal , XSC, acts as a frame sync and shift clock. The bit clock rate is 3.58 MHz.

A typical frame consists of:

- Sync period – 24 clocks wide
- Download period – 144 clocks wide
- Turn-around period – 8 clocks wide
- Upload Period – 80 clocks wide

Total: 256 bit clocks/frame, which is equivalent to a 14 kHz frame rate.

The function of the upload and download periods is to continually update corresponding registers within each device. For example, pressing the VOLUP button in the expansion unit transmits the pin state to the ES1878 where it is AND'ed with the same pin of the ES1878. The ES1878 updates its copy of the master volume register. The ES978 will receive the new value in the master volume register during the next download period of the next frame.

Sync Period

In the sync period, XSC is low for 12-bit clock periods, and then high for 12-bit clock periods.

Download Period

In the download period, data is transmitted serially from the ES1878 to the ES978 via the signal XSD. XSC is the bit shift clock. Data is shifted out of the ES1878 on the falling edge of XSC. Data is shifted into the ES978 on the rising edge of XSC.

The download period is 144 bits wide. Each bit takes 4 oscillator clocks (bit rate = 3.58 MHz). The last 8 bits are a checksum byte.

The upload period is 80 bits wide. The last 8 bits are a checksum byte.

Table 7 contains the data configuration for the download period.

Table 7 Download Period

Byte	Bits	Function
0	1:0 4:2 5 6 7	Mode of expansion analog interface Record source select Master output enable 1: MIDI loopback test 1: MIDI transmit signal (byte 1 contains MIDI data)
1	15:8	MIDI data (if bit 7 of byte 0 is high)
2	23:15	XGPO[7:0] data
3	31:24	Playback mixer – Host audio volume
4	39:32	Playback mixer – Line volume
5	47:40	Playback mixer – Mic volume
6	55:48	Playback mixer – Aux A (CD) volume
7	63:56	Playback mixer – Aux B volume
8	71:64	Playback mixer – I²S/ES689 volume
9	79:72	Reserved
10	87:80	Record mixer – Line volume
11	95:88	Record mixer – Mic volume
12	103:96	Record mixer – Aux A (CD) volume
13	111:104	Record mixer – Aux B volume
14	119:112	Record mixer – I²S/ES689 volume
15	126:120 127	Master volume left 1: Mute left
16	134:128 135	Master volume right 1: Mute right
17	143:136	CRC checksum

Turn Around Period

There are 8 bits between the end of the download period and the start of the upload period.

Upload Period

In the upload period, data is transmitted serially in the opposite direction, from the ES978 to the ES1878 via the same signal wire, XSD.

Table 8 contains the data configuration for the upload period.

Table 8 Upload Period Data Configuration

Byte	Bits	Function
0	3:0 4 5 6 7	Joystick switch status VOLUP input status VOLDN input status MUTE input status 1: MIDI receive data following
1	15:8	MIDI receive data if bit 7 of byte 0 is set.
2	23:16	XGPI input state
3	31:24	Low byte joystick timer A
4	39:32	Low byte joystick timer B
5	47:40	Low byte joystick timer C
6	55:48	Low byte joystick timer D
7	59:56 63:60	High nibble joystick timer A High nibble joystick timer B
8	67:64 71:68	High nibble joystick timer C High nibble joystick timer D
9	79:72	CRC checksum

Expansion Audio Interface – Analog

This interface uses six wires: two analog ground wires and four analog signal wires (XA[3:0]). The four signal wires are used in one of five different modes. In each of these modes, the master always refers to the ES1878 and the slave always refers to the ES978.

- Mode 0 – Stereo playback. Two differential pairs for left and right channels, transmitted from the master to the slave.
- Mode 1 – Stereo record. Two differential pairs for left and right channels, transmitted from the slave to the master.
- Mode 2 – Monophonic full-duplex. Two differential pairs. One pair is for monophonic playback from master to slave, the second pair is for monophonic recording from slave to master. The mono playback signal is input to both left and right host audio inputs of the playback mixer. The mono record signal is derived by averaging the left and right outputs of the record mixer.
- Mode 3 – Stereo full-duplex. The four signals are not used differentially (note that this mode is not supported by the ES1878).
- Mode 4 – Not docked (DOCKED=0). Like Mode 0, except the analog outputs follow AOUT_L and AOUT_R rather than the output of the mixer.

After a change of mode, the data is muted at the receiving end for a period of 25 milliseconds. It is the responsibility of the master not to have contention caused by both ends transmitting on the same signal wire.

Table 9 shows the mode configurations when the notebook unit is docked.

Table 9 Docked Modes for Analog Audio

Mode	XA0	XA1	XA2	XA3
0	-Left Play	+Left Play	-Right Play	+Right Play
1	+Left Record	-Left Record	+Right Record	-Right Record
2	-Play	+Play	+Record	-Record
3 *	Left Record	Left Play	Right Record	Right Play

* Not supported in the ES1878

Mono FDXI and FDXO

FDXI is shared with AUXB_L and FDXO is shared with AUXB_R. The ES1878 supports the use of FDXI and FDXO as input to the ADC and output from the DAC when using the DSP serial port. The ES1878 also supports a new function, in which FDXI is a general mono input to the mixer, controlled by the AUXB volume register, and FDXO is the mono output of the input volume stage (i.e., the recording source select and input volume control).

Mono FDXI/O mode is useful with an external modem that has integrated a CODEC for speakerphone applications.

Bit 1 of Mixer Extension register 48h enables FDXI as a mono input and FDXO as a mono output. When FDXI is a mono input to the mixer, its input impedance is cut in half to 25k ohms. When FDXO is an output, it has a 5k ohm output impedance. When FDXO is not an output it is the AUXB_R input to the mixer and has a 50k ohm pull-up to CMR.

Contact an ESS Field Application Engineer for an application note on how to use the FDXI/FDXO feature.

General-Purpose I/O

Up to seven general-purpose inputs and seven general-purpose outputs are available. Four of the GPO pins have other functions (ISA interrupt request outputs) and may not be available for use as general-purpose outputs. Five of the GPI pins have other functions (volume control, DRQC, DACKBC) and may not be available for use as general-purpose inputs. For more information, refer to the 'GPI/O Registers' section.

Each enabled GPI input can be read by the host processor at any time. Also, each GPI input can be programmed to remotely control a corresponding GPO output in the ES978, thereby saving interconnects between the portable and expansion unit.

Each enabled GPO pin can be controlled either by a writeby the host to an ES1878 register or remotely from a corresponding GPI pin of the ES978.

The worst-case latency between the ES978 and ES1878, due to the serial interconnection, is about 140 µsec.

GPI/O Registers

The GPI/O registers are as follows:

- Configuration_Device_Base+2h

  Bits 6:0 of this register set the state of the GPO[6:0] pins that are enabled as outputs and are not mapped to the GPI pins of the ES978.

- Configuration_Device_Base+3h

  Bits 7:0 of this register set the state of the XGPO[7:0] pins of the ES978 that are not mapped to the GPI pins of the ES1878.

- Vendor-Defined Card-Level register 25h

  This register controls whether any shared function pins are general-purpose inputs/outputs.

- Vendor-Defined Card-Level register 26h

  Vendor-Defined Card-Level register 26h, which is the GPO Map register, selects whether a GPO pin is controlled by Configuration_Device_Base+2h or by the GPI pin of ES978.

- Vendor-Defined Card-Level register 27h

  Vendor-Defined Card-Level register 27h, which is the GPI Map register, selects whether a GPI pin controls a XGPO pin in the ES978, or if the XGPO pin is controlled by Configuration_Device_Base+3h.

Note: Bits 1 and 0 of register Audio_Base+7h do not control GPO0 and GPO1 as in previous ESS *Audio***Drive**® chips. Also, the feature of previous audio controllers that causes GPO0 and GPO1 to change state automatically when the chip is powered down is no longer supported in the ES1878.

Master Volume

The master volume is controlled through programmed I/O or volume control switch inputs. The master volume supports 6-bits per channel plus mute. When docked, the ES1878 transmits the master volume information to the ES978 where it takes effect after the output of the ES978 mixer.

Hardware Volume Controls

VOLUP, VOLDN, and MUTE are three input pins with internal pull-up devices. The state of these pins is AND'ed with the state of the corresponding pins of the ES978 when docked. A software selectable option enables the mute input to be omitted. The mute input is defined as the state when both up and down inputs are low. By default, this feature is disabled.

The hardware volume inputs of the ES1878 can be used as general-purpose inputs (see bits 4 and 5 of Vendor-Defined Card-Level register 25h). In this case they cannot be used as volume control inputs.

PC Speaker

The PC speaker is supported by a 1-bit DAC with volume control. The analog output pin PCSPKO is intended to be externally mixed at the external amplifier, which means that the PC speaker audio is not transmitted to the ES978 through the expansion audio interface (XA[3:0]) but is always heard through the portable speakers.

PC Speaker Volume Control

When the PCSPKI signal is high, a resistive path to analog ground is enabled. The value of the resistor is selected from among 7 choices to control the amplitude of the output signal.

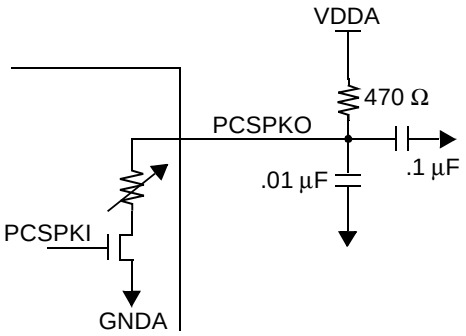


Figure 12 PC Speaker Volume Circuitry

With the external circuit shown in Figure 12, the amplitude of a square wave output on pin PCSPKO should be approximately VDDA/2 for maximum volume, i.e., the internal resistor is approximately 500 ohms (± 30%). The other levels are relative to this amplitude as follows:

off, -18dB, -15dB, -12dB, -9dB, -6dB, -3dB, +0dB

The purpose of the circuit, beyond volume control of the speaker, is to prevent digital noise from the PC speaker signal being mixed into the analog signal. This circuit provides a clean analog signal. The output can either be mixed with the AOUT_L and AOUT_R pins externally or used to drive a simple transistor amplifier to drive an 8 ohm speaker dedicated to producing beeps.

CONFIGURATION DESCRIPTION

Plug and Play (PnP) Configuration

The ES1878 supports the industry-standard ISA Plug and Play (PnP) specification, as well as a software configuration method that does not rely on PnP. An input pin of the ES1878 called PNPEN determines the configuration method.

PNPEN	Mode
0 (ground)	Non-PnP Mode
1 (Vcc)	ISA PnP Mode

Using ISA PnP Mode (PNPEN=1)

There are several design implications of using ISA PnP:

1. The PCI-ISA bridge for the ES1878 must be in subtractive decode mode. This mode is required because PnP can place the I/O addresses of the ES1878 devices at a very large number of locations.
2. All PnP devices within the system must share the same ISA bus and bridge.
3. The internal resource ROM cannot be changed. I/O addresses, interrupts, and DMA channels must be supported as defined by the resource ROM. The joystick port must be supported.
4. All five IRQ lines must be connected to ISA interrupt request channels as follows:

   IRQA -IRQ9
   IRQB -IRQ5
   IRQC -IRQ7
   IRQD -IRQ10
   IRQE -IRQ11
5. All three DRQ/DACK pairs must be connected to ISA DMA signals as follows:

   DRQA DRQ0
   DACKBA -DACK0
   DRQB DRQ1
   DACKBB -DACK1
   DRQC DRQ3
   DACKBC -DACK3

Accessing Non-PnP Mode (PNPEN=0)

Because of the above restrictions imposed by use of PnP, a separate configuration method is implemented in the ES1878. A special sequence of 34 bytes is written consecutively to I/O address 388h. This sequence is called the “bypass key” because it can be used when PNPEN=1 to short-circuit the PnP process and directly enable the configuration device of the ES1878 (note that when PNPEN=1, the sequence is written to I/O address 279h rather than 388h).

When PNPEN=0, the bypass key is required to enable the configuration device of the ES1878. Once the configuration device is enabled, all the PnP registers of the ES1878 are accessible and can be programmed.

Access to PnP Registers

“Bypass Key”

The PnP registers can be directly accessed, bypassing the PnP sequence, by writing a special key sequence to port 279h and concluding with two I/O writes to 279h to set the base address of the configuration ports. The key sequence also sets the activate bit for the configuration device.

If PnP is not supported by the system, it is possible to bypass PnP by issuing a special “bypass key” at any time to the ES1878 to force the configuration device to be enabled at a specific I/O address. The special key is 32 bytes long, written to the PnP Address register (279h if PNPEN=1, 388h if PNPEN=0). The bypass key must be followed immediately by two I/O writes to the PnP Address register to set the low and high bytes of the Address register of the configuration device. The configuration device is also activated by the bypass key. The address of the configuration device must be in the range 100h-FF8h and be aligned on a multiple of 8. You might use an “alias” of the audio device address that you intend to use: for example, E20h for the configuration device if the audio is at 220h.

Note: The entire sequence should be performed with interrupts disabled in order to minimize the chance that an interrupt will cause the sequence to be corrupted.

With interrupts disabled, the following 34 bytes are written to I/O address 388h (address 279h if PNPEN=1).

```
66, a1, c2, f1, ea, e7, 71, aa
c7, 63, 33, 1 b, d, 96, db, 6d
a4, 50, 28, 16, 9b, 4d, b6, c9
f4, 78, 3e, 8d, d6, fb, 7f, 3d
<config_address_low>, <config_address_high>
```

After writing this key, the configuration device will be activated at the specified address. The specified address should be an even multiple of 8 in the range 100h-0FF8h (800h-0FF8h recommended).

The bypass key can be written at any time, and it can be used to move the location of the configuration device (if PNPEN=1, the bypass key will not work unless the ES1878 is in “wait-for-key” mode). For a more detailed description of “wait-for-key” mode, see the ‘Device Configuration’ section.

PnP Configuration Registers

The PnP specification defines a register set for configuration of a "card" consisting of multiple "logical devices". The ES1878 acts as a card of three logical devices: the configuration device, the audio+FM+MPU-401 device, and the joystick device.

The PnP register set consists of a set of registers for the "card", and a bank-selected set of registers for each logical device.

In addition, the PnP specification sets aside a register space for Vendor-Defined Card-Level registers. The function of these registers is determined by the designer of the card or chip. The ES1878 has 14 Vendor-Defined Card-Level registers defined, from PnP register 20h to 2Dh.

The PnP configuration registers of the ES1878 can be accessed in two ways. First, if PNPEN=1, the registers can be read or written as defined by the PnP specification. Second, regardless of the state of PNPEN, the registers can be read or written using two I/O locations of the configuration device.

The first location of the configuration device is written with the PnP register number to be read or written. After programming the PnP register number, the register can be read or written by accessing the second location of the configuration device.

Example

Using Non-PnP Method to Configure the ES1878:

1. Enable the configuration device at I/O address 800h by sending the bypass key with interrupts disabled.
2. Program Vendor-Defined Card-Level registers 20h-2Dh as follows: write Vendor-Defined Card-Level register number to I/O address 800h and data to I/O address 801h.
    a 20h, 21h, 22h: assign ISA IRQ channels to pins. Unused pins are assigned to IRQ1.
    b 23h, 24h: assign ISA DMA channels to pins. Unused pins are assigned DRQ2.
    c 25h: set bit 7 low, other bits as needed to enable GPOs or GPIs.
    d 26h: defines whether GPOs are under software or ES978 control.
    e 27h: defines whether GPIs control ES978 GPOs.
    f 28h: leave all of the bits set to 00h until Windows starts up. The Windows driver will write this register to allow sharing MPU-401 and H/W volume interrupts.
    g 29h: bit 7 high is recommended. Set bits 6:4 based on I^2S interface use. Bit 1 should be left low in the BIOS. Bit 0 is based on the system design.
    h 2Ah: set this register to default value 0Eh.
    i 2Bh: leave bits 7:5 set at 00h. Bits 4:0 settings are based on the system design.
    j 2Ch: settings are based on the system design.
    k 2Dh: leave this register value set at 03h (fully powered on).
3. Configure and enable audio device. Set register 7h (LDN number) to 01h. Then program device registers 60h-65h, 70h, 74h and 75h. Leave register 72h set at 00h (second interrupt not used). Finally, set register 30h to 01h in order to activate the audio device.
4. (Optional) Configure and enable joystick device. Set register 7h (LDN number) to 02h. Then program registers 60h and 61h with base address. Set register 30h to 01h in order to activate the joystick device.

Register Set

As shown in Figure 13 below, the Card-Level registers supported by the ES1878 are the Card-Control Card-Level registers at addresses 00h-07h, and the Vendor-Defined Card-Level registers at addresses 20h-2Fh. The Card-Control Card-Level register at address 07h is a pointer to the Logical Device registers supported by the ES1878 (one set of registers for each logical device on the "card"). In the ES1878, there are three logical devices: the configuration device, the audio+FM+MPU-401 device, and the joystick device.

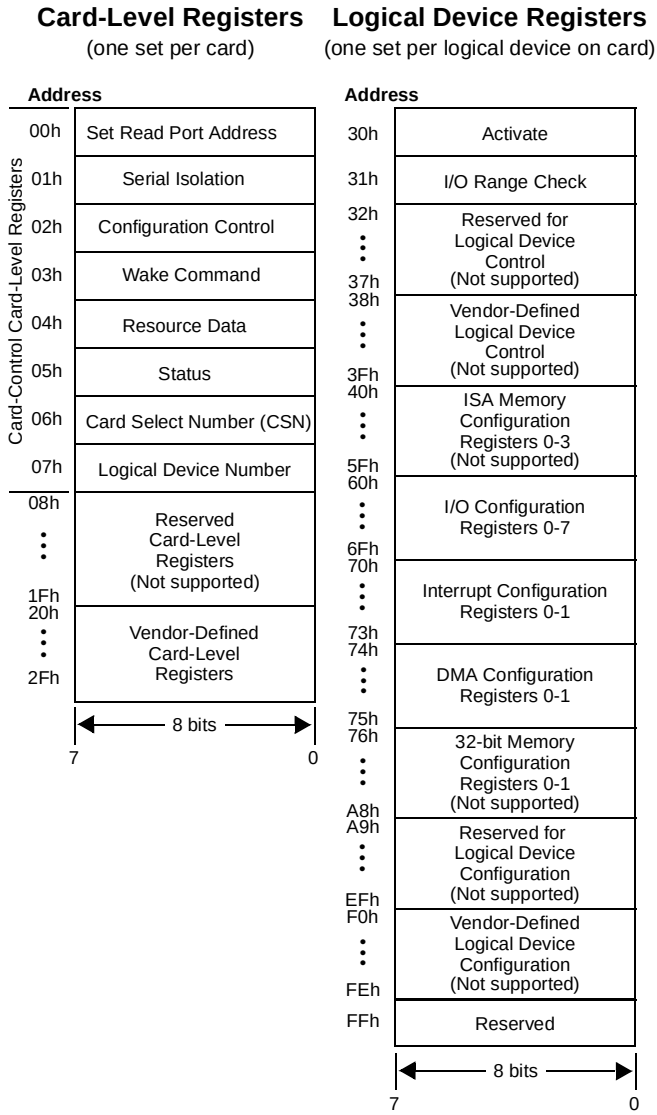


Figure 13 Configuration Register Outline

Card-Control Card-Level Registers (00h – 07h)

Set RD_DATA Port **(00h, Read/Write)**

7	6	5	4	3	2	1	0
Bits 9:2 of the PnP RD_DATA port							

The PnP Read port can be written only when the card is in isolation mode. It is reset low by hardware reset. It can be read only from configuration mode. Bits 1:0 of PnP read port are always 1.

Serial Isolation **(01h, Read-only)**

7	6	5	4	3	2	1	0
data							

Read-only in isolation state.

Configuration Control **(02h, Write-only)**

7	6	5	4	3	2	1	0
					RESET_CSN	WFK	SWR

Bit 2 RESET_CSN command.

Bit 1 WAIT_FOR_KEY command.

Bit 0 Software reset command. Does not work in WAIT_FOR_KEY state.

Wake **(03h, Write-only)**

7	6	5	4	3	2	1	0
data							

If data written matches CSN, then this card goes from Sleep state to Isolation state if CSN=0 and from Sleep state to Configuration if CSN <> 0.

Resource Data **(04h, Read-only)**

7	6	5	4	3	2	1	0
resource data							

Returns next byte of resource data. Only works in configuration mode.

Status **(05h, Read-only)**

7	6	5	4	3	2	1	0
reserved							status

Returns status in bit 0.

Bit 0 0 = not ready.
1 = ready to read resource data. Only works in configuration mode.

CSN **(06h, Read/Write)**

7	6	5	4	3	2	1	0
card select number							

Read/write card select number. Write only works in isolation mode. Causes transition to configuration mode. Read only works in configuration mode.

LDN **(07h, Read/Write)**

7	6	5	4	3	2	1	0
logical device number							

Read/write logical device number. Only works in configuration mode.

Vendor-Defined Card-Level Registers (20h – 2Fh)

IRQB, IRQA **(20h, Read-only)**

7	6	5	4	3	2	1	0
IRQB				IRQA			

Defines IRQ number assigned to B and A pins. Loaded from Configuration ROM Header after PnP reset. Unused IRQ pins should be assigned IRQ #1.

IRQD, IRQC **(21h, Read-only)**

7	6	5	4	3	2	1	0
IRQD				IRQC			

Defines IRQ number assigned to D and C pins. Loaded from Configuration ROM header after PnP reset. Unused IRQ pins should be assigned IRQ #1.

IRQE **(22h, Read-only)**

7	6	5	4	3	2	1	0
0	0	0	0	IRQE			

Defines IRQ number assigned to E pin. Loaded from Configuration ROM header after PnP reset. Unused IRQ pins should be assigned IRQ #1.

DRQB, DRQA **(23h, Read-only)**

7	6	5	4	3	2	1	0
DRQB/DACKBB				DRQA/DACKBA			

Defines DRQ number assigned to B and A pins. Loaded from Configuration ROM header after PnP reset. Unused DRQ pins should be assigned DRQ #2.

DRQC **(24h, Read-only)**

7	6	5	4	3	2	1	0
0	0	0	0	DRQC			

Defines DRQ number assigned to C pin. Loaded from Configuration ROM header after PnP reset. Unused DRQ pins should be assigned DRQ #2.

Shared Function Assignment **(25h, Read-only)**

7	6	5	4	3	2	1	0
MPU part of LDN1/ LDN3	X	VOLUP, VOLDN/ GPI5,GPI6	MUTE/ GPI4	IRQE/ GPO6	IRQD/ GPO5	IRQC/ GPO4	IRQB/ GPO3

Bit 7 0 = MPU is part of LDN#1.
1 = MPU is LDN#3.

Bit 5 0 = VOLUP/VOLDN.
1 = GPI5/GPI6.
Pin 91/92.

Bit 4 0 = MUTE.
1 = GPI4.
Pin 90.

Bit 3 0 = IRQE.
1 = GPO6.
Pin 93.

Bit 2 0 = IRQD.
1 = GPO5.
Pin 94.

Bit 1 0 = IRQC.
1 = GPO4.
Pin 95.

Bit 0 0 = IRQB.
1 = GPO3.
Pin 96.

GPO Map (26h, Read-only)

7	6	5	4	3	2	1	0
X	GPO[6:0]						

One bit for each of GPO[6:0]:

Bits 6:0 0 = GPO pin controlled by port Configuration_Device_Base+2h.
1 = GPO pin controlled by corresponding GPI pin of ES978.

This register is reset to zero by hardware reset but not by PnP reset.

GPI Map (27h, Read-only)

7	6	5	4	3	2	1	0
X	GPI[6:0]						

One bit for each of GPI[6:0]:

Bits 6:0 0 = GPI pin does not control GPO pin of ES978. The GPO pin of the ES978 is controlled by port Configuration_Device_Base+3h.
1 = GPI pin controls corresponding GPO pin of ES978.

This register is reset to zero by hardware reset but not by PnP reset.

MPU-401 & H/W Volume IRQ Number (28h, Read-only)

7	6	5	4	3	2	1	0
H/W volume IRQ number				MPU-401 IRQ number			

Bits 7:4 H/W volume IRQ number (must be shared with audio1 or audio2).

Bits 3:0 MPU-401 IRQ number (alias address with register 70h of MPU-401 LDN #3).

Miscellaneous Digital Control (29h, Read/Write)

7	6	5	4	3	2	1	0
DRQ latch feature	I^2S port enable/ disable	I^2S port format		MIDI loopback test	X	Analog/ digital Joystick	ES1878 /ES978 joystick toggle

Bit 7 0 = DRQ latch feature disabled (default).
1 = DRQ latch feature enabled.

Bit 6 0 = Disable I^2S port.
1 = Enable I^2S port.
By hardware reset default, the I^2S interface is disabled.

Bits 5:4 0 0 = I^2S port is ES689/ES690 format. IIDATA is data. IISCLK is bit clock. IILR should be connected low or float.
0 1 = Reserved.
1 0 = I^2S port is I^2S format (hardware reset default).
1 1 = Reserved.

Bit 3 1 = Enable ES978 MIDI loopback test.

Bit 2 Reserved.

Bit 1 0 = Analog joystick (default).
1 = Digital joystick.

Bit 0 0 = Use ES1878 joystick when docked.
1 = Use ES978 joystick when docked (default).

This register is reset to 21h by hardware reset but not by PnP reset. The hardware reset default format is I^2S.

Special Volume (2Ah, Read-only)

7	6	5	4	3	2	1	0
Volume mixed into ES978 playback				Volume mixed into ES978 record			

Bits 7:4 Volume of host audio mixed into ES978 record mixer (default = 0).

Bits 3:0 Volume of host audio mixed into ES978 playback mixer (default = 0Eh).

This register is reset to 0Eh by hardware reset but not by PnP reset.

Miscellaneous Analog Control (2Bh, Read-only)

7	6	5	4	3	2	1	0
Vol of external record sources		ES1878 master volume mute toggle	ES978 I^2S volume tracking	ES1878 I^2S volume control	ES978 mappable mixer target		

The I^2S DAC volume can track the volume of the FM DAC when its own Mixer Volume Control register 68h is not

used. This is useful when the I²S interface is used for an external wavetable synthesizer. Bit 3 of Vendor-Defined Card-Level register 2Bh when set high will enable tracking with FM volume.

Bits 7:6 Common volume of mix with external record sources:
0 0 = 0 dB (default)
0 1 = -1.5 dB
1 0 = -3 dB
1 1 = -4.5 dB

Bit 5 0 = ES1878 master volume muted when docked (default).
1 = ES1878 master volume not muted when docked

Bit 4 0 = ES978 I²S volume tracks I²S Mixer register 68h of ES1878 (default).
1 = ES978 I²S volume tracks FM Mixer register 36h of ES1878.

Bit 3 0 = ES1878 I²S volume controlled by I²S Mixer register 68h (default).
1 = ES1878 I²S volume controlled by FM Mixer register 36h.

Bits 2:0 Assigns ES978 target for Mappable Mixer register 6Ah:
0 = None (default).
1 = Mic
2 = Line
3 = Aux A
4 = Aux B
5= I²S/ES689/ES690 interface

This register is reset to zero by hardware reset but not by PnP reset.

Resource Assignment (2Ch, Read-only)

7	6	5	4	3	2	1	0
Mic control		Line control		AuxA control		AuxB control	

Bits 7:6 AuxB control (see table below).
Bits 5:4 AuxA control (see table below).

Bits 3:2 Line control (see table below).

Bits 1:0 Mic control (see table below).

Bit Values	ES1878	ES978
0 0	Mute	Mute
0 1	Mute	Enabled: track corresponding 1878 Mixer register
1 0	Enabled	Mute
1 1	Enabled	Enabled: track corresponding 1878 Mixer register

After hardware reset or PnP reset, this register is set to 3.

Power Management (2Dh, Read/Write)

7	6	5	4	3	2	1	0
not used						Power	

Bits 1, 0 = 0 0 Fully powered down.
0 1 Oscillator enabled, everything else powered down.
1 0 Low-power mode: analog enabled, expansion interface enabled, mpu-401 enabled, joystick enabled, I²S interface enabled, PnP enabled. DSP and ES689/ES690 serial interfaces are disabled, audio device and FM disabled.
1 1 Fully powered up.

After hardware reset or PnP reset, this register is set to 3. Suspend and Resume operation requires programming register Audio_Base+7h.

Logical Devices

Table 10 Logical Device Summary

LDN #	Device
LDN #0 (mandatory)	**Configuration device**
30h	Activate Bit 0 is activate bit.
31h	I/O Range Check
60h	I/O base address bits 11:8. If 0, this device is not disabled. 8 locations
61h	I/O base address bits 7:0
LDN #1 (mandatory)	**Audio device**
30h	Activate Bit 0 is activate bit.
31h	I/O Range Check
60h	I/O base address of Audio Processor bits 11:8. If 0, this device is not accessable.16 locations
61h	I/O base address of Audio Processor bits 7:0
62h	I/O base address of FM alias, bits 11:8. If 0, this device is not accessible. 4 locations
63h	I/O base address of FM alias, bits 7:0.
64h	I/O base address of MPU-401, bits 11:8. If 0 this device is not accessible. MPU-401 may also be accessible through LDN #3. 2 locations
65h	I/O base address of MPU-401, bits 7:0.
70h	Interrupt Request Level Select0
71h	Returns 2
72h	Interrupt Request Level Select1
73h	Returns 2
74h	DMA channel select 0 (default = 4)
75h	DMA channel select 1 (default = 4)
LDN #2 (mandatory)	**Joystick device**
30h	Activate Bit 0 is activate bit.
31h	I/O Range Check
60h	I/O base address bits 11:8. If 0, this device is not disabled. 1 location
61h	I/O base address bits 7:0
LDN #3 (optional)	**MPU-401 device**
30h	Activate Bit 0 is activate bit.
31h	I/O Range Check
60h	I/O base address bits 11:8. If 0, this device is not disabled. 2 locations
61h	I/O base address bits 7:0
70h	Interrupt Request Level Select0
71h	Returns 2

LDN0: Configuration Device

Activate (30h, Read/Write)

7	6	5	4	3	2	1	0
reserved							activate

Bit 0 0 = deactivate (default), 1 = activate.

After reset or after a 1 is written to the reset bit in the card's configuration control bit, the default for this register is 0.

I/O Range Check (31h, Read/Write)

7	6	5	4	3	2	1	0
reserved						enable range check	pattern select

Verifies that the I/O range assigned to a logical device does not conflict with the I/O range used by another device.

Bit 1 Enable range check: 0 = disable, 1 = enable.

Bit 0 Pattern select: 0 = AAh, 1 = 55h.

Configuration Device – high (60h, Read/Write)

7	6	5	4	3	2	1	0
0	0	0	0	A[11:8]			

Used to assign an I/O base address to I/O decoder 0 of the logical device. I/O base address is bits 11:8.

Configuration Device – low (61h, Read/Write)

7	6	5	4	3	2	1	0
A[7:3]					0	0	0

I/O base address, bits 7:3.

LDN1: Audio Device

This device actually supports three functions: audio, FM, and MPU-401. Audio requires sixteen I/O locations, one interrupt which is shared with MPU-401, and two DMA channels. FM requires four I/O locations. MPU-401 requires two I/O locations.

Activate **(30h, Read/Write)**

7	6	5	4	3	2	1	0
reserved							activate

Bit 0 0 = deactivate (default), 1 = activate.

I/O Range Check **(31h, Read/Write)**

7	6	5	4	3	2	1	0
reserved						enable range check	pattern select

Verifies that the I/O range assigned to a logical device does not conflict with the I/O range used by another device.

Bit 1 Enable range check: 0 = disable, 1 = enable.

Bit 0 Pattern select: 0 = AAh, 1 = 55h.

Audio Processor I/O Base Address (60h, Read/Write)

7	6	5	4	3	2	1	0
0	0	0	0	A[11:8]			

I/O base address of audio processor, bits 11:8. Sixteen locations.

Audio Processor I/O Base Address (61h, Read/Write)

7	6	5	4	3	2	1	0
A[7:4]				0	0	0	0

I/O base address of audio processor, bits 7:4.

FM Alias I/O Base Address (62h, Read/Write)

7	6	5	4	3	2	1	0
0	0	0	0	A[11:8]			

I/O base address of FM alias, bits 11:8. Four locations.

FM Alias I/O Base Address (63h, Read/Write)

7	6	5	4	3	2	1	0
A[7:2]						0	0

I/O base address of FM alias, bits 7:2.

MPU-401 I/O Base Address (64h, Read/Write)

7	6	5	4	3	2	1	0
0	0	0	0	A[11:8]			

I/O base address of MPU-401 is bits 11:8. (MPU-401 may also be accessible through LDN #3). Two locations.

MPU-401 I/O Base Address (65h, Read/Write)

7	6	5	4	3	2	1	0
A[7:2]						0	0

I/O base address of MPU-401, bits 7:2.

Interrupt Request Level Select 0 (70h, Read/Write)

7	6	5	4	3	2	1	0
0	0	0	0	data			

Interrupt request level select 0.

Bits 3:0 select which interrupt level used for Interrupt 0.

Interrupt Request Type Select 0 (71h, Read/Write)

7	6	5	4	3	2	1	0
0	0	0	0	0	0	1	0

Interrupt request type select 0. Returns 2 (low-to-high transition).

Interrupt Request Level Select 1 (72h, Read/Write)

7	6	5	4	3	2	1	0
0	0	0	0	data			

Interrupt request level select 1.

Bits 3:0 select which interrupt level is used for Interrupt 0.

Interrupt Request Type Select 1 (73h, Read/Write)

7	6	5	4	3	2	1	0
0	0	0	0	0	0	1	0

Interrupt request type select 1. Returns 2 (low-to-high transition).

DMA Channel Select 0 (74h, Read/Write)

7	6	5	4	3	2	1	0
0	0	0	0	0	data		

Bits 2:0 select which DMA channel is in use for DMA 0.

DMA Channel Select 1 (75h, Read/Write)

7	6	5	4	3	2	1	0
0	0	0	0	0	data		

Bits 2:0 select which DMA channel is in use for DMA 1.

LDN2: Joystick Device

Activate Register **(30h, Read/Write)**

7	6	5	4	3	2	1	0
reserved							activate

Bit 0 0 = deactivate (default), 1 = activate.

I/O Range Check **(31h, Read/Write)**

7	6	5	4	3	2	1	0
reserved						enable range check	pattern select

Verifies that the I/O range assigned to a logical device does not conflict with the I/O range used by another device.

Bit 1 Enable range check: 0 = disable, 1 = enable.

Bit 0 Pattern select: 0 = AAh, 1 = 55h.

I/O Decoder 0 Base Address **(60h, Read/Write)**

7	6	5	4	3	2	1	0
0	0	0	0	A[11:8]			

I/O base address bits 11:8. One location.

I/O Decoder 0 Base Address **(61h, Read/Write)**

7	6	5	4	3	2	1	0
A[7:0]							

I/O base address, bits 7:0.

LDN3: MPU-401

The MPU-401, as an independent device, is optional; normally MPU-401 is part of the *Audio***Drive**®.

Activate Register **(30h, Read/Write)**

7	6	5	4	3	2	1	0
reserved							activate

Bit 0 0 = deactivate (default), 1 = activate.

I/O Range Check **(31h, Read/Write)**

7	6	5	4	3	2	1	0
reserved						enable range check	pattern select

Verifies that the I/O range assigned to a logical device does not conflict with the I/O range used by another device.

Bit 1 Enable range check: 0 = disable, 1 = enable.

Bit 0 Pattern select: 0 = AAh, 1 = 55h.

I/O Decoder 0 Base Address **(60h, Read/Write)**

7	6	5	4	3	2	1	0
0	0	0	0	A[11:8]			

I/O base address bits 11:8. Two locations.

I/O Decoder 0 Base Address **(61h, Read/Write)**

7	6	5	4	3	2	1	0
A[7:0]							

I/O base address, bits 7:0.

Interrupt Request Level Select 0 **(70h, Read/Write)**

7	6	5	4	3	2	1	0
0	0	0	0	data			

Interrupt request level select 0.

Bits 3:0 select which interrupt level is used for Interrupt 0.

Interrupt Request Type Select 0 **(71h, Read/Write)**

7	6	5	4	3	2	1	0
0	0	0	0	0	0	1	0

Interrupt request type select 0. Returns 2 (low-to-high transition).

Device Configuration

ES1878 logical device #0 is the *configuration device*. Table 11 shows the eight I/O ports assigned to the configuration device using the key sequence.

Two of the configuration device ports are used to address a set of direct PnP Configuration registers that define I/O resources and activation controls for audio, FM, MPU-401, and joystick devices. These ports are listed in Table 11.

Table 11 Configuration Device

Address Offset	Name
Base+0h	PnP Configuration register address
Base+1h	PnP Configuration register data
Base+2h	ES1878 General-Purpose Output register
Base+3h	ES978 General-Purpose Output register
Base+4h	ES1878 General-Purpose Input register (read-only)
Base+5h	ES978 General-Purpose Input register (read-only)
Base+6h	Interrupt Status register (read-only)
Base+7h	Secondary Interrupt Mask register

Base+0h Sets the PnP Configuration Address register.

Base+1h Sets the PnP Configuration Data register.

Base+2h Sets the state of the ES1878 GPO pins that are not mapped to GPI pins of the ES978.

Base+3h Sets the state of the ES978 GPO pins that are not mapped to GPI pins of the ES1878.

Base+4h ES1878 General-purpose input status (read-only).

Base+5h ES978 General-purpose input status (read-only).

Base+6h Interrupt Status register

bit 0: Audio 1
bit 1: Audio 2
bit 2: Hardware volume
bit 3: MPU-401
bits 4,5:

Bits		PnP States
5	4	
0	0	wait-for-key
0	1	sleep
1	0	isolation
1	1	configure

bit 6: PNPOK status bit
bit 7: Current docking state (status; not an interrupt request)

Base+7h Sets Secondary Interrupt Mask register.
bit 0: Audio 1 (Set high by hardware reset.)
bit 1: Audio 2 (Set high by hardware reset.)
bit 2: Hardware volume (Set high by hardware reset.)
bit 3: MPU-401 (Set high by hardware reset.)
bit 4: reserved
bit 5: reserved
bit 6: reserved
bit 7: reserved

Configuring IRQ, DMA, and GPI/O Pins

IRQ (Registers 20h, 21h, 22h, 70h, 72h)

If only one IRQ channel (IRQA) is connected to the bus, the Vendor-Defined Card-Level registers 20h, 21h, and 22h are to be programmed as follows:

register 20h = 10h + IRQA_channel_number	
	“1”=IRQB is unused
register 21h = 11h	“1” = IRQC/D are unused
register 22h = 01h	“1” = IRQE is unused

The recommended IRQ lines are:

5	first choice
9, 5, 7, 10	second choice
3, 4, 5, 7, 9, 10, 11, 12, 15	third choice

The IRQ channel number must also be programmed into register 70h of LDN1. When register 70h matches one of the channel numbers in registers 20h, 21h, or 22h, a connection is made. The second interrupt of the audio device is not used so register 72h should be zero.

DRQ/DACK (Registers 23h, 24h, 74h, 75h)

If DRQA/DACKBA is set for first audio channel DMA and DRQB/DACKBB is set for second audio channel DMA, program register 23h to be the first DMA channel number in bits 3:0 and the second DMA channel number in bits 7:4. Registers 74h and 75h of LDN1 must also be programmed to match the DRQA and DRQB channel numbers.

Example:

First DMA channel is “1” and second DMA channel is “3”.

register 23h = 31h
register 24h = 02h
“2” = DRQC/DACKBC are unused
register 74h, LDN1 = 01h
register 75h, LDN1 = 03h

GPI/O (Vendor-Defined Card-Level Register 25h)

As long as DRQC/DACKBC are not selected, these pins are usable as general purpose inputs without further setup.

General-purpose inputs, GPI 4, 5, and 6 are the MUTE, VOLUP, and VOLDN inputs. They can be read at any time, but if the pins are not to act as hardware volume control, bits 4 and 5 of Vendor-Defined Card-Level register 25h must be set.

To use IRQ(B-E) as GPOs, the appropriate bits in Vendor-Defined Card-Level register 25h must be set.

I/O PORTS

Table 12 I/O Port for Joystick, Audio, FM, and MPU-401 Devices

Port	Read/Write	Function
Audio Device		
Base+0h - Base+3h	Read/write	20-voice FM synthesizer. Address and data registers.
Base+4h	Read/write	Mixer Address register (port for address of Mixer Indirect registers).
Base+5h	Read/write	Mixer Data register (port for data to/from Mixer Indirect registers).
Base+6h	Read/write	Audio reset and status flags.
Base+7h	Read/write	Power Management register. Suspend request and FM reset.
Base+8h - Base+9h	Read/write	11-voice FM synthesizer. Address and data registers.
Base+Ah	Read-only	Input data from read buffer for command/data I/O. Poll bit 7 of port Audio_Base+Eh to test whether the read buffer contents are valid.
Base+Ch	Read/write	Output data to write buffer for command/data I/O. Read embedded processor status.
Base+Eh	Read-only	Data available flag from embedded processor.
Base+Fh	Read/write	Address for I/O access to FIFO in Extended Mode.
FM Device		
Base+0h - Base+3h	Read/write	20-voice FM synthesizer. Address and data registers.
MPU-401 Device		
Base+0h - Base+1h	Read/write	MPU-401 port (x=0,1,2, or 3) if enabled.
Joystick Device		
Base+0h	Read/write	Joystick.

Port Descriptions

Audio Device

Mixer Address Register
Audio_Base+4h **(Read/Write)**

7	6	5	4	3	2	1	0
X	A6	A5	A4	A3	A2	A1	0

Reading back this register is useful for a “hot-key” application that needs to change the mixer while preserving the address register.

Mixer Data Register
Audio_Base+5h **(Read/Write)**

7	6	5	4	3	2	1	0
D7	D6	D5	D4	D3	D2	D1	D0

Reset and Status Flags
Audio_Base+6h **(Write)**

7	6	5	4	3	2	1	0
0	0	0	0	0	0	FIFO reset	SW reset

Reset and Status Flags
Audio_Base+6h **(Read)**

7	6	5	4	3	2	1	0
Act flag 2	Act flag 1	Act flag 0	Serial act flag	0 if power -down	MIDI mode	FIFO reset	SW reset

Bits 7:4 of port Audio_Base+6h can be used to monitor I/O activity to the ES1878.

Bits 7:5 are set high after any read from port Audio_Base+6h. Specific I/O activity can then set these bits low. When port Audio_Base+6h is read at a later time, these bits indicate whether I/O activity has occurred between the reads from Audio_Base+6h.

In addition, bit 4 can be used to indicate if either the DSP or the ES689/ES690 serial interface is in use. Bit 4 is set high if bits 7 or 5 of Mixer register 48h is high (software serial enable or serial reset). It is also set high if the ES689/ ES690 serial interface is active, which is a combination of

bit 4 of Mixer register 48h set high and MCLK (ES689/ ES690 serial bit clock) being high periodically.

Reading port Audio_Base+6h returns the following information:

Bit 7 Set low by I/O reads/writes to MPU-401, Joystick, Configuration Device, as well as PnP I/O activity (the last including almost any I/O write to 279h or A79h if PNPEN=1.

Bit 6 Set low by I/O reads/writes to audio ports Audio_Base+4h and Audio_Base+5h (mixer ports)

Bit 5 Set low by I/O writes to audio and FM ports excepting Audio_Base+4h, Audio_Base+5h, and Audio_Base+7h. Set low by I/O reads from audio and FM ports excepting Audio_Base+4h, Audio_Base+5h, Audio_Base+6h, and Audio_Base+7h. Also set low by DMA accesses to ES1878.

Bit 4 1 = Serial activity flag. High if DSP serial mode is enabled (SE input pin is high or bit 7 of Mixer Extension register 48h is high) or if an external ES689/ES690 is using MCLK/MSD to drive the FM DACs.

Bit 3 0 = The ES1878 digital audio is currently powered down (power mode 0, 1, and 2).

Bit 2 1 = The ES1878 is processing a MIDI command 30h, 31h, 34h, or 35h. In this mode, the ES1878 is monitoring serial input. Powering down may cause loss of data.

Note that the ES1878 does not automatically wake up based on serial input on the MSI pin.

Bit 1 FIFO Reset register.

Bit 0 Software Reset register.

Power Management Register
Audio_Base+7h **(Read/Write)**

7	6	5	4	3	2	1	0
Suspend request	0	1:Reset FM Synth	0	0	0	0	0

Reading or writing port Audio_Base+7h will not automatically wake up the ES1878.

Bit 7 Suspend request. Pulse high, then low to request suspend.

Bit 6 Reserved, should be set low.

Bit 5 1 = FM synthesizer reset.
0 = Release FM synthesizer reset.

Bits 4:0 Reserved, should be set low.

Read Data Register
Audio_Base+Ah **(Read-only)**

7	6	5	4	3	2	1	0
D7	D6	D5	D4	D3	D2	D1	D0

Read data from embedded audio processor.

Write Data Register
Audio_Base+Ch **(Write)**

7	6	5	4	3	2	1	0
D7	D6	D5	D4	D3	D2	D1	D0

Write data to embedded audio processor.

Read Data Register
Audio_Base+Ch **(Read)**

7	6	5	4	3	2	1	0
D7	D6	D5	D4	D3	D2	D1	D0

Bit 7 Write data to write buffer for command/data I/O. Set Write-Buffer-Not-Available flag until data is processed by the ES1878 .
1 = write buffer not available or ES1878 busy.
0 = write buffer available and ES1878 not busy.

Bit 6 Same as bit 7 of port Audio_Base+Eh.

Bit 5 1 = Extended Mode FIFO Full (256 bytes loaded).

Bit 4 1 = Extended Mode FIFO Empty (0 bytes loaded).

Bit 3 1 = FIFO Half Empty, Extended Mode flag.

Bit 2 1 = ES1878 processor generated an interrupt request (e.g., from Compatibility Mode DMA complete).

Bit 1 1 = Interrupt request generated by FIFO Half Empty flag change. Used by programmed I/O interface to FIFO in Extended Mode.

Bit 0 1 = Interrupt request generated by DMA counter overflow in Extended Mode.

Read Buffer Status Register
Audio_Base+Eh **(Read-only)**

7	6	5	4	3	2	1	0
D7	D6	D5	D4	D3	D2	D1	D0

A read from port Audio_Base+Eh will reset any IRQ request.

Bit 7 1 = Data available in read buffer.

Programmed I/O Access to FIFO
Audio_Base+Fh **(Read/Write)**

7	6	5	4	3	2	1	0
D7	D6	D5	D4	D3	D2	D1	D0

This port can be used to replace Extended Mode DMA with programmed I/O.

FM Device

The FM synthesizer operates in two different modes: Emulation Mode and Native Mode. In Emulation Mode the FM synthesizer is fully compatible with the OPL3 FM synthesizer. In Native Mode the FM synthesizer has increased capabilities and performance for more realistic music. The following register descriptions are for Emulation Mode only.

FM Status Register
FM_Base+0h **(Read)**

7	6	5	4	3	2	1	0
IRQ	FT1	FT2	0	0	0	0	0

Reading this register returns the overflow flags for timers 1 and 2 and the “interrupt request” from these timers (this is not a real interrupt request but is supported as a status flag for backward compatibility with the OPL3 FM synthesizer).

FM Low Bank Address Register
FM_Base+0h **(Write)**

7	6	5	4	3	2	1	0
A7	A6	A5	A4	A3	A2	A1	A0

Low bank register address.

Note: any write to this register will also put the FM synthesizer in Emulation Mode if it is currently in Native Mode.

FM Data Write Register
FM_Base+1h **(Write-only)**

7	6	5	4	3	2	1	0
D7	D6	D5	D4	D3	D2	D1	D0

FM register write. The data written to FM_BASE+1h is written to the current address FM register. Note that register writes must follow the timing requirements of the OPL3 FM synthesizer.

FM High Bank Address Register
FM_Base+2h **(Write-only)**

7	6	5	4	3	2	1	0
A7	A6	A5	A4	A3	A2	A1	A0

High bank register address.

FM Data Write Register
FM_Base+3h **(Write-only)**

7	6	5	4	3	2	1	0
D7	D6	D5	D4	D3	D2	D1	D0

FM register write. Writing to this register in Emulation Mode is the same as writing to register FM_Base+1h.

MPU-401 Device

MPU-401 Data Register
MPU_Base+0h **(Read/Write)**

7	6	5	4	3	2	1	0
D7	D6	D5	D4	D3	D2	D1	D0

This register is used to read data from the MPU-401 receive FIFO or a command acknowledge byte (0FEh). This register is also used to write data to the MPU-401 transmit FIFO.

MPU-401 Command Register
MPU_Base+1h **(Write)**

7	6	5	4	3	2	1	0
D7	D6	D5	D4	D3	D2	D1	D0

The MPU-401 device accepts only two commands:

FFh	Reset/return to Smart Mode. This command generates an acknowledge byte if received when already in Smart Mode.
3Fh	Go to UART Mode. This command generates an acknowledge byte if received while in Smart Mode. It is ignored if the device is already in UART Mode.

MPU-401 Status Register
MPU_Base+1h **(Read)**

7	6	5	4	3	2	1	0
-RR	-TR	X	X	X	X	X	X

Bit 7 0 = read data available in the receive FIFO, or pending acknowledge byte to be read (0FEh).

Bit 6 0 = there is room in the transmit FIFO to accept another byte.

Joystick Device

The joystick device uses only a single I/O port. The device can function in one of two modes: Analog Mode or Digital Mode. The use of this I/O port is defferent depending on the Mode. This section describes Analog Mode. Digital Mode is described in the ‘Game/Joystick Interface’ section.

Joystick_Base+0h **(Write)**

7	6	5	4	3	2	1	0
X	X	X	X	X	X	X	X

Any value written to the Joystick_Base+0h port will restart the timing sequence. This should be done before reading the timer status flags.

Joystick_Base+0h **(Read)**

7	6	5	4	3	2	1	0
SWD	SWC	SWB	SWA	TD	TC	TB	TA

SW(A-D) return the current state of the joystick switch inputs. T(A-D) return the current state of the four one-shot timers connected to the X and Y resistors of the dual joysticks.

PROGRAMMING THE ES1878

Identifying the ES1878

The ES1878 can be identified by reading Mixer Extension register 40h successively.

Mixer Extension register 40h returns the following 8-bit values on four successive reads:

18h, 78h, A[11:8], A[7:0]

where 18h and 78h are data reads indicating the part number (1878) and A[11:0] is the base address of the configuration device. Writing to the Mixer Address register (Audio_Base+4h) resets the sequence so that the next read of 40h returns 18h.

Resetting the Audio Device via Software

The ES1878 audio embedded processor can be reset in one of two ways: hardware reset or software reset. The hardware reset signal comes from the ISA bus. Software reset is controlled by bit 0 of port Audio_Base+6h.

To reset the ES1878 audio processor by software:

1. Write a 1 to port Audio_Base+6h.
2. Delay a short period, for example, by reading back Audio_Base+6h for 3 microseconds.
3. Write a 0 to port Audio_Base+6h.
4. In a loop that lasts from 1 to 10 milliseconds, poll port Audio_Base+Eh bit 7=1 for read data available.

   If bit 7=1, then read the byte from port Audio_Base+Ah. Exit loop if the content is 0AAh; otherwise, continue polling.

Both hardware reset and software reset will:

- Disable Extended Mode.
- Reset the timer divider and filter registers for 8 kHz sampling.
- Stop any DMA in progress.
- Clear any active interrupt request.
- Disable voice input of mixer (see the D1h/ D3h commands).
- Reset Compatibility Mode and Extended Mode DMA counters to 2048 bytes.
- Set analog direction to be DAC, with the DAC value set to mid-level.
- Set input volume for 8-bit recording with AGC to maximum.
- Set input volume for 16-bit recording to mid-range.

In addition to performing actions on the above list, a hardware reset will reset all Mixer registers to default values.

Modes of Operation

The ES1878 supports two modes of operation: Compatibility Mode and Extended Mode. The Compatibility Mode is compatible to the Sound Blaster Pro, and is the default mode after reset. In this mode, the ES1878 processor is an intermediary in all functions between the ISA bus and the CODEC Control register.

The Extended Mode of operation uses a 256-byte FIFO as an intermediary between the ISA bus and the CODEC Control register. The ES1878 processor is mostly idle in this mode. DMA control is handled by dedicated logic. New commands have been added to access the various control registers needed for extended operations. Some of these commands are also useful for Compatibility Mode, such as those configuring DMA and IRQ channels.

In both modes, a set of Mixer Control registers allows application software to control the analog mixer, record source, and output volume.

Compatibility Mode Programming

The following sections describe Compatibility Mode programming considerations.

Compatibility Mode DAC Operation

After reset, the analog circuitry is set up for DAC operations. Any ADC command will cause a switch to the ADC “direction,” and any subsequent DAC command will switch the ES1878 back to the DAC “direction.” The DAC output is filtered and connected to the voice input of the mixer. After reset, the voice input to the mixer is muted: to prevent pops. The ES1878 maintains a status flag called the Voice-Enable/Disable flag that indicates when the voice channel is muted. Use command D1h to enable the voice channel and command D3h to disable the voice channel.

If the ES1878 should not reset before playing a new sound, and the status of the analog circuits is not certain, mute the voice input to the mixer with command D3h, then set up DAC direction and level using the direct-to-DAC command:

10h + 80h

Wait 25 milliseconds for the analog circuitry to settle before enabling the voice channel with command D1h.

Pop sounds may still occur if the DAC level was left at a value other than mid-level (code 80h on an 8-bit scale) by the previous play operation. To prevent this, always finish a DAC transfer with a command to set the DAC level to mid-range:

10h + 80h

8-Bit, 16-Bit, and Compressed Data Formats

The 8-bit samples are unsigned, ranging from 0 to 0FFh, with the DC level around 80h.

16-bit samples are unsigned, least significant byte first, ranging from 0000h to 0FFFFh with the DC level around 8000h.

The ES1878 supports two types of compressed sound DAC operations: **ESPCM**®, which uses a variety of proprietary compression techniques developed by ESS Technology, and ADPCM, which is supported by other sound cards but is of a lower quality.

Both ADPCM and **ESPCM**® are only transferred using DMA transfer. The first block of a multiple-block transfer uses a different command than subsequent blocks. The first byte of the first block is called the reference byte.

Direct Mode DAC vs. DMA Mode DAC

In direct mode, the timing for DAC transfers is handled by the application program. For example, the system timer can be reprogrammed to generate interrupts at the desired sample rate. At each system timer interrupt, the command 10h (or 11h for 16-bit data) is issued followed by the sample. Polling of the Write-Buffer-Available flag is required before writing the command and between the command and the data.

Note: The switched capacitor filter is initialized by reset for an intended sample rate of 8 kHz. In direct mode, the application may wish to adjust this filter appropriate to the actual sample rate. The easiest way to do this is to program the timer with command 40h just as if the application were using DMA mode.

In DMA mode, the programmable timer in the ES1878 controls the rate at which samples are sent to the DAC. The timer is programmed using command 40h, which also sets up the programmable filters inside the ES1878. The ES1878 firmware maintains an internal FIFO (32 levels for 16-bit transfers, 64 levels for 8-bit transfers) that is filled by DMA transfers and emptied by the timed transfers to the DAC.

Before a DMA transfer, the application first programs the DMA controller for the desired transfer size and address, then programs the ES1878 with the same size information. At the end of the transfer, the ES1878 will generate an interrupt request, indicating that the current block transfer is complete. The FIFO gives the application program sufficient time to respond to the interrupt and initiate the next block transfer.

In “normal mode” DMA transfers, the DMA controller must be initialized and the ES1878 be commanded for every block that is transferred. In “auto-initialize mode”, the DMA transfer is continuous, in a circular buffer, and the ES1878 generates an interrupt for the transition between buffer halves. In this mode the DMA controller and ES1878 need to be set up only once.

The ES1878 supports mono 8-bit transfers to DAC at a rate up to 44 kHz. Mono 16-bit transfers are supported up to a rate of 22 kHz.

Stereo DAC Transfers in Compatibility Mode

Stereo DAC transfers are only available using DMA rather than direct mode commands.

To perform a stereo DAC transfer, first program bit 1 of Mixer register 0Eh to be high. Then set the timer divider to twice the per-channel sample rate. The maximum stereo transfer rate for 8-bit data is 22 kHz per channel; so for this case, program the timer divider as if you were doing 44 kHz mono. The maximum stereo transfer rate for 16-bit data is 11 kHz per channel.

For 8-bit data, the ES1878 expects the first byte transferred to be for the right channel, and subsequent bytes to alternate left, right etc.

For 16-bit data, the ES1878 expects the DMA transfers to be a multiple of 4, with repeating groups in the order:

1. left low byte
2. left high byte
3. right low byte
4. right high byte

Be sure to clear bit 1 of Mixer register 0Eh when you are done with the DAC transfer.

Compatibility Mode ADC Operation

The ES1878 analog circuitry is switched from the DAC direction to the ADC direction by the first direct or DMA mode ADC command. Discard the first 25-100 milliseconds of samples because pops might occur in the data due to the change from the DAC to ADC direction. In the ADC direction the voice input to the mixer is automatically muted.

The ES1878 has four recording sources: microphone, line, aux/CD, and mixer. Microphone input is the source after any reset. Select the source using the Mixer Control register 0Ch/1Ch.

The selected source passes through an input volume stage that can be programmed with 16 levels of gain from

0 to +22.5 dB in steps of 1.5 dB. In 8-bit recordings (other than “high-speed mode”), the volume stage is controlled by the ES1878 firmware for the purposes of automatic gain control (AGC). In 16-bit recordings as well as “high-speed mode” 8-bit recordings, the input volume stage is controllable from application software. Use command DDh to change the input volume level from 0 to 15. The reset default is mid-range, 8.

The ES1878 supports direct mode ADC, “normal mode” DMA for ADC, as well all “auto-initialize mode” DMA for ADC. The differences between the various types are described above for DAC.

Note: The switched capacitor filter is initialized by reset for an intended sample rate of 8 kHz. In direct mode, the application may wish to adjust this filter appropriate to the actual sample rate. The best way to do this is to program the timer with command 40h just as if the application were using DMA mode.

The maximum sample rate for direct mode ADC is 22 kHz.

The maximum sample rate for DMA ADC for both 8-bit and 16-bit is 22 kHz, using commands 24h, 25h, 2Ch, or 2Dh.

There is a special “high-speed mode” for ADC that allows 8-bit sampling up to 44 kHz. This mode uses commands 98h (“auto-initialization 99h (“normal”). No AGC is performed: The input volume is controlled with command DDh.

Sending Commands during Compatibility Mode DMA Operations

It is useful to understand the detailed operation of sending a command during a DMA operation.

The ES1878 has an internal 64-byte FIFO used for DMA to the DAC and from the ADC. When the FIFO is full (in the case of DAC, empty in the case of ADC), DMA requests are temporarily suspended and the Busy flag (bit 7 of port 22Ch) is cleared. This allows a window of opportunity to send a command to the ES1878. Use only commands such as D1h and D3h which control the mixer voice enable/disable status, and command D0h, which suspends (i.e., pauses) a DMA transaction.

The ES1878 chip sets the Busy flag when the command window is no longer open. Application software must send a command within 13 microseconds after the Busy flag goes high or the command will be confused with DMA data. This is normally easy to do if the polling is done with interrupts disabled.

As an example of sending a command during DMA, consider the case where the application desires to send command D0h in the middle of a DMA transfer. The application disables interrupts and polls the Busy flag. Because of the FIFO and the rules used for determining the command window, it is possible for the current DMA transfer to complete while waiting for the Busy flag to clear. In this event, the D0 command has no function, and there will be a pending interrupt request from the DMA completion.

This interrupt request can be cleared by reading port 22Eh before enabling interrupts or by signaling the interrupt handler that DMA is inactive so that it does not try to start a new DMA transfer.

Figure 14 shows timing considerations for sending a command.

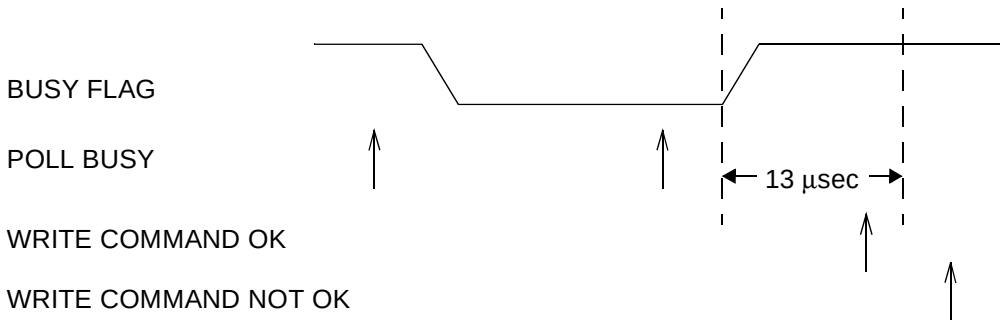


Figure 14 Command Transfer Timing

Extended Mode Programming

This section describes Extended Mode programming considerations.

Extended Mode registers are indirect registers, that is, they are written to and read from using commands sent to a port.

Mixing Modes Not Recommended

It is not recommended to mix Extended Mode commands with Compatibility Mode commands. The Voice-Enable/ Disable commands D1h and D3h are safe to use when using Extended Mode to process ADC or DAC. However, there are other Compatibility Mode commands that are likely to cause problems. The Extended Mode commands may be used to set up just the DMA or IRQ channels before entering the Compatibility Mode.

Accessing the ES1878 Extended Registers

This section describes how to send commands and command-related data to the ES1878's Extended registers, which are indirect registers.

Commands of the format Axh or Bxh are used to write to indirect registers within the ES1878.

After any reset, the command C6h must be issued before using any Extended Mode command.

Commands of the format Axh, Bxh, and Cxh, where x is a numeric value, are used for Extended Mode programming.

Commands of the format Ax and Bx are used to access the "internal" or *indirect* registers of the ES1878. For convenience, the registers are named after the commands used to access them. For example, "register A4h", the DMA Counter Low-Byte register, is written to by "command A4h."

Writing ES1878 Internal Registers

The following shows an example of writing to an ES1878 internal register. To set up the FIFO DMA Counter Reload register to F800h, send the following command/data bytes:

A4h, 00h; register A4 = 0
A5h, F8h; register A5=F8

Always check the write buffer before writing a command to port Audio_Base+Ch, to make sure it is not busy. Also, be sure to send command C6h after every reset if Extended Mode commands will be used.

Reading ES1878 Internal Registers

Command C0h is used to read the ES1878 Internal registers used for Extended Mode. Send command C0h followed by the register number, Axh or Bxh. For example, to read register A4h, send the following command bytes:

C0h, A4h

Then poll the Read Data Buffer Status bit, bit 7 of port Audio_Base+Eh, before reading the register contents.

Command/Data Handshaking Protocol

Writing Commands to the ES1878

Commands written to the chip enter a write buffer. Before writing the command, you must make sure the buffer is not busy.

Bit 7 of port Audio_Base+Ch is the ES1878 Busy flag. It is set when the write buffer is full or when the ES1878 is otherwise busy (for example, during initialization after reset or during Compatibility Mode DMA requests).

To write a command or data byte to the ES1878 processor:

1. Poll bit 7 of port Audio_Base+Ch for 5 to 10 milliseconds.
2. Write the command/data byte to port Audio_Base+Ch.

Note: The port Audio_Base+Ch write buffer is shared with Compatibility Mode DMA write operations. When DMA is active, the Busy flag will be cleared during time windows when a command can be received. Normally, the only commands that should be sent during DMA operations are 0Dxh commands: DMA pause/continue, voice enable/disable, etc. In this situation it is recommended that interrupts be disabled between the time that the Busy bit is polled and the command is written. Also, the time between these instructions should be minimized. For more information, see the section entitled 'Sending Commands During Compatibility Mode DMA Operations'.

Reading the Read Data Buffer of the ES1878

The Read-Data-Buffer-Status flag can be polled by reading bit 7 of port Audio_Base+Eh. When a byte is available the bit will be set high. Note that any read of port Audio_Base+Eh will also clear any active interrupt request from the ES1878. An alternative way of polling the read buffer status bit is via bit 6 of port Audio_Base+Ch, which is the same flag. The buffer status flag is cleared automatically by reading the byte from port Audio_Base+Ah.

Programming the ES1878 Mixer

The ES1878 has a set of Mixer registers that is backward compatible with the Sound Blaster Pro, but with an extended, alternate way of accessing the registers to provide for greater functionality.

There are two I/O addresses used by the mixer: Audio_Base+4h is the address port; Audio_Base+5h is the data port. In the Sound Blaster Pro, Audio_Base+4h is write only, while Audio_Base+5h is read/write. To set a Mixer register, write its address to Audio_Base+4h, then write the data to Audio_Base+5h. To read the register, read from Audio_Base+5h after setting the address into Audio_Base+4h.

The Mixer registers are not affected by software reset. To reset the registers to initial conditions, write any value to mixer address 0:

Write 0 to Audio_Base+4h (set mixer address to 0)
Write 0 to Audio_Base+5h (write 0 to address 0 to reset mixer)

The Sound Blaster Pro Mixer Volume controls are mostly 3 bits per channel. See the Sound Blaster Compatibility register 04h in Table 15 for details. Bits 0 and 4 are always high when read. The ES1878 offers an alternative way to write each Mixer register: If the address bit 4 is high, all 8 bits of the register are readable and writable. This is called “Extended Access.” If the address bit 4 is low, the interface is Sound Blaster Pro compatible, and bits 0 and 4 are cleared by a write and forced high on all reads.

The Sound Blaster Pro registers that have 3 bits per channel are listed below:

Register	Function	Extended Access Register for 4 Bits/Channel
04h	Voice Volume	14h
22h	Master Volume	32h
26h	FM Volume	36h
28h	CD (Aux) Volume	38h
2Eh	Line Volume	3Eh

For example, if 00h is written to Sound Blaster Pro register 04h, 11h is read back because bits 0 and 4 are “stuck high” on reads. Inside the register, these bits are “stuck low,” so that writing 00h is the same as writing 11h.

A write or read to address 14h instead of 04h allows for direct access to all 8 bits of the Mixer register.

Extended Access to Mic Mix Volume

If Sound Blaster Compatibility Mode register address 0Ah is used to control Mic Mix Volume, only bits 2 and 1 are significant. Bit 0 is stuck high on read and stuck low on writes. Furthermore, this is a mono control, panning is not supported.

For Extended Access, use register address 1Ah instead. This offers 4-bits/channel for pan control of the mono microphone input to the mixer. Refer to the Mic Preamp register in the ‘Extension Registers’ section.

Mic Mix Volume **1Ah**

7	6	5	4	3	2	1	0
Mic mix volume left				Mic mix volume right			

On reset, this register assumes the value of 00h.

Access to this register via address 0Ah is mapped as follows:

Write to 0Ah	D2=0, D1=0	Mic Mix Volume = 00h
	D2=0, D1=1	Mic Mix Volume = 55h
	D2=1, D1=0	Mic Mix Volume = AAh
	D2=1, D1=1	Mic Mix Volume = FFh
Read from 0Ah	D2 = Mic Mix Volume register bit 3	
	D1 = Mic Mix Volume register bit 2	
	D0 = 1	
	Others are undefined.	

Extended Access to ADC Source Select

In Sound Blaster Compatibility Mode in the Sound Blaster Pro mixer, there are three choices for recording source, set by bits 2 and 1 of Mixer register 0Ch. Note that bit 0 is set to 0 upon any write to 0Ch and set to one upon any read from 0Ch:

D2	D1	Source Selected
0	0	Microphone (default)
0	1	CD (Aux) Input
1	0	Microphone
1	1	Line input

For Extended Access, use register address 1Ch to select recording from the mixer as follows:

D2	D1	D0	Source Selected
x	0	x	Microphone (default)
0	1	x	CD (Aux) input
1	1	0	Line input
1	1	1	Mixer

Programming the FIFO for DMA Playback

Data Formats

There are eight formats available from the combination of the following three options as shown in Table 13.

- Mono or stereo
- 8-bit or 16-bit
- Signed or unsigned

For stereo data, the data stream always alternates channels in successive samples: first left, then right. For 16-bit data, the low byte always precedes the high byte.

Programming Steps for DMA Playback:

1. Reset.

   Write 3h to port Audio_Base+6h instead of 1h as in Compatibility Mode. Bit 1 high specifically clears the FIFO. The remainder of the software reset is identical to Compatibility Mode. After reset, command C6h is issued to enable Extended Mode commands. Reset disables the voice input to the mixer. This is intended to mask any pops created during the setup of the DMA transfer.

2. Program direction and type: registers B8h, A8h, and B9h:

   Register B8h: 0 for normal DAC transfer, 4 for auto-initialize DAC transfer.

   Register A8h: read this register first to preserve the bits. Modify only bits 1 and 0:
   Bits 1,0 = 10 = mono.

   Bits 1,0 = 01 = stereo.

   Register B9h:

   0 Single Transfer DMA.
   1 Demand Transfer DMA: 2 bytes per DMA request.
   2 Demand Transfer DMA: 4 bytes per DMA request.

3. Clocks and counters: registers A1h, A2h, A4h, and A5h:

   Register A1h = Sample Rate Clock Divider
   Register A2h = Filter Clock Divider
   Registers A4h/A5h = DMA Counter Reload register low/high, 2's complement

4. Initialize and Configure DACs: registers B6h and B7h.

   The DACs must be configured and initialized with a command sequence depending on the data format shown in Table 13.

5. Enable/Select DMA Channel and IRQ Channel, registers B1h and B2h:

   Register B1h – Interrupt Configuration register.
   Make sure bits 4 and 6 are high, clear bits 7 and 5.

   Register B2h – DRQ Configuration register.
   Make sure bits 4 and 6 are high; clear bits 7 and 5.

6. Configure system interrupt controller and DMA controller

Table 13 Command Sequences for DMA Playback

Mono	Stereo	8-bits	16-bits	Unsigned	Signed	Sequence
X		X		X		Reg B6 = 80h Reg B7 = 51h Reg B7 = D0h
X		X			X	Reg B6 = 00h Reg B7 = 71h Reg B7 = F0h
X			X	X		Reg B6 = 80h Reg B7 = 51h Reg B7 = D4h
X			X		X	Reg B6 = 00h Reg B7 = 71h Reg B7 = F4h
	X	X		X		Reg B6 = 80h Reg B7 = 51h Reg B7 = 98h
	X	X			X	Reg B6 = 00h Reg B7 = 71h Reg B7 = B8h
	X		X	X		Reg B6 = 80h Reg B7 = 51h Reg B7 = 9Ch
	X		X		X	Reg B6 = 00h Reg B7 = 71h Reg B7 = BCh

7. Recommended: Delay approximately 100 milliseconds before enabling voice input to mixer.
8. Enable voice to mixer with command D1h.
9. To Start DMA:

   Set bit 0 of register B8h high while preserving all other bits.

10. During DMA:

    For auto-initialize, do not send any commands to the ES1878 at interrupt time, except for reading Audio_Base+Eh to clear the interrupt request.

    For normal mode, initialize the system DMA controller with the address and count of the next block to transfer. Update the ES1878 transfer count registers if the count is changed. To start the next transfer, clear bit 0 of register B8h, then set it high again.

    To stop a DMA transaction in progress, clear bit 0 of register B8h. To stop DMA after the current auto-initialize block is finished, clear bit 2 of register B8h, wait for the interrupt, and then clear bit 0 of the B8h.

11. After the DMA transaction is finished, restore the system interrupt controller and DMA controller to their idle state. Monitor the FIFO Empty status flag in register Audio_Base+Ch to be sure data transfer is completed. A delay of 25 milliseconds is required to let

the filter outputs settle to DC levels, then mute the voice input to the mixer with command D3h.

12. Finally, issue another software reset to the ES1878 to initialize the appropriate registers.

Programming the FIFO for DMA Record

Data Formats

There are eight formats available from the combination of the following three options:

- Mono or stereo
- 8-bit or 16-bit
- Signed or unsigned

For stereo data, the data stream always alternates channels in successive samples: first left, then right. For 16-bit data, the low byte always precedes the high byte.

No Automatic Gain Control for 8-bit Recordings

In Extended Mode, there is no Automatic Gain Control (AGC) performed while recording. If AGC is necessary, use 16-bit recordings and perform AGC in system software.

Programming Steps for DMA Recording:

1. Reset.

   Write 3h to port Audio_Base+6h instead of 1h as in Compatibility Mode. Bit 1 high specifically clears the FIFO. The remainder of the software reset is identical to Compatibility Mode. After reset, command C6h is issued to enable Extended Mode commands.

2. Select the input source using the Mixer register 0Ch.
3. Program Input Volume register B4h.
4. Program direction and type: registers B8h, A8h.
   a. Register B8h: 0Ah for normal ADC transfer, 0Eh for auto-initialize ADC transfer.

      At this point, the direction of the analog circuits becomes ADC rather than DAC. Unless the recording monitor is enabled, there will be no output from AOUT_L or AOUT_R until the direction is restored to DAC.

   b. Register A8h: read this register first to preserve the bits. Modify only bits 3, 1, and 0:

      Bits 1,0 = 1,0 for mono and 0,1 for stereo.

      Bit 3 = 0 Disable record monitor for now.

   c. Register B9h:
      0 = Single Transfer DMA.
      1 = Demand Transfer, 2 bytes per DMA request.
      2 = Demand Transfer, 4 bytes per DMA request.
5. Clocks and counters: registers A1h, A2h, A4h, and A5h.
   a. Register A1h = Sample Rate Clock Divider. Set bit 7 high for sample rates greater than 22 kHz.
   b. Register A2h = Filter Clock Divider.
   c. Registers A4h/A5h = DMA Counter Reload Register low/high, 2's complement.
6. Delay 100 milliseconds to allow the analog circuits to settle.
7. Enable record monitor if desired:

   Register A8h Bit 3=1: Enable Record Monitor (optional).

8. Initialize and Configure ADC: register B7h.

   The ADCs must be configured and initialized with a command sequence depending on the data format shown in Table 14.

Table 14 Command Sequence for DMA Record

Mono	Stereo	8-bits	16-bits	Unsigned	Signed	Sequence
x		x		x		Reg B7 = 51h Reg B7 = D0h
x		x			x	Reg B7 = 71h Reg B7 = F0h
x			x	x		Reg B7 = 51h Reg B7 = D4h
x			x		x	Reg B7 = 71h Reg B7 = F4h
	x	x		x		Reg B7 = 51h Reg B7 = 98h
	x	x			x	Reg B7 = 71h Reg B7 = B8h
	x		x	x		Reg B7 = 51h Reg B7 = 9Ch
	x		x		x	Reg B7 = 71h Reg B7 = BCh

9. Enable/select DMA Channel and IRQ Channel registers B1h and B2h:
   a. Register B1h: Interrupt Configuration register.
   b. Verify that bits 4 and 6 are high. Clear bits 7 and 5.
   c. Register B2h: DRQ Configuration register:
   d. Verify that bits 4 and 6 are high. Clear bits 7 and 5.
10. Configure system interrupt controller and DMA controller.
11. To start DMA: Set bit 0 of register B8h high. Leave other bits unchanged.
12. During DMA:

    For auto-initialize, do not send any commands to the ES1878 at interrupt time, except for reading Audio_Base+Eh to clear the interrupt request.

    For normal mode, initialize the system DMA controller with the address and count of the next block to transfer. Update the ES1878 Transfer Count registers if the count is changed. To start the next transfer, clear bit 0 of register B8h, then set it high again.

To stop a DMA transaction in progress, clear bit 0 of register B8h. To stop a DMA transaction after the current auto-initialize block is finished, clear bit 2 of register B8h, wait for the interrupt, and then clear bit 0 of the B8h.

13. After a DMA transaction is finished, restore the system interrupt controller and DMA controller to their idle state.
14. Finally, issue another software reset to the ES1878 to initialize the appropriate registers. This will return the ES1878 to the DAC direction and turn off the record monitor.

Programming the FIFO for I/O Block Transfer

For some applications, DMA is not suitable or available for data transfer, and it is not possible to take exclusive control of the system for DAC and ADC transfers. In these situations, use I/O block transfers within an interrupt handler. The REP OUTSB instruction of the 80x86 family transfers data from memory to an I/O port specified by the DX register. The REP INSB instruction is the complementary function. Use ES1878 port Audio_Base+Fh for block transfers.

I/O transfers to FIFO are nearly identical to the DMA process described above, except that an I/O access to port Audio_Base+Fh replaces the DMA cycle. Some differences are described here.

The DRQ Control Register B2h, bits 7:5 should all be low. This is because no actual DRQ/DACKB cycle is needed.

The IRQ Control Register B1h must have bit 5 high to enable an interrupt on FIFO half-empty transitions. Bit 6 should be low to avoid an interrupt generated by the DMA counter.

To program in this mode, it is useful to understand how the FIFO Half-Empty flag generates an interrupt request: An interrupt request is generated on the rising edge of the FIFO Half-Empty flag. This flag can be polled by reading port Audio_Base+Ch. The meaning of this flag depends on the direction of the transfer:

DAC FIFOHE flag is set high if 0-127 bytes in FIFO.

ADC FIFOHE flag is set high if 128-256 bytes in FIFO.

For DAC, an interrupt request is generated when the number of bytes in the FIFO changes from $\geq$ 128 to < 128. This indicates to the system processor that 128 bytes can be safely transferred without overfilling the FIFO. Before the first interrupt can be generated, the FIFO needs to be primed, or filled, with more than 128 bytes. Keep in mind that data may be taken out of the FIFO while it is being filled by the system processor. In this case, there may never be $\geq$ 128 bytes in the FIFO unless more than 128 bytes is transferred. One solution is to poll the ES1878 FIFOHE flag to be sure it goes low in the interrupt handler (or when priming the FIFO) and perhaps send a second block of 128 bytes.

For ADC, the interrupt request is generated when the number of bytes in the FIFO changes from < 128 to $\geq$ 128, indicating that the system processor can safely read 128 bytes from the FIFO. Before the first interrupt can be generated, the FIFO should be emptied (or mostly so) by reading from Audio_Base+Fh and polling the FIFOHE flag. It is not safe to indiscriminately use the FIFO reset, bit 1, of port Audio_Base+6h to clear the FIFO, because it may get ADC data out-of-sync.

As in DMA mode, bit 0 of register B8h enables transfers between the system and the FIFO inside the ES1878.

Note: The ES1878 is designed for I/O block transfer up to an ISA bus speed of 8.33 MHz.

Full-Duplex DMA Mode (No DSP Serial Port)

The ES1878 supports monophonic full-duplex DMA. In FD mode, the left DMA channel records while the right channel plays back. To support FD mode, a second DMA channel has been added to the ES1878. This second DMA channel is programmed through Mixer Extension registers.

In FD mode, the first DMA channel (the old ES1788 DMA channel) is programmed for a mono recording in the same manner as is usually done. Extended Mode registers A1h and A2h define the sample rate and filter frequency for both record and playback. In other words, the record and playback must be at the same sample rate (synchronous).

After the first DMA channel is set up, the new, second DMA channel is programmed. Mixer Extension registers 74h and 76h are set to the 2's complement DMA transfer count. The second DMA channel supports auto-initialize mode as well as normal mode. The playback buffer in system memory does not have to be the same size as the record buffer. When the DMA transfer count rolls over to 0, it can generate an interrupt that is independent of the interrupt generated by the first DMA channel.

If the record and playback buffers are the same size, then a single interrupt can be used. The DMA Transfer Count registers are programmed with the same value for both channels. The second DMA channel should be enabled first, before the record channel. For example, assume there are two half-buffers in a circular buffer. When the record channel completes filling the first half, it will generate an interrupt. It is necessary to assure that the playback channel is not still accessing the first half at the time of the interrupt. This can be guaranteed by starting the playback channel first. It has a 32-word FIFO that will be filled quickly by DMA.

The recommended method is as follows:

Program both DMA controllers for auto-initialize DMA within separate circular buffers of the same size, N.

- Program the record DMA channel for monophonic, 16-bit recording, auto-initialize mode, but don't set bit 7 of Extended register B7h or bit 0 of Extended register B8h at this time. This includes setting registers A1h and A2h to define the sample rate and filter frequency, as well as programming the 2's complement of the half-buffer-size (N/2) into Extended registers A4h and A5h.
- Program the playback DMA channel for monophonic, 16-bit playback, auto-initialize mode. Set the 2's complement transfer count for 64 bytes. Since the second channel is in auto-initialize mode, and the second channel interrupt is not being used, any value can be used for the transfer count. Sixty-four bytes allows the playback channel to get a “head-start” on the record channel by polling the second channel interrupt request bit after starting the second channel DMA.
- Before starting the second channel DMA, clear the second channel interrupt request bit by writing a 0 to bit 7 of Mixer Extension register 7Ah.
- Enable full-duplex mode by setting bit 0 of Mixer Extension register 78h. Since the playback FIFO is presumably empty, the value 0 is transferred to the playback DAC at each sample clock. A click or pop may be heard when full-duplex mode is enabled. To prevent this, use the D1 command to enable the DAC input to the mixer after a suitable delay (about 25 milliseconds).
- Enable playback DMA by setting bit 1 of Mixer Extension register 78h. After 64 bytes are transferred, bit 7 of 7Ah should go high. Poll this bit with a suitable time-out (for example, 10 milliseconds).
- After bit 7 of 7Ah goes high, enable recording by setting bit 7 of Extended Mode register B7 and bit 0 of Extended Mode register B8h.
- As usual, the first 50 to 100 milliseconds of recorded data should be discarded until analog circuits have settled.

To exit full-duplex mode, clear bits 0 and 1 of Mixer Extension register 78h.

I²S Serial Interface Software Enable

By hardware reset default, the I²S interface is disabled. Bit 6 of Vendor-Defined Card-Level register 29h enables the I²S interface when it is set high. This register is accessed through the Configuration Device.

I²S DAC Volume Control

The I²S DAC has its own mixer volume control register – mixer register 68h. This register is reset to zero (mute) by hardware reset. Bits 7:4 select the left channel volume, and bits 3:0 select the right channel volume.

Alternatively, the I²S DAC volume can be set by the volume of the FM DAC at register 36h. This is useful when the I²S interface is used for an external wavetable synthesizer. Bit 3 of Vendor-Defined Card-Level register 2Bh when set high will enable tracking with FM volume.

AUDIO REGISTERS

Types of Register Access

There are two types of audio registers in the ES1878:

- Mixer registers.

  These registers are accessed through I/O ports Audio_Base+4h and Audio_Base+5h. Audio_Base+4h is written with the register address. The register can then be read/written via Audio_Base+5h. These registers control many functions other than the mixer.

- Extension or Extended Mode registers

  These registers are used to control Extended Mode DMA playback and record through the first DMA channel. Extended Mode registers are accessed by an extension to the Sound Blaster common interface. This interface uses I/O ports Audio_Base+Ah, Audio_Base+Ch, and Audio_Base+Eh to transfer read data, write data/commands, and write status respectively.

  Examples of this are reading default settings and disabling the mic preamp at register A9h.

  1 Reading default settings

  Write Audio_Base+Ch to register C6h
  Write Audio_Base+Ch to register C0h.
  Write Audio_Base+Ch to register A9h.
  Read Audio_Base+Ah

  2 Disabling Mic preamp

  Write Audio_Base+Ch to register A9h
  Write a 0 to Audio_Base+Ch bit 2.

Mixer Registers

Sound Blaster Compatible Mixer Registers

For more information, see Sound Blaster Pro Technical Reference.

Table 15 Sound Blaster Compatibility Registers

Reg	D7	D6	D5	D4	D3	D2	D1	D0	Remark
00h	Write: reset mixer								Mixer reset
02h									
04h	DAC play volume left			x	DAC play volume right			x	Digital audio playback volume
06h									
08h									
0Ah	x	x	x	x	x	Mic mix volume		x	
0Ch	x	x	F1[a]	x	F0[a]	ADC Source		x	See note for F0, F1.
0Eh	x	x	F2[a]	x	x	x	Stereo	x	See note for F2.
20h									
22h	Master volume left			x	Master volume right			x	
24h									
26h	FM volume left			x	FM volume right			x	
28h	CD (AUXA) volume left			x	CD (AUXA) volume right			x	
2Ah									
2Ch									
2Eh	Line volume left			x	Line volume right			x	

a. Sound Blaster filter control bits F2, F1, and F0 have no function in the ES1878 and are ignored.

Sound Blaster Pro Master Volume Emulation

Sound Blaster Pro emulations for master volume means that the 6-bit volume counters can be written via the Sound Blaster Pro Mixer register 22h (or 32h). Sound Blaster Pro emulation is enabled by default, and can be disabled by setting bit 0 of Mixer register 62h.

The master volume registers 60h and 62h can always be read, regardless of whether Sound Blaster Pro emulation is enabled, using the Sound Blaster Pro Mixer registers 22h (and 32h). The following 6-bit to 4-bit translation table is used:

Table 16 Sound Blaster Pro Master Volume Emulation

Mute	Master Volume	Value Read at 32h	Value Read at 22h
1	xx	0	1
0	0-24	1	1
0	25-30	2	3
0	31-34	3	3
0	35-38	4	5
0	39-42	5	5
0	43-46	6	7
0	47-50	7	7
0	51-54	8	9
0	55	9	9
0	56-57	10	11
0	58	11	11
0	59-60	12	13
0	61	13	13
0	62	14	15
0	63	15	15

Filter Control Registers

The Sound Blaster Pro mixer has three bits that control input and output filters. They are labeled as F0, F1, and F2 in Table 15 and Table 17. They have no function in the ES1878 and their values are ignored.

Mixer Stereo Control Bit

The mixer stereo control bit is bit 1 of register 0Eh. It is normally zero. Set this bit high to enable Sound Blaster Pro compatible stereo DAC functions, and program the DAC sample rate to be twice the sample rate of each channel. For example, for 22 kHz stereo, program the “sample rate” to be 44 kHz using command 40h.

This bit enables stereo only for DMA transfer to the DAC in Compatibility Mode. It should not be used in Extended Mode.

After any write to Mixer register 0Eh, the ES1878 will expect the next 8-bit DMA sample to be for the right channel. Subsequent samples will then alternate left, right, left, right.

For 16-bit stereo DMA to DAC, the data is always expected to be in the order:

left low, left high, right low, right high

and DMA transfers should be in multiples of four bytes.

Clear this bit after completing the stereo DMA transfer, because this bit is unaffected by software reset (only mixer reset).

ESS Mixer Registers

Table 17 ESS Mixer Registers Summary

Reg	D7	D6	D5	D4	D3	D2	D1	D0	Remark
00h	Write: reset mixer								
14h	DAC play volume left				DAC play volume right				
1Ah	Mic mix volume left				Mic mix volume right				Mic mix volume
1Ch	X	X	F1[a]	X	F0[a]	ADC source			
1Eh	X	X	F2[a]	X	X	X	Stereo	X	
32h	Master volume left				Master volume right				
36h	FM volume left				FM volume right				
38h	CD (AuxA) volume left				CD (AuxA) volume right				
3Ah	AuxB volume left				AuxB volume right				
3Ch						PC speaker volume			
3Eh	Line volume left				Line volume right				
40h	ES1878 identification value (read only)								Identifies the ES1878
42h	Input over-ride	Mic 0 dB	IS1	IS0	Input volume				Serial mode input control
44h	Output override	Output signal			Output volume				Serial mode output control
46h	Analog control override	0	Left ADC	Right ADC	AC1	AC0	FDXO enable	FDXI enable	Serial mode miscellaneous analog control
48h	SW SE	2's comp	Serial reset	Enable ES689/ ES690 Intfc	Active low sync	0	Mono FDXI/O	Interleave Mode	Serial mode miscellaneous control • Wavetable input enable • FDXI/FDXO enable
4Ch	Filter over-ride	0	0	0	2's complement filter divider				Serial mode filter divider control
4Eh	TX SRC 1	TX SRC 0	TX 16/8	TX stereo/ mono	RX SRC 1	RX SRC 0	RX 16/8	RX stereo/ mono	Serial mode format/source/target control
60h	0	1:Mute	Left master volume						Left master volume counter value
62h	0	1:Mute	Right master volume						Right master volume counter value
64h	2-wire mode	MPU-401 int mask	X	Read-only HMV int request	X	X	HMV int mask	1: Disable SB Pro master volume emulation	Master volume control
66h	Clear hardware volume interrupt request (write-only)								
68h	Left I^2S volume				Right I^2S volume				I^2S volume control
6Ah	Mappable volume register value								
74h	Two's complement transfer count – low byte								Second DMA transfer count reload register
76h	Two's complement transfer count – high byte								
78h	Single/demand transfer		0	1: Auto-initialize	0	0	1:Enable second channel DMA	1:Enable full-duplex mode	Second DMA control 1
7Ah	Second channel IRQ	IRQ mask	0	0	0	1: Signed	1: Stereo 0: Mono	1: 16-bit 0: 8-bit	Second DMA control 2

a. Sound Blaster filter control bits F2, F1, and F0 have no function in the ES1878 and are ignored.

Register Detailed Descriptions

Reset Mixer Register **00h**

7	6	5	4	3	2	1	0
Write: reset mixer							

DAC Play Volume Register **14h**

7	6	5	4	3	2	1	0
DAC play volume left				DAC play volume right			

On reset, this register assumes the value of 88h.

Mic Mix Volume Register **1Ah**

7	6	5	4	3	2	1	0
Mic mix volume left				Mic mix volume right			

On reset, this register assumes the value of 00h.

Record Source Select Register **1Ch**

7	6	5	4	3	2	1	0
X	X	F1*	X	F0*	ADC source		

On reset, this register assumes the value of 00h.

* Sound Blaster filter control bits F1 and F0 have no function in the ES1878 and are ignored.

For extended access, use register address 1Ch to select recording from the mixer as follows:

D2	D1	D0	Source Selected
x	0	x	Microphone (default)
0	1	x	CD (Aux) input
1	1	0	Line input
1	1	1	Mixer

Stereo Flag **1Eh**

7	6	5	4	3	2	1	0
X	X	F2*	X	X	X	Stereo	X

On reset, this register assumes the value of 00h.

* Sound Blaster filter control bit F2 has no function in the ES1878 and is ignored.

Master Volume Register **32h**

7	6	5	4	3	2	1	0
Master volume left				Master volume right			

On reset, this register assumes the value of 88h.

This register provides backward compatible access to master volume. New applications can also use registers 60h and 62h which have more resolution.

FM Volume Register **36h**

7	6	5	4	3	2	1	0
FM volume left				FM volume right			

On reset, this register assumes the value of 88h.

CD (AuxA) Volume Register **38h**

7	6	5	4	3	2	1	0
CD (AuxA) volume left				CD (AuxA) volume right			

On reset, this register assumes the value of 00h.

AuxB Volume Register **3Ah**

7	6	5	4	3	2	1	0
AuxB volume left				AuxB volume right			

On reset, this register assumes the value of 00h.

PC Speaker Volume Register **3Ch**

7	6	5	4	3	2	1	0
reserved					PC speaker volume		

On reset, this register assumes the value of 04h.

Line Volume Register **3Eh**

7	6	5	4	3	2	1	0
Line volume left				Line volume right			

On reset, this register assumes the value of 00h.

ES1878 Identification Register **40h**

7	6	5	4	3	2	1	0
ES1878 identification value							

The ES1878can be identified by reading Mixer Extension register 40h successively.

Mixer Extension register 40h returns on successive reads:

18h, 78h, A[11:8], A[7:0]

where 18h and 78h are data reads indicating the part number (1878) and A[11:0] is the base address of the configuration device. Writing to the Mixer Address register (Audio_Base+4h) resets the sequence so that the next read of 40h returns 18h.

Serial Mode Input Control **42h**

7	6	5	4	3	2	1	0
Input override	Mic 0 dB	IS1	IS0	Input volume			

Bit 7 1 = IS1/IS0 and input volume replace normal values as programmed by the application when the ES1878 is in serial mode.

0 = IS1/IS0 and input volume are unchanged during serial mode.

Bit 6 1 = the 26 dB microphone preamp is bypassed during serial mode (if bit 7 is high).

Bits 5:4 IS1/IS0 select the input source during serial mode if bit 7 is high. These values override the normal mixer settings as shown in the following:

IS1	IS0	Input source
0	0	LINE
0	1	AUXA (CD)
1	0	Microphone
1	1	Mixer

Bits 3:0 Input volume. If bit 7 is high during serial mode, this value overrides the input volume settings set by command B4h.

Serial Mode Output Control 44h

7	6	5	4	3	2	1	0
Output override			Output signal			Output volume	

Bit 7 1 = Output volume during serial mode is from this register rather than from the Mixer Master Volume register. Output signal control is always in force during serial mode regardless of the state of this bit.

Bits 6:4 Controls the signal routed to speaker outputs AOUT_L and AOUT_R.

6	5	4	Signal
0	0	0	Mute
0	0	1	FDXI monitor in both channels
0	1	0	FDXO monitor in both channels
0	1	1	FDXI monitor in left channel, FDXO in right
1	0	0	Mixer output
1	0	1	Mixer output except DAC playback
1	1	0	Mixer output except DAC playback & FM
1	1	1	Reserved

Bits 3:0 Output volume. Replaces normal mixer master volume setting if bit 7 is high during serial mode.

Serial Mode Miscellaneous Analog Control 46h

7	6	5	4	3	2	1	0
Analog control override	0	Left ADC	Right ADC	AC1	AC0	FDXO enable	FDXI

Bit 7 1 = bits 6:0 of this register take effect during serial mode.
0 = bits 6:0 do not ever take effect.

Bit 6 Reserved. Should be set to 0.

Bit 5 1 = Left channel combined ADC and DAC is in ADC mode.
0 = Left channel combined ADC and DAC is in DAC mode.

Bit 4 1 = Right channel combined ADC and DAC is in ADC mode.
0 = Right channel combined ADC and DAC is in DAC mode.

Bits 3:2 Analog control bits 1:0. These special control signals control interconnections in the analog circuitry. They should be set appropriately for the application as follows:

AC1	AC0	Application
0	0	Stereo wave playback or record.
1	1	Mono wave playback or record.
1	0	Full-duplex (mono record and playback).

Bit 1 1 = Enables FDXO output connection to output pin FOUT_R (right channel filter output).
0 = FDXO has pull-up to CMR.

Bit 0 1 = Enables FDXI input connection from left channel filter input and thus to the input of the left channel ADC.
0 = FDXI input has pull-up to CMR. The left channel filter input and ADC comes from the input volume stage as usual.

Serial Mode Miscellaneous Control 48h

7	6	5	4	3	2	1	0
SW SE	2's comp	Serial reset	Enable ES689/ ES690 intfc	Active low sync	0	Mono FDXI/O	Inter-leave Mode

Bit 7 1 = Force serial mode. This pin is synchronized by input clock DCLK, which must be running.

Bit 6 1 = Data format is 2's complement.
0 = Data format is unsigned.

Bit 5 1 = Reset Serial register left/right toggle flags.
0 = Release reset.

Serial reset also inhibits FDXO connection to FOUT_R and “zeros” all shift registers.

Bit 4 1 = Enable ES689/ES690 to acquire FM DACs when serial activity present on pins MCLK and MSD.
0 = Prevent ES689/ES690 from acquiring FM DACs.

Bit 3 1 = Sync pulses (FSR, FSX) are active low.

Bit 2 Reserved. Always writes 0.

Bit 1 1 = Enable mono FDXI/FDXO mode.

Bit 0 1 = Enable Interleave Mode.

Serial Mode Filter Divider **4Ch**

7	6	5	4	3	2	1	0
Filter override	0	0	0	2's complement filter divider			

Controls the filter clock rate during serial mode.

Bit 7 1 = During serial mode, the filter clock is generated by dividing down the serial clock.
0 = During serial mode, the filter clock is generated as usual.

Bit 6 Reserved. Always write 0.

Bit 5 Reserved. Always write 0.

Bit 4 Reserved. Always write 0.

Bits 3:0 These bits are a 2's complement value that divides the serial clock. The ratio of the filter -3 dB frequency to the filter clock is about 1:41.

Examples:

02h (-14) External Serial Clock 2.048 MHz / 14 / 41 = 3568 Hz for 8000 Hz sample rate.

0Eh (-2) Internal Serial Clock 1.591 MHz / 2 / 41 = 19.4 kHz for 44,100 Hz sample rate. Note that the sample rate divider is an integer multiple of the filter divider for 44,100, which gives maximum performance of DACs and ADCs.

Serial Mode Format/Source/Target **4Eh**

7	6	5	4	3	2	1	0
TX SRC 1	TX SRC 0	TX 16/8	TX stereo/ mono	RX SRC 1	RX SRC 0	RX 16/8	RX stereo/ mono

Bits 7:6 Transmit register source.

7	6	Source
0	0	None: Transmit register held at "zero" code
0	1	FIFO
1	0	Left ADC or stereo ADC transmission
1	1	Right ADC

Bit 5 1 = Transmit length is 16 bits, unsigned.
0 = Transmit length is 8 bits, unsigned.

Bit 4 1 = Transmit mode is stereo. Left and right channels alternate, with left channel data preceding right channel data.
0 = Transmit mode is mono.

Bits 3:2 Receive register target.

7	6	Source
0	0	None: Receive register held at "zero" code
0	1	FIFO
1	0	DAC (if mono, right channel receives data, left channel receives complement of data)
1	1	FM DAC (if mono, right channel receives data, left channel receives complement of data)

Bit 1 1 = Receive length is 16 bits, unsigned.
0 = Receive length is 8 bits, unsigned.

Bit 0 1 = Receive mode is stereo. Left and right channels alternate, with left channel data preceding right channel data.
0 = Receive mode is mono.

Left Master Volume Counter Value **60h**

7	6	5	4	3	2	1	0
0	1: MUTE	Left master volume					

Bits 5:0 select the attenuation level in steps of -1.5 dB. The maximum setting of 3Fh corresponds to 0 dB attenuation.

Right Master Volume Counter Value **62h**

7	6	5	4	3	2	1	0
0	1: MUTE	Right master volume					

Bits 5:0 select the attenuation level in steps of -1.5 dB. The maximum setting of 3Fh corresponds to 0 dB attenuation.

Master Volume Control **64h**

7	6	5	4	3	2	1	0
2-wire mode	MPU-401 interrupt mask	x	Read-only HMV interrupt request	x	x	HMV interrupt mask	1: Disable SB Pro master volume control

Bit 7 When set high, having both UP and DOWN inputs low will produce act as a MUTE input low.

Bit 6 This bit is AND'ed with the MPU-401 interrupt request. If it is low, the MPU-401 interrupt request stays low. This bit is cleared by hardware reset.

Bit 4 Read-only interrupt request from hardware volume event.

Bit 1 This bit is AND'ed with the hardware volume interrupt request. This bit is cleared by hardware reset.

Bit 0 When low, a write to the Sound Blaster Pro Master Volume register will be translated into a write to the hardware master volume counters, Mixer registers 60h and 62h. If high, the Sound Blaster Pro Master Volume registers are, in effect, read-only. This bit is cleared by hardware reset.

Clear Hardware Volume Interrupt Request 66h

Any write to this register resets the hardware volume interrupt request.

I²S Volume Control 68h

7	6	5	4	3	2	1	0
Left I²S volume				Right I²S volume			

This register is reset to zero by hardware or Mixer reset.

Normally, this Mixer register controls the volume of the I²S DAC in the ES1878 as well as the I²S DAC in the ES978 when docked. However, either or both of these DACs can be controlled by the FM Volume register, rather than by this register. Bits 3 and 4 of PnP Configuration register 2Bh select the volume control for the I²S DACs.

Mappable Volume Register 6Ah

7	6	5	4	3	2	1	0
Mappable volume							

This register is reset to zero by hardware or Mixer reset.

This Mixer register can be assigned to control any single mixer input of the ES978. Normally, each mixer input of the ES978 is slaved to a Mixer Volume register of the ES1878, so that both sources (in the ES978 and the ES1878) are controlled using the same Mixer register. Instead, a selected mixer input of the ES978 can be controlled independently using Mixer register 6Ah.

Bits [2:0] of PnP Configuration register 2Bh assign the Mappable Volume register to a mixer input of the ES978.

Second DMA Transfer Count Reload Register 74h

7	6	5	4	3	2	1	0
2's complement transfer count – low byte							

Second DMA Transfer Count Reload Register 76h

7	6	5	4	3	2	1	0
2's complement transfer count – high byte							

Second DMA Control 78h

7	6	5	4	3	2	1	0
Single/ demand transfer		0	Auto-init		0	2nd DMA channel enable	Full-duplex enable

Bits 7:6
0 0 Single transfer: 1 DACK per DRQ
0 1 Demand transfer: 2 DACKs per DRQ
1 0 Demand transfer: 4 DACKs per DRQ
1 1 Demand transfer: 8 DACKs per DRQ

Bit 5 Reserved. Write 0.

Bit 4 1 = Auto-initialize mode. After the transfer counter rolls over to 0, it is automatically reloaded and DMA continues. The second channel interrupt flag will be set high.

0 = Normal mode. After the transfer counter rolls over to 0, it is reloaded but DMA stops. Bit 1 of this register is cleared. The second channel interrupt flag will be set high.

Bits 3:2 Reserved. Write 0.

Bit 1 When high, second channel DMA is enabled. This bit is cleared when the transfer counter rolls over to 0 if not in auto-initialize mode.

Bit 0 When high, full-duplex mode is enabled. The left channel is used for recording, and the right channel is used for playback.

This register is reset to zero by hardware or software reset by bit 0 of port Audio_Base+6h.

Second DMA Control 2 7Ah

7	6	5	4	3	2	1	0
Interrupt request latch		reserved			Data sign	Stereo/ mono	16-bits/ 8-bits

This register is reset to zero by hardware or software reset.

Bit 7 Second DMA channel Interrupt Request Latch. This latch is set high when the DMA counter rolls over to 0, or when a 1 is written to this bit. The latch is cleared by writing a 0 to this bit or by hardware or software reset.

Bit 6 This bit is AND'ed with bit 7 to produce the second DMA channel interrupt request.

Bits 5:3 Reserved, write 0.

Bit 2 1 = Data is in signed, 2's complement format.
0 = Unsigned data.

Bit 1 1 = Stereo data. This format is reserved for Interleave Mode when using the DSP serial interface.
0 = Mono data.

Bit 0 1 = 16-bit samples.
0 = 8-bit samples.

Extension Registers

Table 18 Audio Processor Extension Registers

Reg	D7	D6	D5	D4	D3	D2	D1	D0	Remark
A1h	1: > 22 kHz 0: <= 22 kHz	Extended Mode sample rate divider							S/W reset, unknown
A2h	Filter clock divider								S/W reset, setup for 8 kHz sampling
A4h	DMA transfer counter reload – low byte								
A5h	DMA transfer counter reload – high byte								
A9H	Reserved					Enable mic preamp	Reserved		
B1h	Processor IRQ	Enable IRQ on DMA counter overflow	Enable IRQ on FIFOHE change	x	IS1, IS0		IS1, IS0		Interrupt control
B2h	Processor DRQ	Enable DRQ for extended DMA	Enable DRQ for Compatibility Mode DMA	Enable pull-down inactive DRQ	DS1 register read-only	DS0 register read-only	x	x	DMA control
B4h	Input volume right				Input volume left				Record volume
BAh	0	1: Enable SR adjust	1: Disable time delay on analog wake-up	Sign	Adjust magnitude				
BBh	0			Sign	Adjust magnitude				
BCh	FM DAC volume 1	FM DAC volume 0	1	1	0	1	1	0	

Extension Register Descriptions

Extended Mode Sample Rate Generator **A1h**

7	6	5	4	3	2	1	0
1: > 22 kHz 0: <= 22 kHz	Sample rate divider						

This register should be programmed for the sample rate for all DAC and ADC operations in Extended Mode.

The clock source for sample rate generator is 397.7 kHz if bit 7 is 0 and 795.5 kHz if bit 7 is 1.

The sample rate is determined by the two's complement divider in bits 7:0:

Sample_Rate = 397.7 kHz / (128-x) if bit 7 = 0.
= 795.5 kHz / (256-x) if bit 7 = 1.
where: x = value in bits 7:0 of register A1h.

Filter Divider **A2h**

7	6	5	4	3	2	1	0
Filter clock divider							

This register controls the low-pass frequency of the switch-capacitor filters inside the ES1878. Generally, the filter roll-off should be positioned at 80% - 90% of the Sample_Rate/ 2 frequency. The ratio of the roll-off frequency to the filter clock frequency is 1:82. In other words, first determine the desired roll-off frequency by taking 80% of the Sample_Rate divided by 2, then multiply by 82 to find the desired Filter Clock frequency. Use the formula below to determine the closest divider:

Filter_Clock_Frequency = 7.16 MHz / (256-Filter_Divider_Register)

DMA Transfer Count Reload Register **A4h**

7	6	5	4	3	2	1	0
DMA transfer counter reload – low byte							

On reset, this register assumes the value of 00h.

DMA Transfer Count Reload Register **A5h**

7	6	5	4	3	2	1	0
DMA transfer counter reload – high byte							

On reset, this register assumes the value of F8h.

The FIFO control logic of the ES1878 has a 16-bit counter for controlling transfers to and from the FIFO. These registers are the reload value for that counter which is the value that gets copied into the counter after each overflow (plus at the beginning of the initial DMA transfer). The counter will be incremented after each successful byte is transferred via DMA. Since the counter counts up towards FFFFh and then overflows, the reload value is in 2's complement form.

For Auto-Initialize DMA, the counter is used to generate interrupt requests to the system processor: in this mode DMA continues indefinitely as far as the ES1878 is

concerned. In a typical application the counter is programmed to be one-half of the DMA buffer maintained by the system processor. In this case an interrupt is generated whenever DMA switches from one half of the circular buffer to the other.

For Normal Mode DMA, DMA requests will be halted at the time that the counter overflows, until a new DMA transfer is commanded by the system processor. Again, an interrupt request will be generated to the system processor if bit 6 of register B1 is set high.

Mic Preamp **A9h**

7	6	5	4	3	2	1	0
Reserved					Enable mic preamp	Reserved	

Bit 2, if set high, enables +26 dB gain in the microphone preamp. If set low, the microphone preamp has no gain (0 dB). This bit is set high by hardware reset.

Note: All other bits must first be read and preserved when writing to this register.

Legacy Audio Interrupt Control **B1h**

7	6	5	4	3 2 1 0
Game Compatible IRQ	Enable IRQ overflow ext mode DMA counter	Enable IRQ for FIFO1 half-empty status edge	x	See text

On hardware reset, the IRQs are disabled, so the ES1878 must program bits 3 and 2 of Extended Mode Indirect register B1h, which control assignment of IRQs. On reset, bits 7:5 are cleared and all IRQs are disabled.

Bit 7 Processor IRQ. Reserved for Compatibility Mode. Should be left zero for Extended Mode.

Bit 6 Should be set high to receive interrupts for each overflow of the ES1878 DMA counter in Extended Mode.

Bit 5 Should be set high to receive interrupts for FIFO half-empty transitions when doing block I/O to or from the FIFO in Extended Mode.

Bit 4 Bit 4 of Extended register B1h is don't-care, not the interrupt enable control. The audio device activate bit serves the purpose of enabling the interrupt pin.

Bits 3:0 Read-only. They decode the selected interrupt number for the first audio interrupt as follows:

Bits				Audio 1 Interrupt
3	2	1	0	
0	0	0	0	2, 9, all others
0	1	0	1	5
1	0	1	0	7
1	1	1	1	10

Legacy Audio DRQ Control **B2h**

7	6	5	4	3 2 1 0
Game compatible DRQ	Enable DRQ for extended DMA	Enable DRQ for game compatible DMA	x	See text

On hardware reset, the DRQs are all disabled, and the ES1878 processor needs to program the select registers bits 3 and 2. The selected DRQ output is either active or inactive. It is inactive if bits D6 and D5 are both low.

The inactive DRQ can either be high-impedance or have an active pull-down device, determined by the value of bit 4 of register B2h.

The active pull-down device is required for Compatibility Mode to prevent false DRQ requests during the interval between the time that the application programs the system DMA controller and the time the application informs the ES1878 to proceed with DMA. The active pull-down device should be able to sink enough current to override any pull-up device on the system motherboard. It is designed to simulate a resistor < 500 ohms.

On any reset, all DRQ sources are disabled by clearing bits 7:5. Bits 1 and 0 are don't-care.

Bit 7 Reserved for game compatible DRQ; should be left zero for Extended Mode.

Bit 6 1 = Enable DRQ outputs and DACKB inputs for DMA transfers in Extended Mode.
0 = Enable block I/O to/from the FIFO in Extended Mode.

Bit 5 Enable DRQ game compatibility DMA; should be left zero for Extended Mode.

Bit 4 Bit 4 of Extended register B2h is don't-care. The DRQ lines always drive, (there is no enable). If neither bit 6 nor 5 is set high, the first audio DRQ is always low. The pull-down feature of DRQ pins in previous parts is discontinued in the ES1878.

Bits 3:0 Read-only. The selected DMA channel numbers for the first audio DMA channel are decoded to set register B2h, bits 3 and 2, and 1 and 0 as follows:

Bits				Audio 1 DMA
3	2	1	0	
0	1	0	1	0
1	0	1	0	1
1	1	1	1	3
0	0	0	0	all others

Input Volume Control Register **B4h**

7	6	5	4	3	2	1	0
Input volume right				Input volume left			

On reset, this register assumes the value of FFh.

Digital ADC Offset Adjust

Extended registers BAh and BBh can be used to digitally adjust the ADC results in order to cancel a D.C. offset.

Left Channel ADC Offset Adjust **BAh**

7	6	5	4	3	2	1	0
0	Enable SR ADJ	1: Disable time delay on analog wake-up	Sign	Adjust magnitude			

Bit 6 1 = Enable sample rate adjustment for more accuracy. The ES1878 includes this feature of the ES1888 that adjusts the sample rates. A 1 causes the 14.31818 MHz clock to be adjusted slightly to make the 44.1 kHz sample rate (and integral divisions of that rate) nearly exact. This bit is cleared by hardware reset.

Bit 5 Normally, the AOUT_L and AOUT_R pins are muted for 100 milliseconds ± 20 milliseconds after hardware reset or after the analog subsystems wake from power-down. This delay can be disabled by setting bit 5. This bit is cleared by hardware reset.

Bit 4 The offset that is added to the left channel ADC result is calculated as follows:

If sign = 0: Offset = Magnitude * 64
If sign = 1: Offset = -Magnitude * 64

These registers are reset to zero by hardware reset and are unaffected by software reset.

Right Channel ADC Offset Adjust **BBh**

7	6	5	4	3	2	1	0
0			Sign	Adjust magnitude			

Bit 4 The offset that is added to the right channel ADC result is calculated as follows:

If sign = 0: Offset = Magnitude * 64
If sign = 1: Offset = -Magnitude * 64

These registers are reset to zero by hardware reset and are unaffected by software reset.

FM DAC Volume Register **BCh**

7	6	5	4	3	2	1	0
FM DAC volume 1	FM DAC volume 0	1	1	0	1	1	0

Bits 7:6 See table below:

Bit 7	Bit 6	FM DAC Level
VOL 1	VOL 0	
0	0	-1.5 dB: recommended level if ES689/ES690 shares FM DAC
0	1	+1.5 dB
1	0	0 dB (hardware reset default)
1	1	-3 dB

Bits 5:0 Must be set to the value of 110110, as shown, for proper operation.

AUDIO PROCESSOR COMMAND SUMMARY

Table 19 Command Summary

Command	Data Byte(s) Write/Read	Function
10h	1 write	Direct write 8-bit DAC. Data is 8-bit unsigned format.
11h	2 writes	Direct write 16-bit DAC. Data is 16-bit unsigned format, first low byte then high byte.
14h	2 writes	Start normal mode DMA for 8-bit DAC transfer. Data is transfer count-1, least byte first. Stereo DAC transfer if Stereo flag is set in Mixer register 0Eh. Maximum sample rate is 44 kHz mono, 22 kHz stereo.
15h	2 writes	Start normal mode DMA for 16-bit DAC transfer. Data is transfer count-1, least byte first. Stereo DAC transfer if stereo flag is set in Mixer register 0Eh. Maximum sample rate is 22 kHz mono, 11 kHz stereo.
1Ch		Start auto-initialize mode DMA for 8-bit DAC transfer. Block size must be previously set by command 48h. Stereo DAC transfer if stereo flag is set in Mixer register 0Eh. Maximum sample rate is 44 kHz mono, 22 kHz stereo.
1Dh		Start auto-initialize mode DMA for 16-bit DAC transfer. Block size must be previously set by command 48h. Stereo DAC transfer if stereo flag is set in Mixer register 0Eh. Maximum sample rate is 22 kHz mono, 11 kHz stereo.
20h	1 read	Direct mode 8-bit ADC. Data is 8-bit unsigned. Firmware-controlled input volume for AGC.
21h	2 read	Direct mode 16-bit ADC, returns least byte first. Data is 16-bit unsigned format. Input volume controlled via command DDh.
24h	2 writes	Start normal mode DMA for 8-bit ADC transfer. Data is transfer count-1, least byte first. Firmware-controlled input volume for AGC. Maximum sample rate is 22 kHz: use command 99h for higher rates up to 44 kHz.
25h	2 writes	Start normal mode DMA for 16-bit ADC transfer. Data is transfer count-1, least byte first. Input volume controlled via command DDh. Maximum sample rate is 22 kHz.
2Ch		Start auto-initialize mode DMA for 8-bit ADC transfer. Block size must be previously set by command 48h. Firmware-controlled input volume for AGC. Maximum sample rate is 22 kHz: use command 98h for higher rates up to 44 kHz.
2Dh		Start auto-initialize mode DMA for 16-bit ADC transfer. Block size must be previously set by command 48h. Input volume is controlled via command DDh. Maximum sample rate is 22 kHz.
30h/31h		MIDI input mode. Detects MIDI serial input data and transfers to Data register, setting Data-Available flag in register Audio_Base+Eh. Command 31h will also generate an interrupt request for each byte received. Exit MIDI input mode by executing a write to port Audio_Base+Ch. The data written is ignored. A software reset will also exit this mode.
34h/35h		MIDI UART mode. Acts like commands 30h/31h, except that any data written to Audio_Base+Ch will be transmitted as MIDI serial output data. The only way to exit this mode is a software reset.
38h	1 write	MIDI output. Transmit one byte.
40h	1 write	Set time constant, X, for timer used for DMA mode DAC/ADC transfers: rate = 1 MHz / (256-X). X must be less than or equal to 233. For stereo DAC, program sample rate for twice the per-channel rate.
41h	1 write	Alternate set time constant, X: rate = 1.5 MHz / (256-X). This command provides more accurate timing for certain rates such as 22,050. X must be less than or equal to 222. For stereo DAC, program sample rate for twice the per-channel rate.
42h	1 write	Set filter clock independently of timer rate. (note that the filter clock is automatically set by commands 40h/41h). Filter clock rate = 7.16E6 / (256-X). The relationship between the low-pass filter -3 dB point and the filter clock rate is approximately 1:82.
48h	2 writes	Set block size to -1 for high-speed mode and auto-init mode transfer, least byte first.

Table 19 Command Summary (Continued)

Command	Data Byte(s) Write/Read	Function
64h	2 writes	Start **ESPCM**® 4.3-bit (low compression) format DMA transfer to DAC. Data is transfer count-1, least byte first.
65h	2 writes	Same as command 64h, except with reference byte flag.
66h	2 writes	Start **ESPCM**® 3.4-bit (medium compression) format DMA transfer to DAC. Data is transfer count-1, least byte first.
67h	2 writes	Same as command 66h, except with reference byte flag.
6Ah	2 writes	Start **ESPCM**® 2.5-bit (high compression) format DMA transfer to DAC. Data is transfer count-1, least byte first.
6Bh	2 writes	Same as command 6Ah, except with reference byte flag.
6Eh	2 writes	Start **ESPCM**® 4.3-bit (low compression) format ADC, compression, and DMA transfer. Data is transfer count-1, least byte first.
6Fh	2 writes	Same as command 6Eh, except with reference byte flag.
74h	2 writes	Start ADPCM 4-bit format DMA transfer to DAC. Data is transfer count-1, least byte first.
75h	2 writes	Same as command 74h, except with reference byte flag.
76h	2 writes	Start ADPCM 2.6-bit format DMA transfer to DAC. Data is transfer count-1, least byte first.
77h	2 writes	Same as command 76h, except with reference byte flag.
7Ah	2 writes	Start ADPCM 2-bit format DMA transfer to DAC. Data is transfer count-1, least byte first.
7Bh	2 writes	Same as command 7Ah, except with reference byte flag.
80h	2 writes	Generate silence period. Data is number of samples-1.
90h		Start auto-initialize, DMA 8-bit transfer to DAC. Transfer count must be previously set by command 48h.
91h		Start DMA 8-bit transfer to DAC. Transfer count must be previously set by command 48h.
98h		Start high-speed mode, auto-initialize, DMA 8-bit transfer from ADC. Transfer count must be previously set by command 48h. There is no AGC. Input volume is controlled with command DDh. Maximum sample rate is 44 kHz.
99h		Start high-speed mode, DMA 8-bit transfer from ADC. Transfer count must be previously set by command 48h. There is no AGC. Input volume is controlled with command DDh. Maximum sample rate is 44 kHz.
Axh, Bxh, Cxh		(where x = 00h to 0Fh) ES1878 Extension commands. Many of these commands are used to access the ES1878's indirect registers. For information on these registers, see the register descriptions.
C0h		Enable reads of ES1878 registers used for Extended Mode: Axh, Bxh.
C1h		Resume after suspend.
C6h		Enable ES1878 Extension commands Ax, Bx. Must be issued after every reset.
C7h		Disable ES1878 Extension commands Ax, Bx.
D0h		Pause DMA. Internal FIFO operations will continue until the FIFO is empty (DAC transfer) or full (ADC transfer). It is not necessary to use this command to stop DMA if the transfer is completed normally and the end-of-DMA interrupt is generated.
D1h		Enable voice DAC input to mixer.
D3h		Disable voice DAC input to mixer.
D4h		Continue DMA after command D0h.
D8h	1 read	Return voice DAC enable status: 0=disabled FFh=enabled
E1h	2 read	Return version number high (3), followed by version number low (1). This indicates Sound Blaster Pro compatibility.

POWER MANAGEMENT

Power management in the ES1878 is controlled by PnP Configuration register 2Dh. In previous ESS *Audio***Drive**® chips, power management was controlled via I/O port Audio_Base+7h. Only bit 5 (FM reset) and bit 7 (suspend request) of I/O port Audio_Base+7h are supported in the ES1878.

GPO, XSD, and XSC are not affected during power-down. XA[3:0] are high-impedance during power-down Modes 0 and 1.

Power Management Characteristics

The ES1878 has four power modes. The mode is determined by bits 1 and 0 of Vendor-Defined Card-Level register 2Dh.

In any mode, the configuration device can always be read and written.

Mode Transitions

The mode can be changed at any time with only one restriction: if a crystal is connected to XI/XO, and the chip is in mode 0, then the chip must be placed in mode 1 for a period of 25 milliseconds or more to allow the oscillator to settle, before changing to mode 2 or mode 3.

Table 20 Power Mode Description

Mode	Description	Notes
0	Full power-down. Crystal oscillator disabled. AOUT_L/R held at approximately CMR by high value resistors.	All inputs static at VDDD or GND.
1	Crystal oscillator enabled. Analog powered down.	All inputs other than XI are static.
2	Analog powered up. ES978 interface up. Joystick, MPU-401 up. I^2S up. Audio, FM, ES689/ES690 interface, and DSP serial interface down.	Digital standby.
3	Full power on. This is the state after hardware reset.	Normal operating conditions.

BIOS Power Management

There are at least three types of BIOS power management:

1. Suspend-to-Disk. Here the context of the ES1878 is uploaded to disk, then power is removed from the ES1878. Later, power is applied to the ES1878 along with a hardware reset, and the context of the chip is downloaded.
2. Idle power-down. The processor is stopped or slowed. Application software is not running. The ES1878 is put in full power-down mode by BIOS. The power supply is still connected to the chip. Later, the BIOS returns the ES1878 to full power up.
3. Power reduction. If the system can generate an System Management Interrupt (SMI) upon I/O access to the audio and FM addresses, then the BIOS can implement a power reduction technique: the BIOS periodically polls the activity flags of the ES1878 in order to determine if the chip is in use. If not in use for some period, it can power down the chip and enable the SMI. The first application to access the audio or FM address space will trigger the SMI, which causes the BIOS to power up the ES1878, and de-activate the SMI.

BIOS power management is well suited to a DOS environment. It must also work with the ES1878 Windows driver which implements power management as well.

Suspend-to-Disk / Resume-from-Disk

Suspend-to-Disk is the name given to the procedure where the entire context of the ES1878 is uploaded to be saved on disk. After saving the context, power can be removed entirely from the ES1878. When power is re-applied (and a hardware reset is given), the ES1878 state must be restored from the saved context.

The suspend procedure consists of the following tasks:

1 Upload PnP configuration information.
2 Upload FM registers.
3 Upload mixer registers.
4 Upload MPU-401 state.
5 Upload audio state (using suspend request: bit 7 of port "Audio_Base+7h").

An example DOS assembly language program is available demonstrating suspend-to-disk, followed by resume-from-disk, from a TSR that hooks the system timer interrupt. This program demonstrates how a DOS application such as a game can be suspended in the middle of audio playback.

DMA and Interrupts During Suspend-to-Disk

The ES1878 cannot properly suspend and resume during audio playback unless the DMA and interrupt controller are also properly suspended. Alternatively, the DMA and interrupt controllers should not have power removed. In the latter case, it is important that all DRQ and IRQ lines are held low. All bus lines should be low if power is removed from a device connected to the bus that has its power supply removed. Also, during and after reset, the ES1878 DRQ and IRQ lines are high-impedance. If a DRQ or IRQ line floats high, it can produce a false DMA cycle or interrupt.

TIMING DIAGRAMS

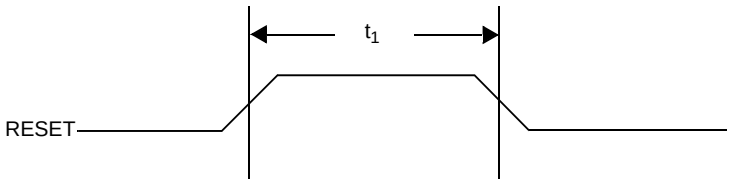


Figure 15 Reset Timing

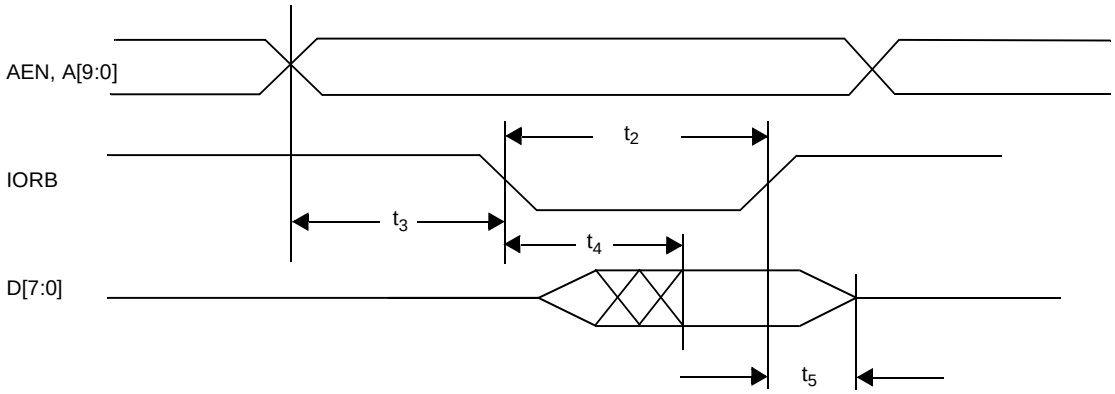


Figure 16 I/O Read Cycle

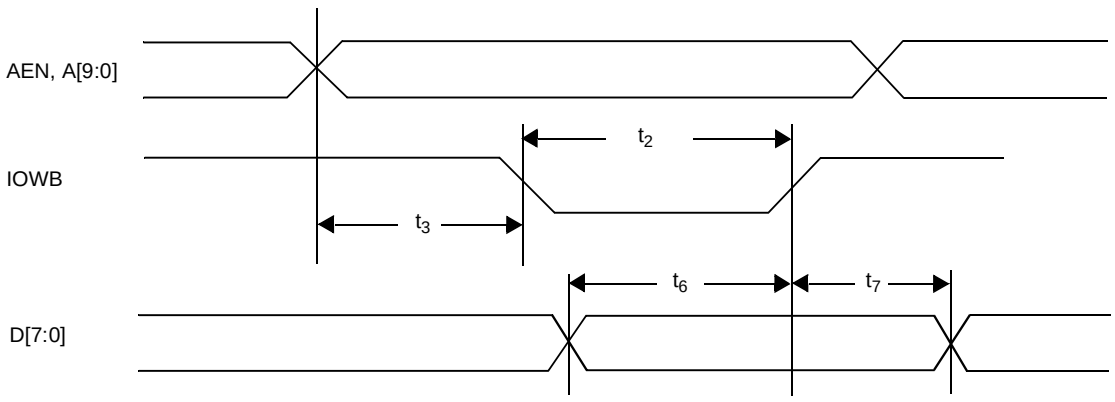


Figure 17 I/O Write Cycle

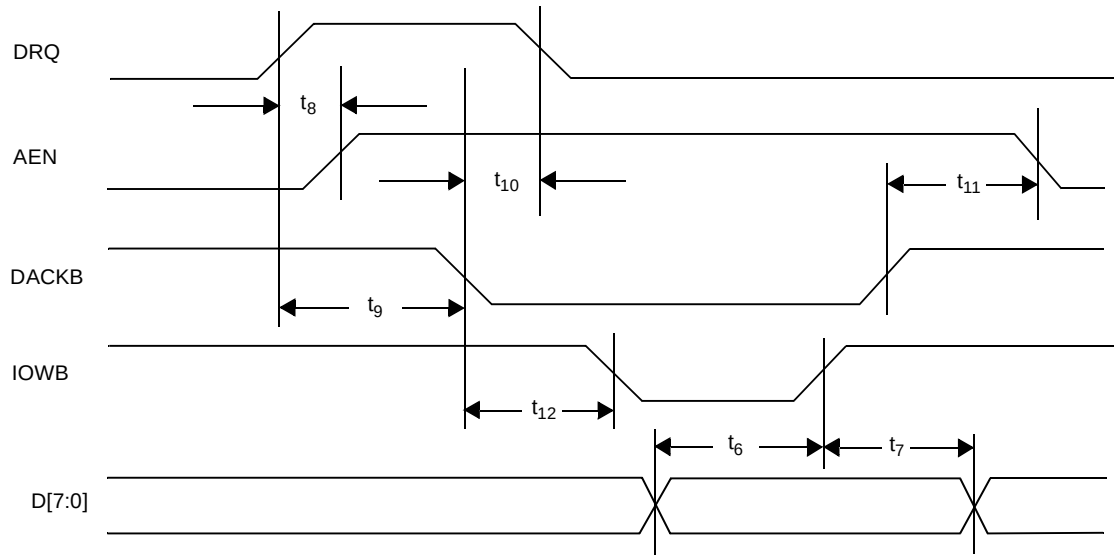


Figure 18 Compatibility Mode DMA Write Cycle

Note: In Compatibility Mode DMA, the DMA request is reset by the acknowledge signal going low. In Extended Mode DMA, the DMA request is reset when the acknowledge signal is low AND the correct command signal is low – either IORB (for DMA read from I/O device) or IOWB (for DMA write to I/O device). For Extended Mode DMA, the time t_{10} is relative to the later of the falling edge of the acknowledge signal or the command signal.

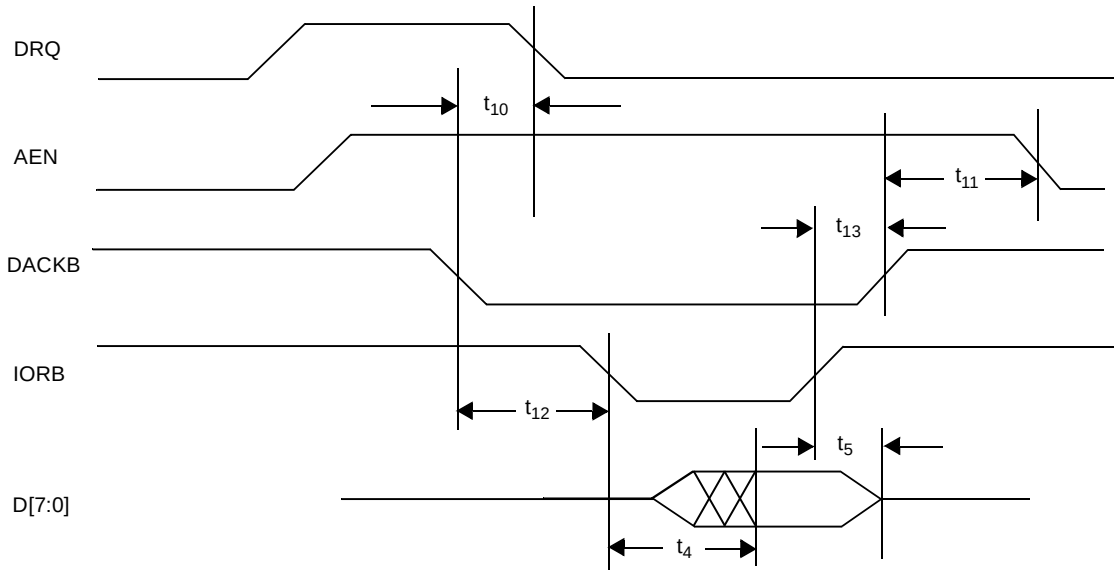


Figure 19 Compatibility Mode DMA Read Cycle

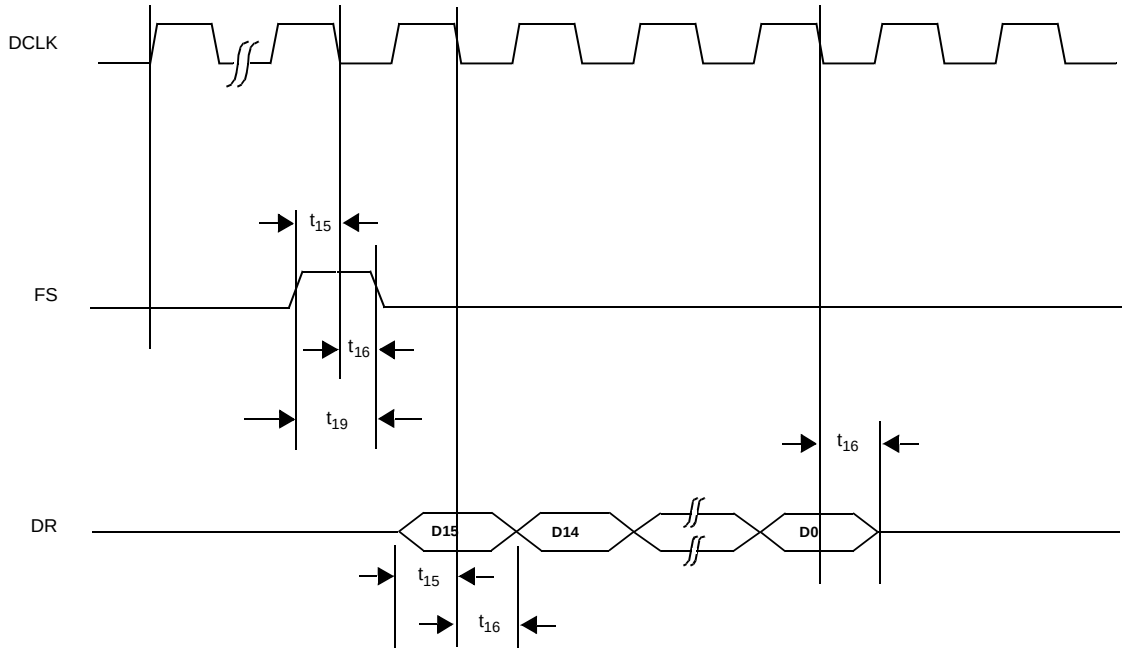


Figure 20 Serial Mode Receive Operation

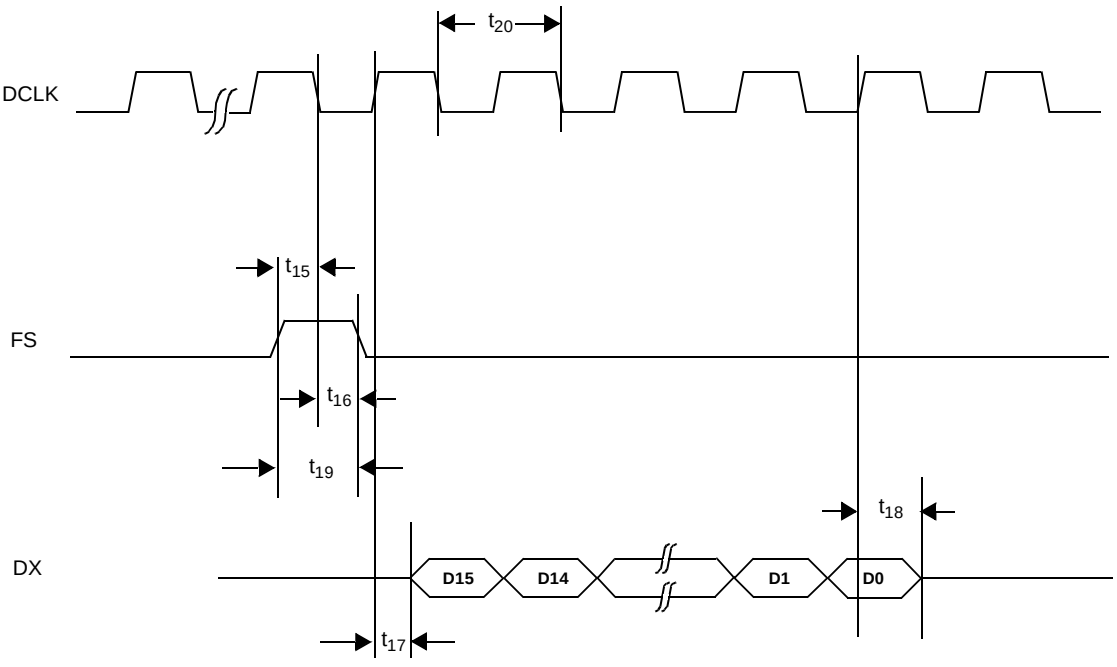


Figure 21 Serial Mode Transmit Operation

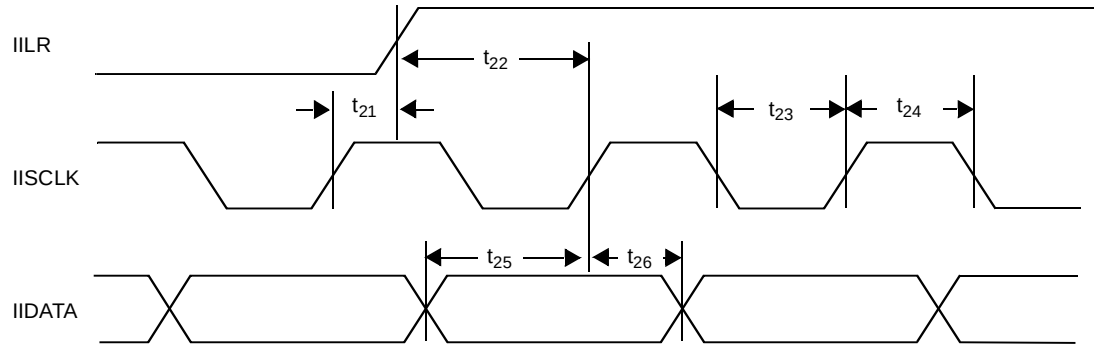
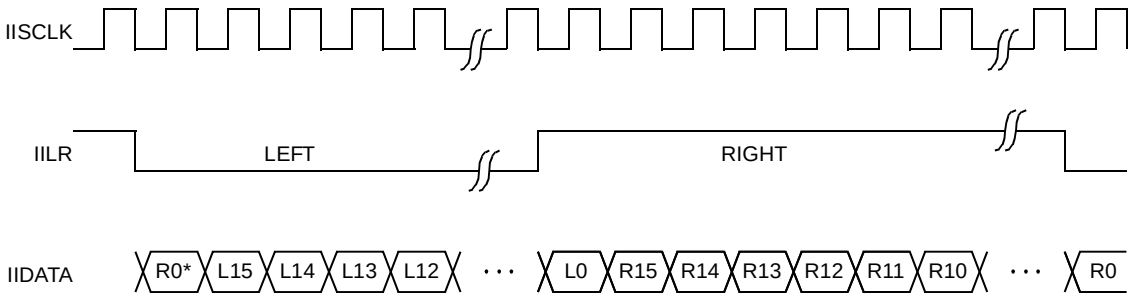


Figure 22 Serial Input Timing for I²S Interface



* Note: LSB of right channel, previous sample.

Figure 23 I²S Digital Input Format with 16 SCLK Periods

Table 21 ES1878 Timing Characteristics

Symbol	Parameter	Min	Typ	Max	Unit
t_1	Reset pulse width	300			ns
t_2	IORB, IOWB pulse width	100			ns
t_3	IORB, IOWB address setup time	10			ns
t_4	Read data access time			70	ns
t_5	Read data hold time			10	ns
t_6	Write data setup time	5			ns
t_7	Write data hold time	10			ns
t_8	DMA request to AEN high	0			ns
t_9	DMA request to DMA ACK low	10			ns
t_{10}	DMA ACK to request release[a]			30	ns
t_{11}	DMA ACK high to AEN low	0			ns
t_{12}	DMA ACK to IOWB, IORB low	0			ns
t_{13}	IOWB, IORB to DMA ACK release	20			ns
t_{14}	Crystal frequency, XI/XO		14.318		MHz
t_{15}	FS, DS setup time to DCLK falling edge	15			ns
t_{16}	FS, DR hold time from DCLK falling edge	10			ns
t_{17}	DX delay time from DCLK rising edge			20	ns
t_{18}	DX hold time from DCLK rising edge	10			ns
t_{19}	FS pulse width	60	500		ns
t_{20}	DCLK clock frequency		2.048		MHz
t_{21}	IISCLK delay	2			ns
t_{22}	IISCLK setup	32			ns
t_{23}	Bit clock low	22			ns
t_{24}	Bit clock high	22			ns
t_{25}	Data setup time	32			ns
t_{26}	Data hold time	2			ns

a. Note: In Compatibility Mode DMA, the DMA request is reset by the acknowledge signal going low. In Extended Mode DMA, the DMA request is reset when the acknowledge signal is low AND the correct command signal is low – either IORB (for DMA read from I/O device) or IOWB (for DMA write to I/O device). For Extended Mode DMA, the time t_{10} is relative to the later of the falling edge of the acknowledge signal or the command signal.

ELECTRICAL CHARACTERISTICS

Absolute Maximum Ratings

Rating	Symbol	Value
Analog supply voltage range	VDDA	-0.3 to 7.0 V
Digital supply voltage range	VDDD	-0.3 to 7.0 V
Input voltage	VIN	-0.3 to 7.0 V
Operating temperature range	TA	0 to 70 °C
Storage temperature range	TSTG	-50 to 125 °C

Thermal Characteristics

The ES1878 is designed to operate at case temperatures less than 78 °C.

WARNING: Stressing the device beyond the Absolute Maximum Ratings may cause permanent damage. These are stress ratings only. Operation beyond the Operating Conditions is not recommended, and extended exposure beyond the Operating Conditions may affect device reliability.

Operating Condition

Digital supply voltage . . . . . 3.0 V to +5.50 V

Analog supply voltage. . . . . 4.75 V to +5.25 V

DC Electrical Characteristics

Table 22 Digital Characteristics

(VDDD = 5.0 V ± 10%; TA = 25 °C)

Symbol	Parameter	Min	Typ	Max	Unit (conditions)
VIH1	Input high voltage: all inputs except XI.	2.0 V			VDDD = min
VIH2	Input high voltage: XI.	3.0 V			VDDD = min
VIL	Input low voltage		0.8 V		VDDD = max
VOL1	Output low voltage: all outputs except D[7:0], IRQ(A-E), DRQ(A-C), XSC, XSD		0.4 V		IOL = 4 mA, VDDD = min
VOH1	Output high voltage: all outputs except D[7:0], IRQ(A-E), DRQ(A-C), XSC, XSD	2.4 V			IOH = -3 mA, VDDD = max
VOL2	Output low voltage: D[7:0], IRQ(A-E), DRQ(A-C), XSC, XSD		0.4 V		IOL = 16 mA, VDDD = min
VOH2	Output high voltage: D[7:0], IRQ(A-E), DRQ(A-C), XSC, XSD	2.4 V			IOH = -12 mA, VDDD = max
VOL3	Output low voltage		0.4 V		IOL = 0.8 mA

Table 23 Analog Characteristics

(VDDA = 5.0 V ± 5%; TA = 25 °C)

Pins	Parameter	Min	Typ	Max	Units
CMR, VREF	Reference voltage		2.25		volts
LINE_L, LINE_R, AUXA_L, AUXA_R, AUXB_L, AUXB_R, MIC	Input impedance	30k	70k	100k	ohms
CIN_L, CIN_R		35k	50k	65k	ohms
FDXI		30k	70k	100k	ohms
FOUT_L, FOUT_R	Output impedance	3.5k	5k	6.5k	ohms
AOUT_L, AOUT_R max load for full-scale output range		10k	10k		ohms
FDXO			5k	6.5k	ohms
MIC	Input voltage range	10		125	mVp-p
LINE_L, LINE_R, AUXA_L, AUXA_R, AUXB_L, AUXB_R		0.5		VDDA - 1.0	volts
FDXI				3.5	Vp-p
AOUT_L, AOUT_R full-scale output range	Output voltage range	0.5		VDDA - 1.0	volts
FDXO			2.0		Vp-p
MIC	Mic preamp gain		26		decibels

Power Management Characteristics

Table 24 Current Consumption for Power Modes

Mode	Typical IDDD at 3.3V	Typical IDDD at 5.0V	Typical IDDA at 5.0V
0	10 µA	10 µA	20 µA
1	1 mA	3 mA	20 µA
2	3 mA	5 mA	30 mA
3	16 mA	30 mA	30 mA

MECHANICAL DIMENSIONS

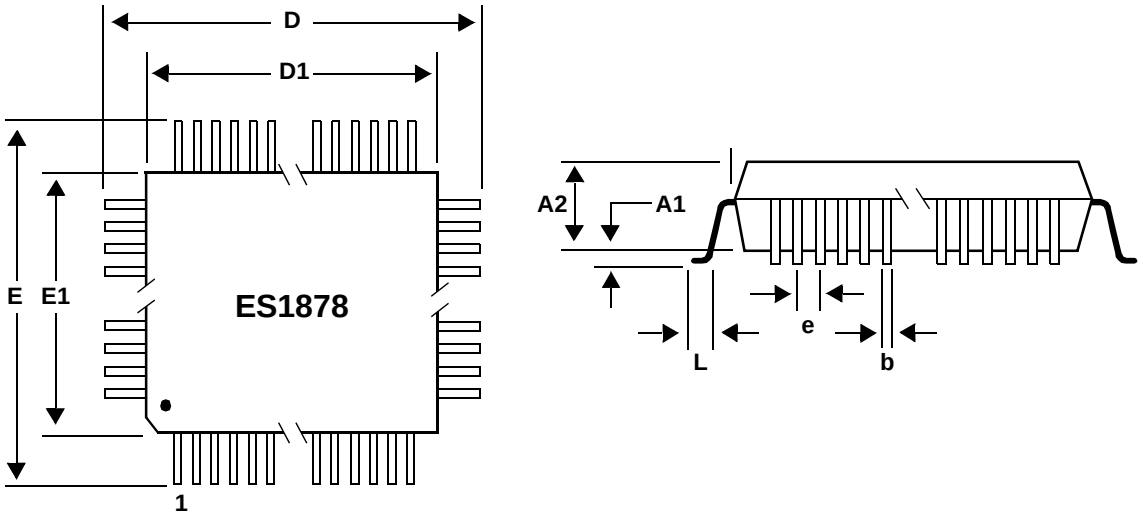


Figure 24 ES1878 Physical Dimensions

Symbol	Description	Millimeters		
		Min	Nom	Max
D	Lead-to-lead, X-axis	15.75	16.00	16.25
D1	Package's outside, X-axis	13.90	14.00	14.10
E	Lead-to-lead, Y-axis	15.75	16.00	16.25
E1	Package's outside, Y-axis	13.90	14.00	14.10
A1	Board standoff	0.05	0.10	0.15
A2	Package thickness	1.35	1.40	1.45
b	Lead width	0.17	0.22	0.27
e	Lead pitch		0.50	
L	Lead length	0.45	0.60	0.75
	Coplanarity			0.102
	Number of leads in X-axis		25	
	Number of leads in Y-axis		25	
	Number of leads, total		100	
	Package type		SQFP	

APPENDIX A: ES1878 INTERNAL PNP RESOURCE ROM

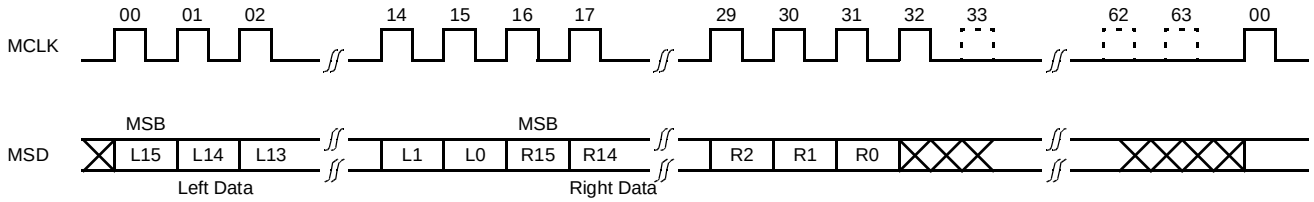
```
; PnP Resource ROM for ES1878 v001
;
; *********************************************************
; Start of ESS Proprietary Header (8 bytes)
;
; *********************************************************

0A5H                                ; PnP OK byte
059H                                ; IRQA = 9 IRQB = 5   (automatically loaded into vendor register 20H)
0A7H                                ; IRQC = 7 IRQD = 10 (automatically loaded into vendor register 21H)
00BH                                ; IRQE = 11 (automatically loaded into vendor register 22H)
010H                                ; DRQA = 0 DRQB = 1(automatically loaded into vendor register 23H)
003H                                ; DRQC = 3 (automatically loaded into vendor register 24H)
000H                                ; MPU-401 part of audio, enable all interrupt requests and
                                    ; volume controls (automatically loaded into vendor register 25H)
000H                                ; reserved

; *********************************************************
; Start of PnP Resource Header
;
; *********************************************************

016H, 073H, 018H, 078H              ; "ESS1878" product id for ES1878
0FFH, 0FFH, 0FFH, 0FFH              ; serial number FFFFFFFF (not supported)
000H                                ; header checksum
00AH, 010H, 010H                    ; PnP 1.0,   ESS version 1.0
082H, 023H, 000H, "ESS ES1878 Plug and Play AudioDrive"

; *********************************************************
; LOGICAL DEVICE 0 -- Configuration Ports
; 8 bytes at any I/O address that is a multiple of 8
;
; *********************************************************

015H, 016H, 073H, 000H, 004H, 000H          ; ESS0004
047H, 001H, 000H, 008H, 0F8H, 00FH, 008H, 008H; 800H-FF8H 8 bytes

; *********************************************************
; LOGICAL DEVICE 1 -- Audio Controller w/FM and MPU-401
;
; *********************************************************

015H, 016H, 073H, 018H, 078H, 000H  ; ESS1878

                                    ; Basic configuration 0000
031H, 000H
02AH, 002H, 008H                    ; DMA 0:   DRQ 1
02AH, 009H, 008H                    ; DMA 1:   DRQ 0 3
022H, 020H, 000H                    ; INT 0:   IRQ 5
04BH, 020H, 002H, 010H              ; audio 220-22F 16 bytes
04BH, 088H, 003H, 004H              ; FM      388-38B
04BH, 030H, 003H, 002H              ; MIDI    330-331
```

```
                                        ; Basic configuration 0001
031H, 001H
02AH, 002H, 008H                        ; DMA 0:   DRQ 1
02AH, 009H, 008H                        ; DMA 1:   DRQ 0 3
022H, 0A0H, 006H                        ; INT 0:   IRQ 5 7 9 10
047H, 001H, 020H, 002H, 040H, 002H, 020H, 010H; 220 240 16 bytes
04BH, 088H, 003H, 004H                  ; 388-38B
047H, 001H, 000H, 003H, 030H, 003H, 030H, 002H; 300 or 330 2 bytes

                                        ; Basic configuration 0002
031H, 001H
02AH, 00BH, 008H                        ; DMA 0:   DRQ 0 1 3
02AH, 00BH, 008H                        ; DMA 1:   DRQ 0 1 3
022H, 0A0H, 00EH                        ; INT 0:   IRQ 5 7 9 10 11
047H, 001H, 020H, 002H, 080H, 002H, 020H, 010H; 220 240 260 280 16 bytes
04BH, 088H, 003H, 004H                  ; 388-38B
047H, 001H, 000H, 003H, 030H, 003H, 030H, 002H; 300 or 330 2 bytes

                                        ; Basic configuration 0003
031H, 001H
02AH, 00BH, 008H                        ; DMA 0:   DRQ 0 1 3
02AH, 00BH, 008H                        ; DMA 1:   DRQ 0 1 3
022H, 0A0H, 00EH                        ; INT 0:   IRQ 5 7 9 10 11
047H, 001H, 020H, 002H, 080H, 002H, 020H, 010H; 220 240 260 280 16 bytes
04BH, 088H, 003H, 004H                  ; 388-38B
047H, 001H, 000H, 008H, 0FEH, 00FH, 002H, 002H; 800/801-FFE/FFF 2 bytes

                                        ; Basic configuration 0004
031H, 002H
02AH, 00BH, 008H                        ; DMA 0:   DRQ 0 1 3
02AH, 00BH, 008H                        ; DMA 1:   DRQ 0 1 3
022H, 0A0H, 00EH                        ; INT 0:   IRQ 5 7 9 10 11
047H, 001H, 020H, 002H, 080H, 002H, 020H, 010H; 220 240 260 280 16 bytes
047H, 001H, 000H, 008H, 0FCH, 00FH, 004H, 004H; 800/804-FFC/FFF 4 bytes
047H, 001H, 000H, 008H, 0FEH, 00FH, 002H, 002H; 800/801-FFE/FFF 2 bytes

038H                                    ; end configurations

; ******************************************************
; LOGICAL DEVICE 2 -- Joystick
;
; ******************************************************

015H, 016H, 073H, 000H, 005H, 000H      ; ESS0005

031H, 000H                              ; Basic configuration 0000
04BH, 001H, 002H, 001H                  ; 201
031H, 001H                              ; Basic configuration 0001
047H, 001H, 000H, 002H, 00FH, 002H, 001H, 001H; 200/200-20F/20F 1 byte

038H                                    ; end dependent functions
01CH, 041H, 0D0H, 0B0H, 02FH            ; Compatible ID: PNPB02F
079H, 000H                              ; end tag + checksum
```

APPENDIX B: ES689/ES690 DIGITAL SERIAL INTERFACE

In order for the ES689/ES690 to acquire the FM DAC, bit 4 of Mixer Extension register 48h inside the ES1878 must be set high. When bit 4 is set high, activity on the MCLK signal will cause the FM DAC to be connected to the ES689/ ES690. If MCLK stays low for more than a few sample periods the ES1878 will reconnect the FM DAC to the FM synthesizer.

After reset, the ES689/ES690 will transmit samples continuously. In this mode bit 4 of Mixer Extension register 48h must be set/cleared to assign the current owner of the FM DAC. The ES689/ES690 can be programmed to enter Activity-Detect Mode using system exclusive command 4. In this mode, the ES689/ES690 will block the serial port output (i.e., set MSD and MCLK low) if no MIDI input is detected on the MSI pin for a period of 5 seconds. It will resume output of data on the serial port as soon as a MIDI input is detected on the MSI pin. This is the recommended mode of operation.



Bit Clock Rate (MCLK):	2.75 MHz
Sample Rate:	42,968.75 Hz
MCLK Clocks per Sample:	33 clocks (+ 31 missing clocks)
MSD Format:	16 bits, unsigned (offset 8000h), MSB first

MSD changes after rising edge of MCLK. Hold time relative to MCLK rising edge is 0-25 nanoseconds.

APPENDIX C: I²S ZV INTERFACE REFERENCE

(Excerpted from “PCMCIA Document Number 0135 – Release 010 1/15/96”)

Overview

The following diagram shows the system level concept of the ZV Port. The diagram demonstrates how TV in a window could be achieved in a portable computer with a low cost PC Card. An MPEG or teleconferencing card could also be plugged into the PC Card slot.

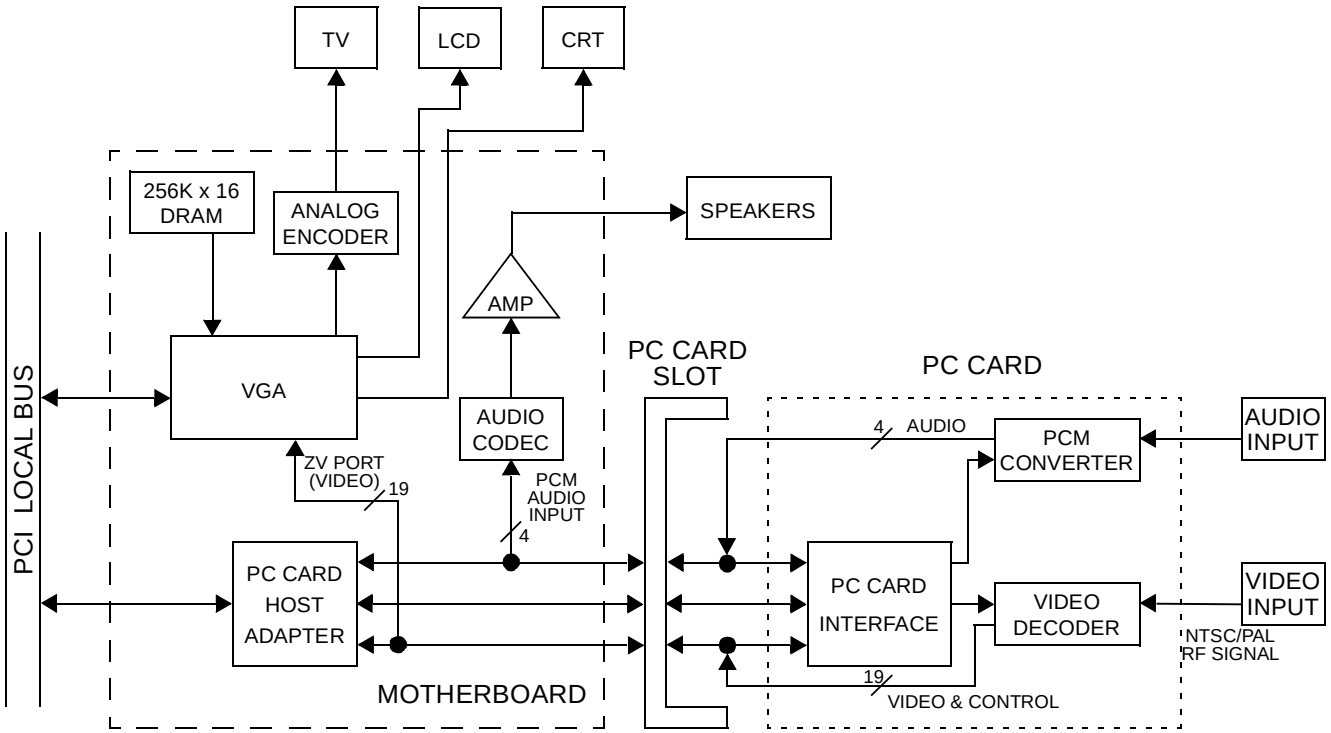


Figure 25 Example ZV Port Implementation

The Audio Interface

The ZV Port compliant PC Card sends audio data to the host computer using Pulse Code Modulation (PCM). The audio data is transferred using the serial I²S format. The audio circuitry in the host system is primarily a PCM DAC.

The PCM audio DAC is a complete stereo digital-to-analog system including digital-interpolation, delta-sigma digital-to-analog conversion, digital de-emphasis and analog filtering. Only the normal power supply decoupling components and one resistor and capacitor per channel for analog signal reconstruction are required.

The DAC accepts data at standard audio frequencies including 48 kHz, 44.1 kHz, 32 kHz, and 22 kHz. Audio data is input via the serial data input pin, SDATA. The Left/Right Clock (LRCLK) defines the channel and delineation of data. There Serial Clock (SCLK) clocks the audio data into the input data buffer. The Master Clock (MCLK) is used to operate the digital interpolation filter and the delta-sigma modulator.

Table 25 Common Clock Frequencies

	MCLK(MHz)	
LRCLK (KHz)	256x	384x
22	5.632	8.448
32	8.192	12.2880
44.1	11.2896	16.9344
48	12.2880	18.4320

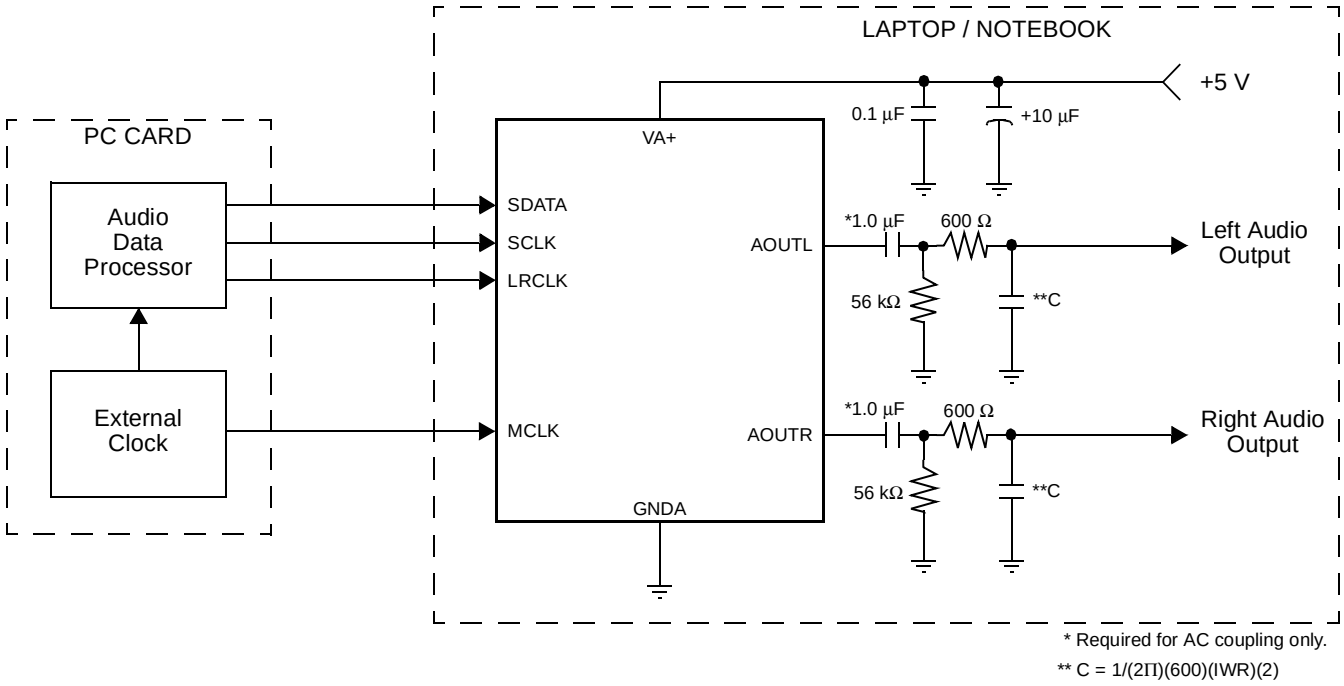


Figure 26 Typical ZV Port Audio Implementation

Audio Interface Timing

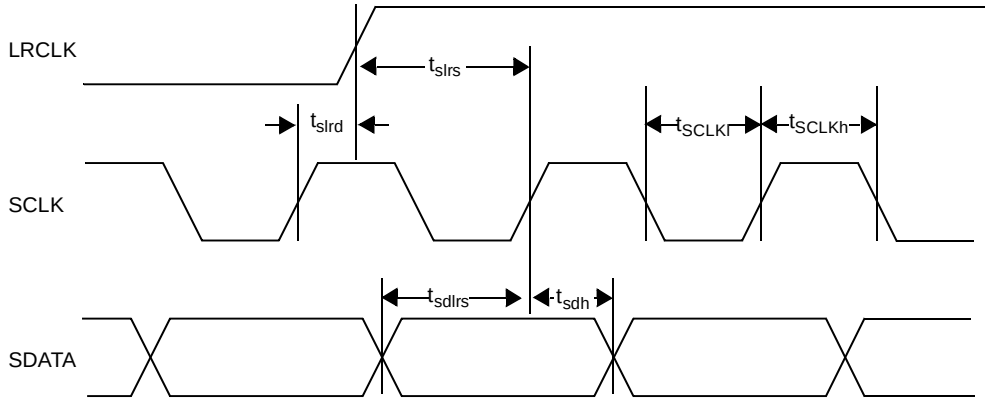


Figure 27 Audio Interface Timing

Table 26 AC Parameters for Audio Signals

Symbol	Parameter	Min
t_{slrd}	LRCLK delay	2ns
t_{slrs}	LRCLK setup	32ns
t_{SCLKl}	bit clock low	22ns
t_{SCLKh}	bit clock high	22ns
t_{sdlrs}	data setup	32ns
t_{sdh}	data hold	2ns

LRCLK

This signal determines which audio channel (left/right) is currently being input on the audio Serial Data input line. LRCLK is low to indicate the left channel and high to indicate the right channel. Typical frequency values for this signal are 48 kHz, 44.1 kHz, 32 kHz, and 22 kHz.

SCLK

This signal is the serial digital audio PCM clock.

SDATA

This signal is the digital PCM signal that carries the audio information. Digital audio data is transferred using the I^2S format.

MCLK

This signal is the Master clock for the digital audio.MCLK is asynchronous to LRCLK, SDATA, and SCLK.

The MCLK must be either 256x or 384x the desired Input Word Rate (IWR). IWR is the frequency at which words for each channel are input to the DAC and is equal to the LRCLK frequency. The following table illustrates several standard audio word rates and the required MCLK and LRCLK frequencies. Typically, most devices operate with 384 Fs master clock.

The ZV Port audio DAC should support a MCLK frequency of 384 Fs. This results in the frequencies shown below.

LRCLK (kHz) Sample Frequency	SCLK (MHz) 32 x Fs	MCLK (MHz) 384x Fs
22	0.704	8.448
32	1.0240	12.2880
44.1	1.4112	16.9344
48	1.5360	18.4320

I²S Format

The I²S format is shown in Figure 28 below. the digital audio data is left channel-MSB justified to the high-to-low going edge of the LRCLK plus one SCLK delay.

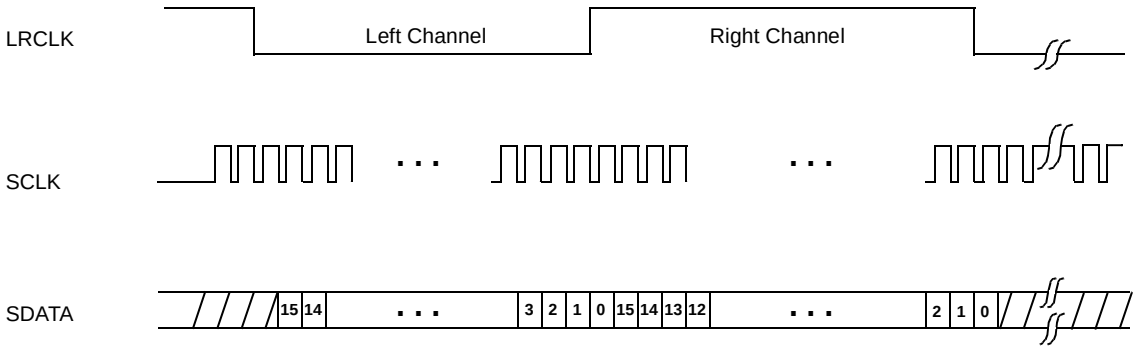


Figure 28 I²S Digital Input Format with 16 SCLK periods

ZV Port Pin Assignments

Table 27 shows the function of various PC Card signals when the ZV Port custom interface mode is set in the PC Card Host Adapter. PC Card signals not mentioned in the table below, remain unchanged from the 16-bit PC Card I/O and Memory interface.

Table 27 ZV Port Interface Pin Assignments

PC Card Pin Number	I/O and Memory Interface Signal Name	I/O and Memory I/O[a]	ZV Port Interface Signal Name	ZV Port I/O[a]	Comments
8	A10	I	HREF	O	Horizontal Sync to ZV Port
10	A11	I	VSYNC	O	Vertical Sync to ZV Port
11	A9	I	Y0	O	Video Data to ZV Port YUV:4:2:2 format
12	A8	I	Y2	O	Video Data to ZV Port YUV:4:2:2 format
13	A13	I	Y4	O	Video Data to ZV Port YUV:4:2:2 format
14	A14	I	Y6	O	Video Data to ZV Port YUV:4:2:2 format
19	A16	I	UV2	O	Video Data to ZV Port YUV:4:2:2 format
20	A15	I	UV4	O	Video Data to ZV Port YUV:4:2:2 format
21	A12	I	UV6	O	Video Data to ZV Port YUV:4:2:2 format
22	A7	I	SCLK	O	Audio SCLK PCM Signal
23	A6	I	MCLK	O	Audio MCLK PCM Signal
24:25	A[5:4]	I	RESERVED	RFU	Put in three state by Host Adapter No connection in PC Card
26:29	A[3:0]	I	ADDRESS[3:0]	I	Used for accessing PC Card
33	IOIS16#	O	PCLK	O	Pixel Clock to ZV Port
46	A17	I	Y1	O	Video Data to ZV Port YUV:4:2:2 format
47	A18	I	Y3	O	Video Data to ZV Port YUV:4:2:2 format
48	A19	I	Y5	O	Video Data to ZV Port YUV:4:2:2 format
49	A20	I	Y7	O	Video Data to ZV Port YUV:4:2:2 format
50	A21	I	UV0	O	Video Data to ZV Port YUV:4:2:2 format
53	A22	I	UV1	O	Video Data to ZV Port YUV:4:2:2 format
54	A23	I	UV3	O	Video Data to ZV Port YUV:4:2:2 format
55	A24	I	UV5	O	Video Data to ZV Port YUV:4:2:2 format
56	A25	I	UV7	O	Video Data to ZV Port YUV:4:2:2 format
60	INPACK#	O	LRCLK	O	Audio LRCLK PCM Signal
62	SPKR#	O	SDATA	O	Audio PCM Data Signal

a. “I” indicates signal is input to PC Card, “O” indicates signal is output from PC Card

APPENDIX D: SCHEMATIC EXAMPLES

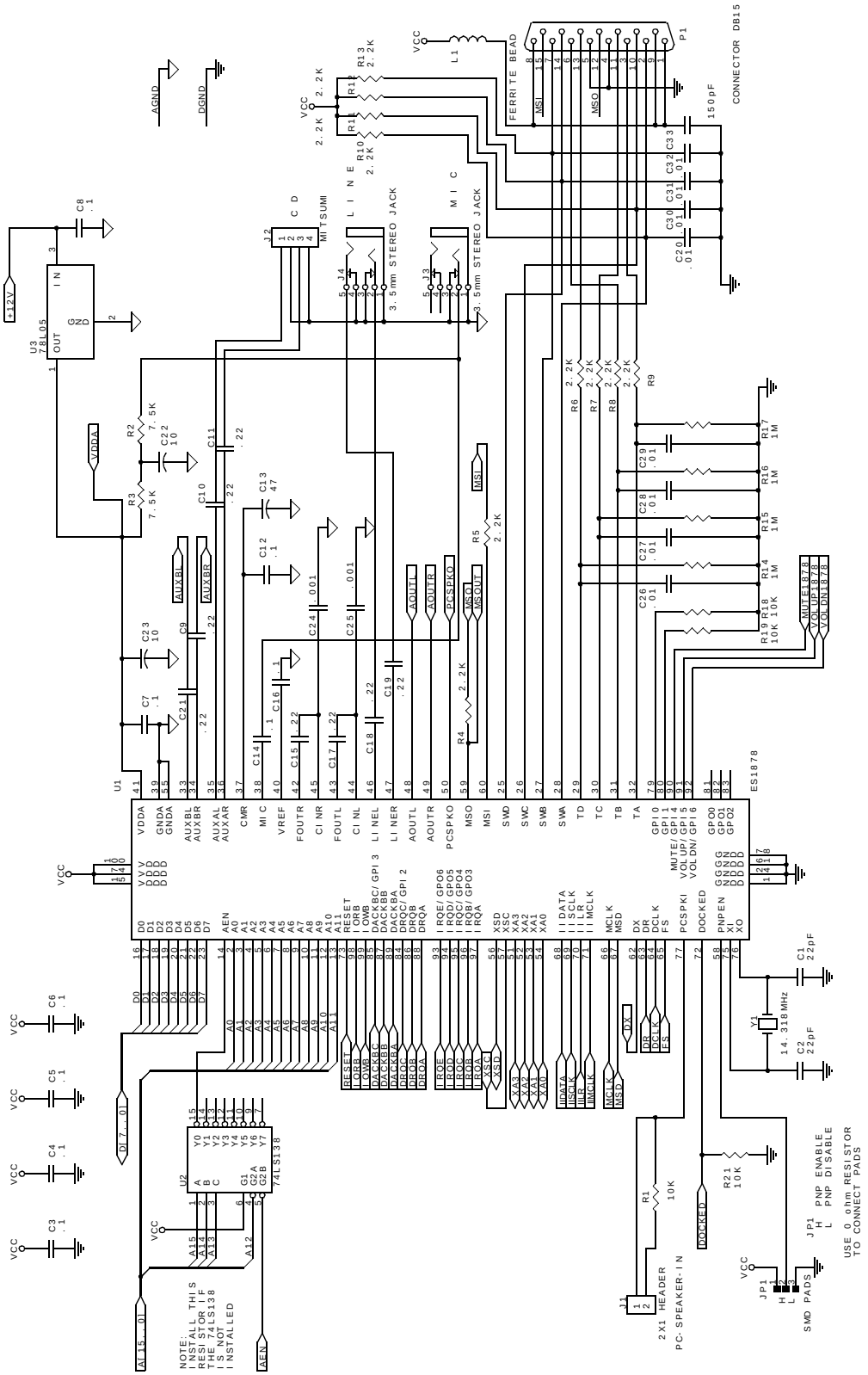


Figure 29 ES1878 Schematic

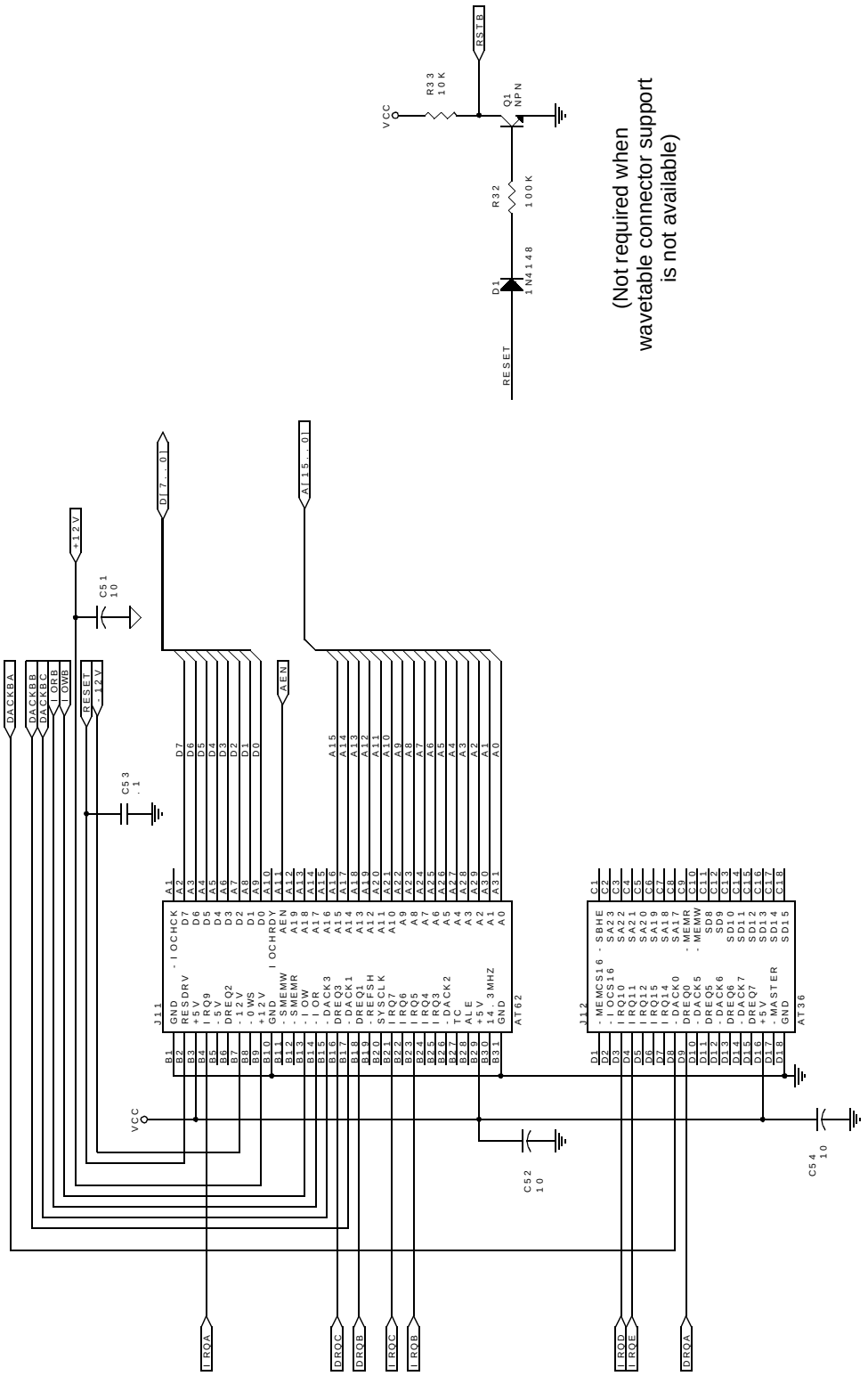


Figure 30 PC Interface

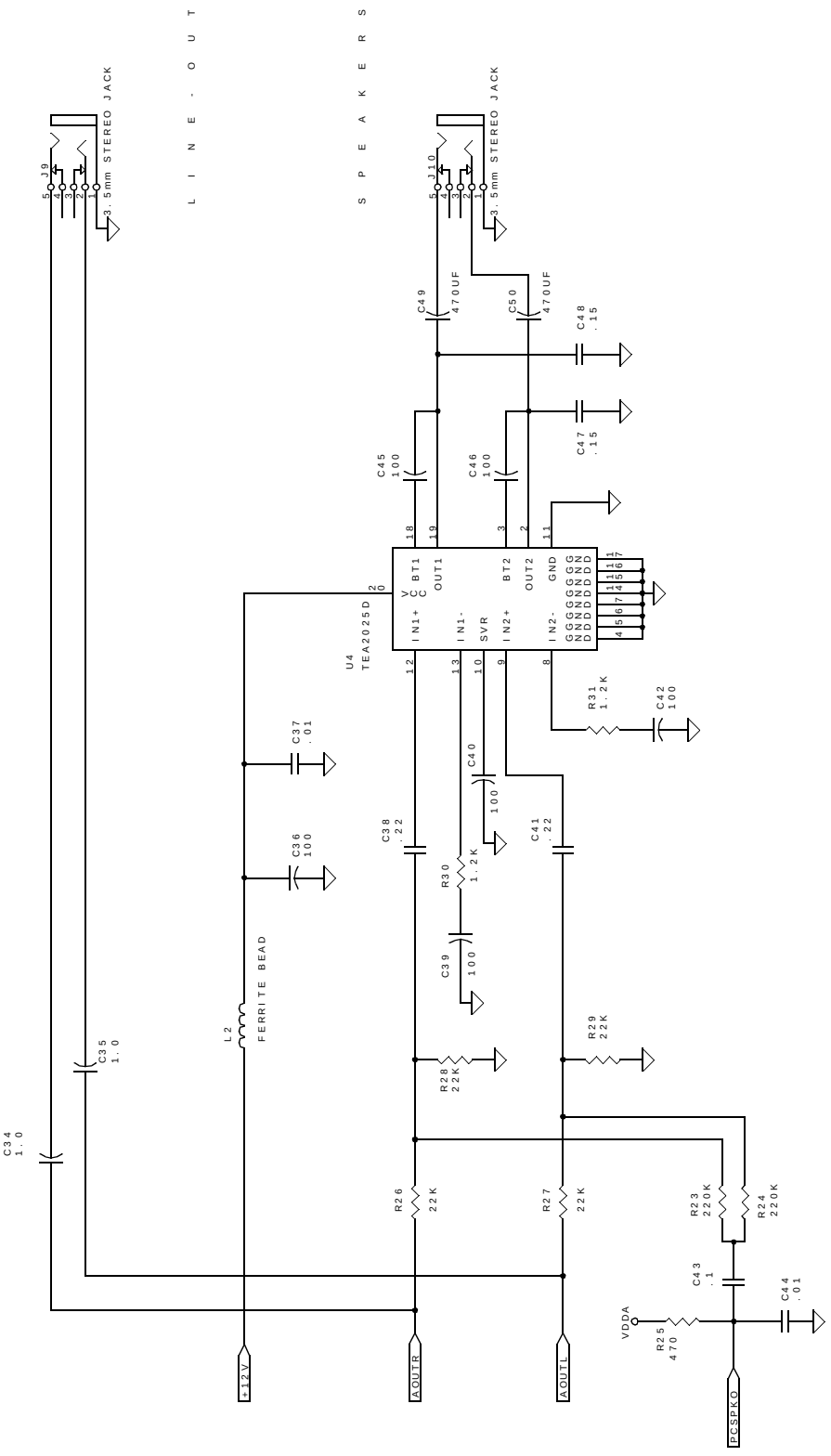
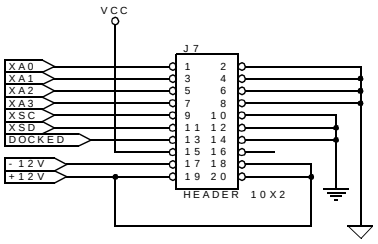
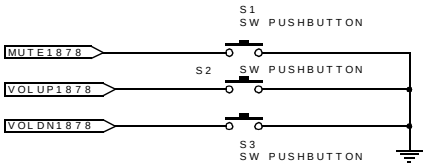


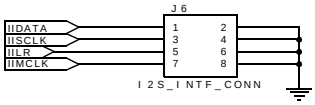
Figure 31 Amplifier Section



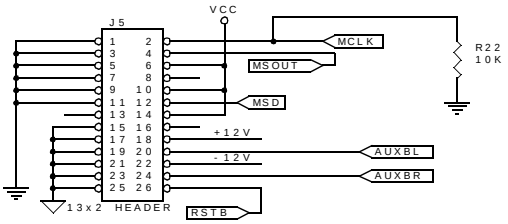
Notebook to Expansion Audio Interface



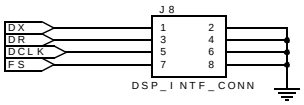
Hardware Volume Control Buttons



I²S Zoom Video Interface



Wavetable



Modem DSP Interface

Figure 32 Switch and Connector Section

APPENDIX E: LAYOUT GUIDELINES

PCB Layout

Notebook, Motherboard, Pen-based, and PDA portable computers have the following similarity in PCB layout design:

6 Multi-layer (usually 4 to 8 layer).
7 Double-sided SMT.
8 CPU, corelogic (chip set), system memory, VGA controller, and video memory reside in the same PCB.

This is a very noisy environment for adding an audio circuit. The following are the guidelines for PCB layout for ESS *Audio***Drive**® chip application.

Component Placement

The audio circuit-related components must be grouped in the same area. The audio I/O jack and connector are considered audio-related components as well. There are two possible placements for these audio components:

A grouped on one side of the PCB.
B separated on both sides of the PCB.

In Case B, audio component grouping will take less space.

Analog Ground Plane

Audio circuits require two layers of analog ground planes for use as shielding for all analog traces.

In component placement case A (Figure 33), the first layer of analog ground plane is on the analog component side, the second analog ground plane is on the inner layer, and the analog traces are embedded between these two planes.

In component placement case B (Figure 34), the analog ground planes are on both sides of the PCB, with the analog traces shielded in the middle.

Case A:

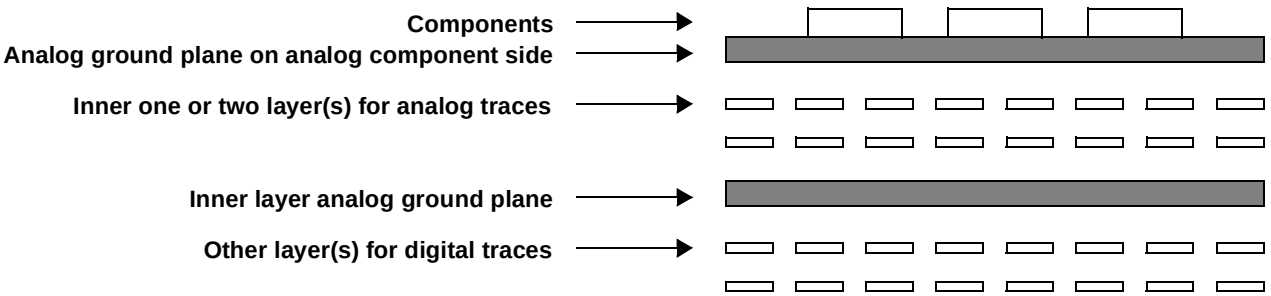


Figure 33 Analog Components on One Side of the PCB

Case B:

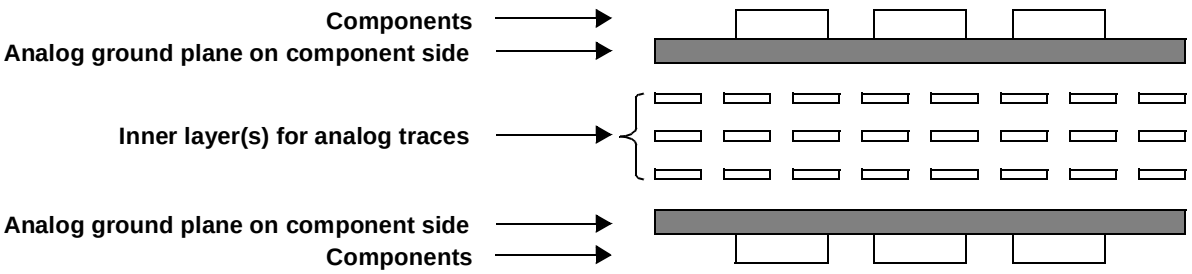


Figure 34 Analog Components on Both Sides of the PCB.

Special Notes

The analog traces should be places as short as possible.

The MIC-IN circuit is the most sensitive of the audio circuits, and requires proper and complete shielding.

APPENDIX F: ES1878 BILL OF MATERIALS

Table 28 ES1878 Bill of Materials (BOM)

Item	Quantity	Reference	Part
1	2	C1,C2	22 pF
2	11	C3,C4,C5,C6,C7,C8,C12,C14,C16,C43,C53	.1 µF
3	10	C9,C10,C11,C15,C17,C18,C19,C21,C38,C41	.22 µF
4	1	C13	47 µF
5	10	C20,C26,C27,C28,C29,C30,C31,C32,C37,C44	.01 µF
6	5	C22,C23,C51,C52,C54	10 µF
7	2	C24,C25	.001 µF
8	1	C33	150 pF
9	2	C34,C35	1.0 µF
10	6	C36,C39,C40,C42,C45,C46	100 µF
11	2	C47,C48	.15 µF
12	2	C49,C50	470 µF
13	1	D1	1N4148
14	1	JP1 (H PnP Enable) (L PnP Disable)	0 ohm Resistor
15	1	J1	2X1 HEADER
16	1	J2	MITSUMI
17	4	J3,J4,J9,J10	3.5mm STEREO JACK
18	1	J5	13x2 HEADER
19	1	J6	4X2 HEADER I2S_INTF_CONN
20	1	J7	HEADER 10X2
21	1	J8	4X2 HEADER DSP_INTF_CONN
22	2	L1,L2	FERRITE BEAD
23	1	P1	CONNECTOR DB15
24	1	Q1	NPN
25	6	R1,R18,R19,R21,R22,R33	10K
26	2	R2,R3	7.5K
27	10	R4,R5,R6,R7,R8,R9,R10,R11,R12,R13	2.2K
28	4	R14,R15,R16,R17	1M
29	2	R23,R24	220K
30	1	R25	470
31	4	R26,R27,R28,R29	22K
32	2	R30,R31	1.2K
33	1	R32	100K
34	3	S1,S2,S3	SW PUSHBUTTON
35	1	U1	ES1878
36	1	U2	74LS138
37	1	U3	78L05
38	1	U4	TEA2025D
39	1	Y1	14.318 MHz

ESS Technology, Inc.
48401 Fremont Blvd.
Fremont, CA 94538
Tel: 510-226-1088
Fax: 510-492-1098