Features

- **MPEG I/II-Layer 3 Hardwired Decoder**
  - **Stand-alone MP3 Decoder**
  - **48, 44.1, 32, 24, 22.05, 16 kHz Sampling Frequency**
  - **Separated Digital Volume Control on Left and Right Channels (Software Control using 31 Steps)**
  - **Bass, Medium, and Treble Control (31 Steps)**
  - **Bass Boost Sound Effect**
  - **Ancillary Data Extraction**
  - **CRC Error and MPEG Frame Synchronization Indicators**
- **20-bit Stereo Audio DAC**
  - **93 dB SNR Playback Stereo Channel**
  - **32 Ohm/ 20 mW Stereo Headset Drivers**
  - **Stereo Line Level Input, Differential Mono Auxiliary Input**
- **Programmable Audio Output for Interfacing with External Audio System**
  - **I^2S Format Compatible**
- **Mono Audio Power Amplifier**
  - **440mW on 8 Ohms Load**
- **USB Rev 1.1 Controller**
  - **Full Speed Data Transmission**
- **Built-in PLL**
  - **MP3 Audio Clocks**
  - **USB Clock**
- **MultiMediaCard® Interface, Secure Digital Card Interface**
- **Standard Full Duplex UART with Baud Rate Generator**
- **Power Management**
  - **Power-on Reset**
  - **Idle Mode, Power-down Mode**
- **Operating Conditions:**
  - **2.7 to 3V, ±10%, 25 mA Typical Operating at 25°C**
  - **37 mA Typical Operating at 25°C Playing Music on Earphone**
  - **Temperature Range: -40°C to +85°C**
  - **Power Amplifier Supply 3.2V to 5.5V**
- **Packages**
  - **CTBGA 100-pin**

Typical Applications

- MP3-Player
- PDA, Camera, Mobile Phone MP3
- Car Audio/Multimedia MP3
- Home Audio/Multimedia MP3
- Toys
- Industrial Background Music / Ads

Single-Chip MP3 Decoder with Full Audio Interface

AT83SND2CMP3A1
AT83SND2CMP3
AT83SND2CDVX

Rev. 7524D–MP3–07/07

Description

The AT83SND2CMP3 has been developped as a versatile remote controlled MP3 player for very fast MP3 feature implementation into most existing system. It perfectly fits features needed in mobile phones and toys, but can also be used in any portable equipment and in industrial applications.

Audio files and any other data can be stored in a Nand Flash memory or in a removable Flash card such as MultiMediaCard (MMC) or Secure Digital Card (SD). Music collections are very easy to build, as data can be stored using the standard FAT12/16 and FAT32 file system.

Thanks to the USB port, data can be transferred and maintained from and to any computer based on Windows®, Linux® and Mac OS®.

File system is controlled by the AT83SND2CMP3 so the host controller does not have to handle it.

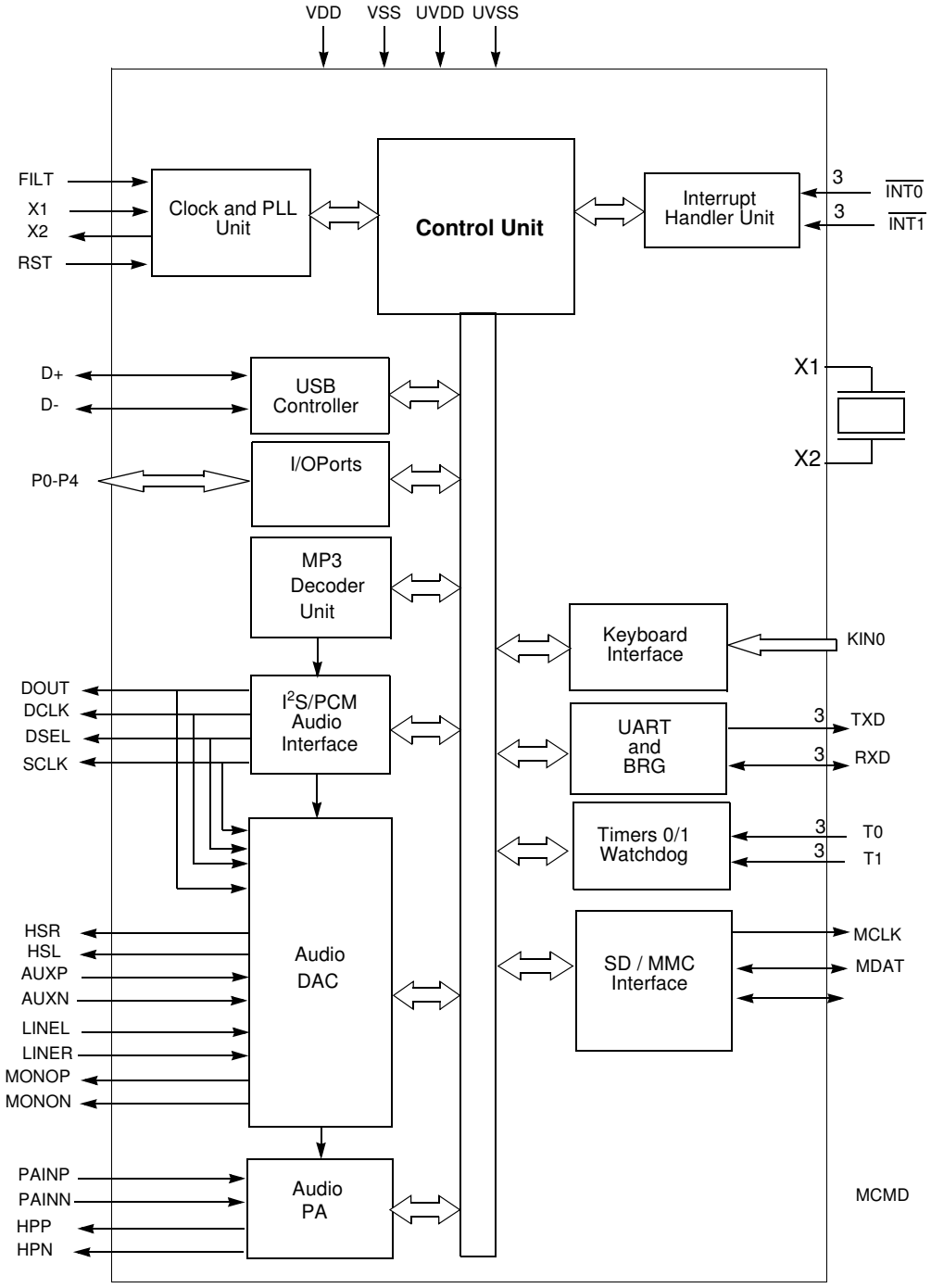
In addition to the USB device port, the MP3 audio system can be connected to any embedded host through a low cost serial link UART. Host controller can fully remote control the MP3 decoder behaviour using a command protocol over the serial link.

File system is controlled by the AT83SND2CMP3 so host controller does not have to handle it.

Files can also be uploaded or dowloaded from host environment to NAND Flash or Flash Card.

Block Diagram

Figure 1. Block Diagram

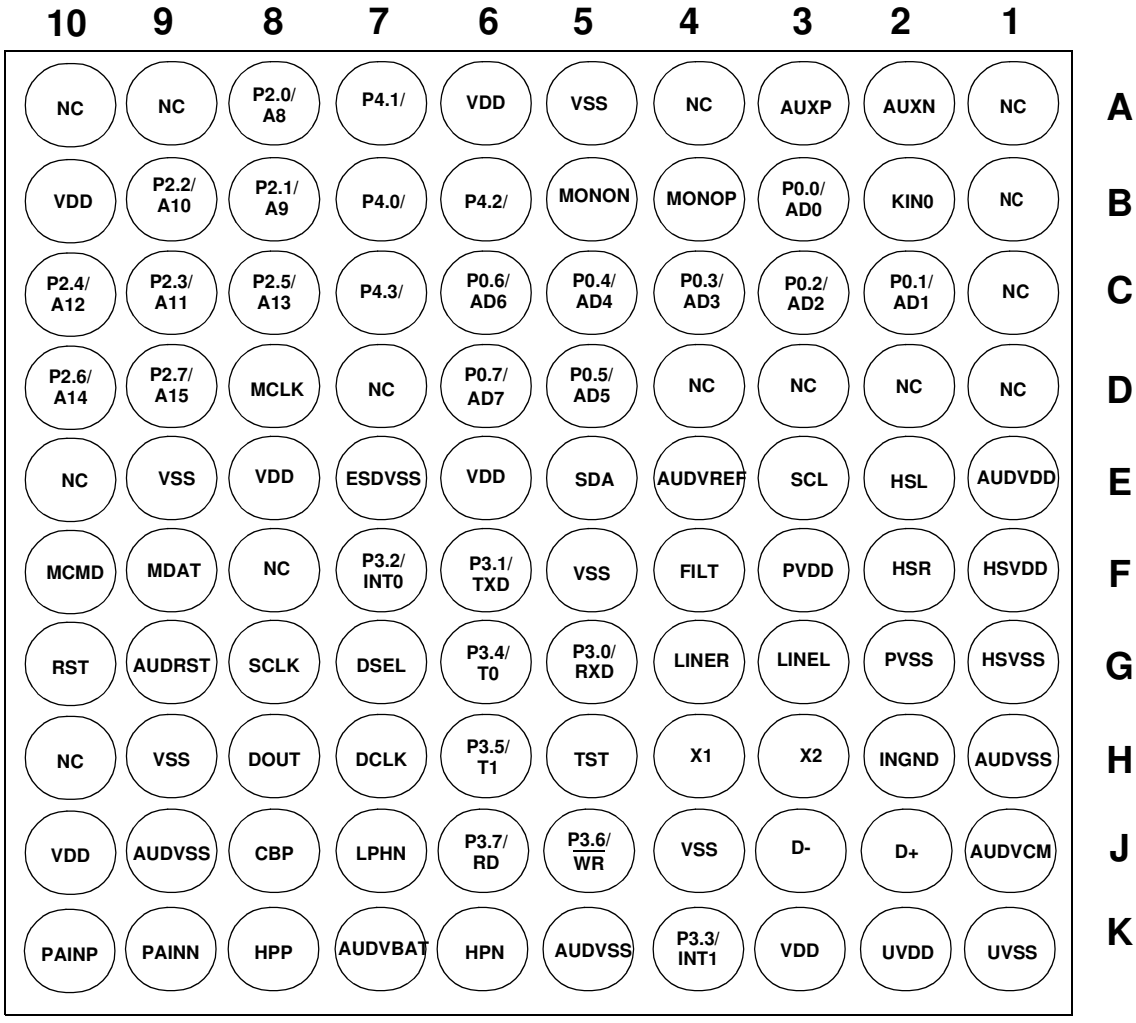


3 Alternate function of Port 3
4 Alternate function of Port 4

Pin Description

Pinouts

Figure 3. AT83SND2CMP3 100-pin BGA Package



1. NC = Do Not Connect

Signals

All the AT83SND2CMP3 signals are detailed by functionality in following tables.

Table 1. Ports Signal Description

Signal Name	Type	Description	Alternate Function
P0.7:0	I/O	**Port 0** P0 is an 8-bit open-drain bidirectional I/O port. Port 0 pins that have 1s written to them float and can be used as high impedance inputs. To avoid any parasitic current consumption, floating P0 inputs must be polarized to V_{DD} or V_{SS}.	AD7:0
P2.7:0	I/O	**Port 2** P2 is an 8-bit bidirectional I/O port with internal pull-ups.	A15:8
P3.7:0	I/O	**Port 3** P3 is an 8-bit bidirectional I/O port with internal pull-ups.	RXD TXD $\overline{INT0}$ $\overline{INT1}$ T0 T1 $\overline{WR}$ $\overline{RD}$
P4.3:0	I/O	**Port 4** P4 is an 8-bit bidirectional I/O port with internal pull-ups.	

Table 2. Clock Signal Description

Signal Name	Type	Description	Alternate Function
X1	I	**Input to the on-chip inverting oscillator amplifier** To use the internal oscillator, a crystal/resonator circuit is connected to this pin. If an external oscillator is used, its output is connected to this pin. X1 is the clock source for internal timing.	-
X2	O	**Output of the on-chip inverting oscillator amplifier** To use the internal oscillator, a crystal/resonator circuit is connected to this pin. If an external oscillator is used, leave X2 unconnected.	-
FILT	I	**PLL Low Pass Filter input** FILT receives the RC network of the PLL low pass filter.	-

Table 3. Timer 0 and Timer 1 Signal Description

Signal Name	Type	Description	Alternate Function
$\overline{INT0}$	I	**Timer 0 Gate Input** $\overline{INT0}$ serves as external run control for timer 0, when selected by GATE0 bit in TCON register. **External Interrupt 0** $\overline{INT0}$ input sets IE0 in the TCON register. If bit IT0 in this register is set, bit IE0 is set by a falling edge on INT0#. If bit IT0 is cleared, bit IE0 is set by a low level on INT0#.	P3.2

Signal Name	Type	Description	Alternate Function
$\overline{INT1}$	I	**Timer 1 Gate Input** $\overline{INT1}$ serves as external run control for timer 1, when selected by GATE1 bit in TCON register. **External Interrupt 1** $\overline{INT1}$ input sets IE1 in the TCON register. If bit IT1 in this register is set, bit IE1 is set by a falling edge on INT1#. If bit IT1 is cleared, bit IE1 is set by a low level on INT1#.	P3.3
T0	I	**Timer 0 External Clock Input** When timer 0 operates as a counter, a falling edge on the T0 pin increments the count.	P3.4
T1	I	**Timer 1 External Clock Input** When timer 1 operates as a counter, a falling edge on the T1 pin increments the count.	P3.5

Table 4. Audio Interface Signal Description

Signal Name	Type	Description	Alternate Function
DCLK	O	**DAC Data Bit Clock**	-
DOUT	O	**DAC Audio Data Output**	-
DSEL	O	**DAC Channel Select Signal** DSEL is the sample rate clock output.	-
SCLK	O	**DAC System Clock** SCLK is the oversampling clock synchronized to the digital audio data (DOUT) and the channel selection signal (DSEL).	-

Table 5. USB Controller Signal Description

Signal Name	Type	Description	Alternate Function
D+	I/O	**USB Positive Data Upstream Port** This pin requires an external 1.5 KΩ pull-up to V_{DD} for full speed operation.	-
D-	I/O	**USB Negative Data Upstream Port**	-

Table 6. MutiMediaCard Interface Signal Description

Signal Name	Type	Description	Alternate Function
MCLK	O	**MMC Clock output** Data or command clock transfer.	-
MCMD	I/O	**MMC Command line** Bidirectional command channel used for card initialization and data transfer commands. To avoid any parasitic current consumption, unused MCMD input must be polarized to V_{DD} or V_{SS}.	-
MDAT	I/O	**MMC Data line** Bidirectional data channel. To avoid any parasitic current consumption, unused MDAT input must be polarized to V_{DD} or V_{SS}.	-

Table 7. UART Signal Description

Signal Name	Type	Description	Alternate Function
RXD	I/O	**Receive Serial Data** RXD sends and receives data in serial I/O mode 0 and receives data in serial I/O modes 1, 2 and 3.	P3.0
TXD	O	**Transmit Serial Data** TXD outputs the shift clock in serial I/O mode 0 and transmits data in serial I/O modes 1, 2 and 3.	P3.1

Table 8. Keypad Interface Signal Description

Signal Name	Type	Description	Alternate Function
KIN0	I	**Keypad Input Line** Holding this pin high or low for 24 oscillator periods triggers a keypad interrupt.	-

Table 9. System Signal Description

Signal Name	Type	Description	Alternate Function
RST	I	**Reset Input** Holding this pin high for 64 oscillator periods while the oscillator is running resets the device. The Port pins are driven to their reset conditions when a voltage lower than V_{IL} is applied, whether or not the oscillator is running. This pin has an internal pull-down resistor which allows the device to be reset by connecting a capacitor between this pin and V_{DD}. Asserting RST when the chip is in Idle mode or Power-Down mode returns the chip to normal operation.	-
$\overline{TST}$	I	**Test Input** Test mode entry signal. This pin must be set to V_{DD}.	-

Table 10. Power Signal Description

Signal Name	Type	Description	Alternate Function
VDD	PWR	**Digital Supply Voltage** Connect these pins to +3V supply voltage.	-
VSS	GND	**Circuit Ground** Connect these pins to ground.	-
PVDD	PWR	**PLL Supply voltage** Connect this pin to +3V supply voltage.	-
PVSS	GND	**PLL Circuit Ground** Connect this pin to ground.	-
UVDD	PWR	**USB Supply Voltage** Connect this pin to +3V supply voltage.	-

Signal Name	Type	Description	Alternate Function
UVSS	GND	**USB Ground** Connect this pin to ground.	-

Table 11. Audio Power Signal Description

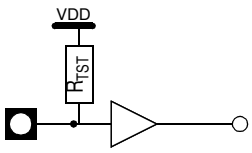
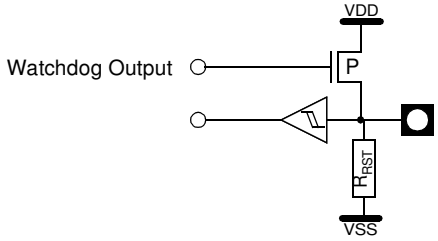
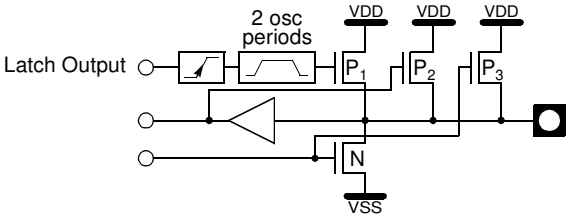
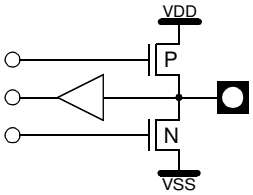
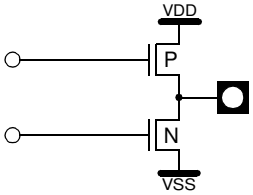
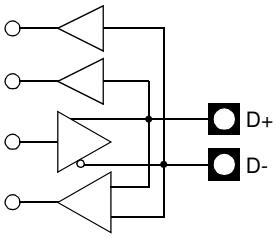
Signal Name	Type	Description	Alternate Function
AUDVDD	PWR	**Audio Digital Supply Voltage**	-
AUDVSS	GND	**Audio Circuit Ground** Connect these pins to ground.	-
ESDVSS	GND	**Audio Analog Circuit Ground for Electrostatic Discharge**. Connect this pin to ground.	-
AUDVREF	PWR	**Audio Voltage Reference pin for decoupling**.	-
HSVDD	PWR	**Headset Driver Power Supply**.	-
HSVSS	GND	**Headset Driver Ground.** Connect this pin to ground.	-
AUDVBAT	PWR	**Audio Amplifier Supply.**	-

Table 12. Stereo Audio Dac and Mono Power Amplifier Signal Description

Signal Name	Type	Description	Alternate Function
LPHN	O	**Low Power Audio Stage Output**	-
HPN	O	**Negative Speaker Output**	-
HPP	O	**Positivie Speaker Output**	-
CBP	O	**Audio Amplifier Common Mode Voltage Decoupling**	-
PAINN	I	**Audio Amplifier Negative Input**	-
PAINP	I	**Audio Amplifier Positive Input**	-
AUDRST	I	**Audio Reset (Active Low)**	-
MONON	O	**Audio Negative Monaural Driver Output**	-
MONOP	O	**Audio Positive Monaural Driver Outpu**t	-
AUXP	I	**Audio Mono Auxiliary Positive Input**	-
AUXN	I	**Audio Mono Auxiliary Negative Input**	-
HSL	O	**Audio Left Channel Headset Driver Output**	-
HSR	O	**Audio Right Channel Headset Driver Output**	-
LINEL	I	**Audio Left Channel Line In**	-
LINER	I	**Audio Right Channel Line In**	-
INGND	I	**Audio Line Signal Ground Pin for decoupling**.	-
AUDVCM	I	**Audio Common Mode reference for decoupling**	-

Internal Pin Structure

Table 13. Detailed Internal Pin Structure

Circuit[1]	Type	Pins
	Input	$\overline{TST}$
	Input/Output	RST
	Input/Output	P3 P4
	Input/Output	P0 MCMD MDAT
	Output	ALE SCLK DCLK DOUT DSEL MCLK
	Input/Output	D+ D-

Notes: 1. For information on resistors value, input/output levels, and drive capability, refer to the DC Characteristics.
2. When the Two Wire controller is enabled, P_3 transistors are disabled allowing pseudo open-drain structure.

Clock Controller

The clock controller is based on an on-chip oscillator feeding an on-chip Phase Lock Loop (PLL). All internal clocks to the peripherals and CPU core are generated by this controller.

Oscillator

The X1 and X2 pins are the input and the output of a single-stage on-chip inverter (see Figure 4) that can be configured with off-chip components such as a Pierce oscillator (see Figure 5). Value of capacitors and crystal characteristics are detailed in the section “DC Characteristics”.

The oscillator outputs three different clocks: a clock for the PLL, a clock for the CPU core, and a clock for the peripherals as shown in Figure 4. These clocks are either enabled or disabled, depending on the power reduction mode as detailed in the section. The peripheral clock is used to generate the Timer 0, Timer 1, MMC, SPI, and Port sampling clocks.

Figure 4. Oscillator Block Diagram and Symbol

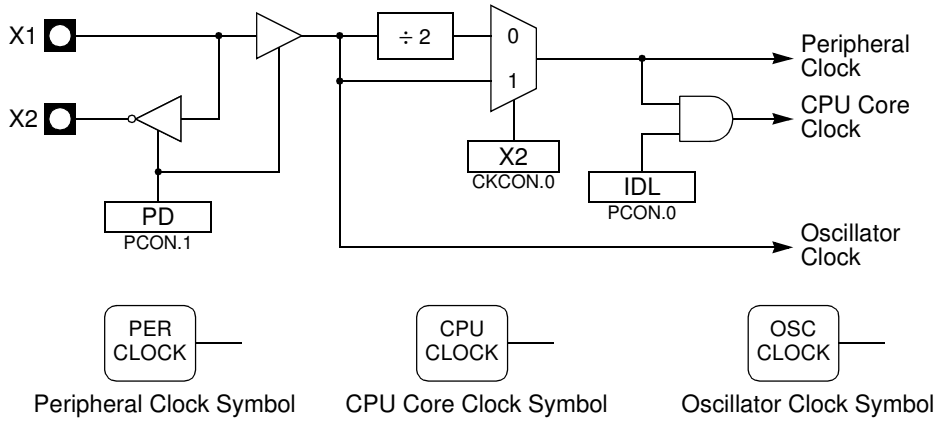
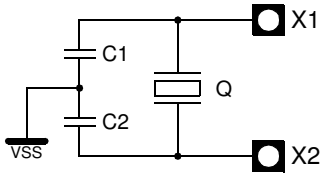


Figure 5. Crystal Connection



PLL

PLL Description

The PLL is used to generate internal high frequency clock (the PLL Clock) synchronized with an external low-frequency (the Oscillator Clock). The PLL clock provides the MP3 decoder, the audio interface, and the USB interface clocks. Figure 6 shows the internal structure of the PLL.

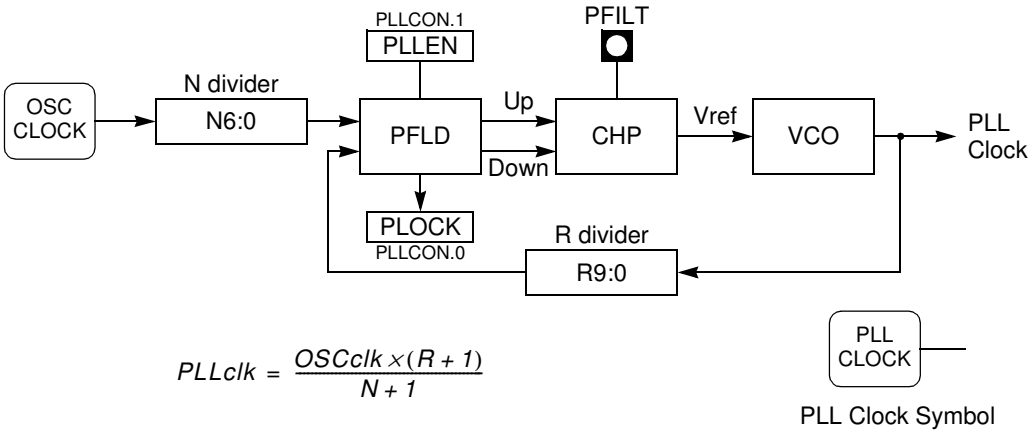
The PFLD block is the Phase Frequency Comparator and Lock Detector. This block makes the comparison between the reference clock coming from the N divider and the reverse clock coming from the R divider and generates some pulses on the Up or Down signal depending on the edge position of the reverse clock. The PLLEN bit in PLLCON register is used to enable the clock generation.

The CHP block is the Charge Pump that generates the voltage reference for the VCO by injecting or extracting charges from the external filter connected on PFILT pin (see

Figure 7). Value of the filter components are detailed in the Section “DC Characteristics”.

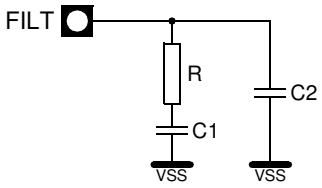
The VCO block is the Voltage Controlled Oscillator controlled by the voltage V_{ref} produced by the charge pump. It generates a square wave signal: the PLL clock.

Figure 6. PLL Block Diagram and Symbol



$$PLLclk = \frac{OSCclk \times (R + 1)}{N + 1}$$

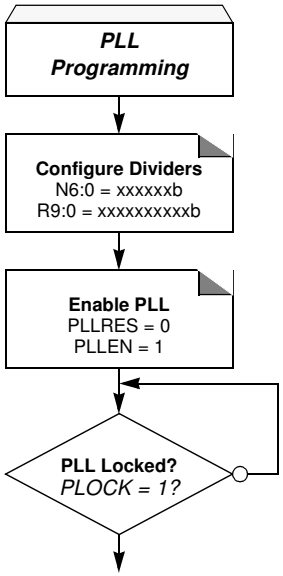
Figure 7. PLL Filter Connection



PLL Programming

The PLL is programmed using the flow shown in Figure 8. The PLL clock frequency will depend on MP3 decoder clock and audio interface clock frequencies.

Figure 8. PLL Programming Flow



MP3 Decoder

The product implements a MPEG I/II audio layer 3 decoder better known as MP3 decoder.

In MPEG I (ISO 11172-3) three layers of compression have been standardized supporting three sampling frequencies: 48, 44.1, and 32 kHz. Among these layers, layer 3 allows highest compression rate of about 12:1 while still maintaining CD audio quality. For example, 3 minutes of CD audio (16-bit PCM, 44.1 kHz) data, which needs about 32M bytes of storage, can be encoded into only 2.7M bytes of MPEG I audio layer 3 data.

In MPEG II (ISO 13818-3), three additional sampling frequencies: 24, 22.05, and 16 kHz are supported for low bit rates applications.

The AT83SND2CMP3 can decode in real-time the MPEG I audio layer 3 encoded data into a PCM audio data, and also supports MPEG II audio layer 3 additional frequencies.

Additional features are supported by the AT83SND2CMP3 MP3 decoder such as volume control, bass, medium, and treble controls, bass boost effect and ancillary data extraction.

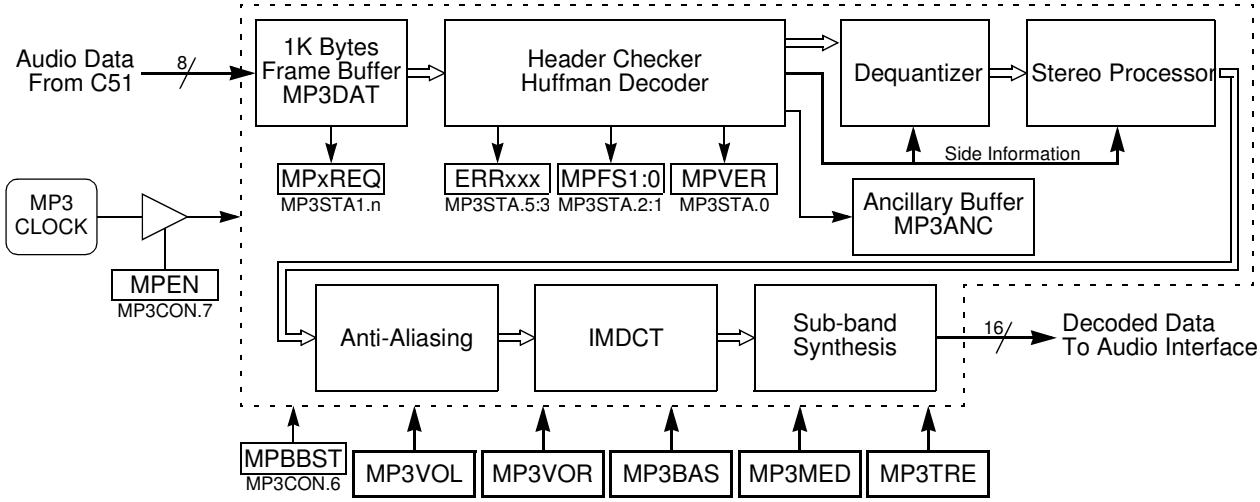
Decoder

Description

The core interfaces to the MP3 decoder through nine special function registers: MP3CON, the MP3 Control register; MP3STA, the MP3 Status register; MP3DAT, the MP3 Data register; MP3ANC, the Ancillary Data register; MP3VOL and MP3VOR, the MP3 Volume Left and Right Control registers; MP3BAS, MP3MED, and MP3TRE, the MP3 Bass, Medium, and Treble Control registers; and MPCLK, the MP3 Clock Divider register.

Figure 9 shows the MP3 decoder block diagram.

Figure 9. MP3 Decoder Block Diagram

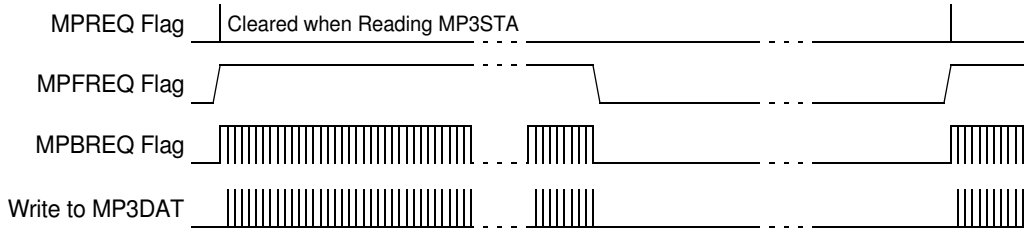


MP3 Data

The MP3 decoder does not start any frame decoding before having a complete frame in its input buffer[1]. In order to manage the load of MP3 data in the frame buffer, a hardware handshake consisting of data request and data acknowledgment is implemented. Each time the MP3 decoder needs MP3 data, it sets the MPREQ, MPFREQ and MPBREQ flags respectively in MP3STA and MP3STA1 registers. MPREQ flag can generate an interrupt if enabled as explained in Section “Interrupt”. The CPU must then load data in the buffer by writing it through MP3DAT register thus acknowledging the previous request. As shown in Figure 10, the MPFREQ flag remains set while data (i.e a frame) is requested by the decoder. It is cleared when no more data is requested and set again when new data are requested. MPBREQ flag toggles at every Byte writing.

Note: 1. The first request after enable, consists in 1024 Bytes of data to fill in the input buffer.

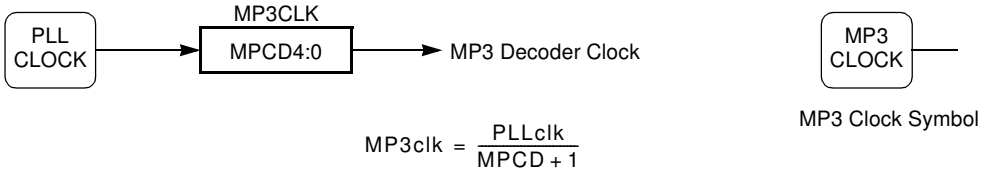
Figure 10. Data Timing Diagram



MP3 Clock

The MP3 decoder clock is generated by division of the PLL clock. The division factor is given by MPCD4:0 bits in MP3CLK register. Figure 11 shows the MP3 decoder clock generator and its calculation formula. The MP3 decoder clock frequency depends only on the incoming MP3 frames.

Figure 11. MP3 Clock Generator and Symbol



$$MP3clk = \frac{PLLclk}{MPCD + 1}$$

As soon as the frame header has been decoded and the MPEG version extracted, the minimum MP3 input frequency must be programmed according to Table 14.

Table 14. MP3 Clock Frequency

MPEG Version	Minimum MP3 Clock (MHz)
I	21
II	10.5

Audio Controls

Volume Control

The MP3 decoder implements volume control on both right and left channels. The MP3VOR and MP3VOL registers allow a 32-step volume control according to Table 15.

Table 15. Volume Control

VOL4:0 or VOR4:0	Volume Gain (dB)
00000	Mute
00001	-33
00010	-27
11110	-1.5
11111	0

Equalization Control

Sound can be adjusted using a 3-band equalizer: a bass band under 750 Hz, a medium band from 750 Hz to 3300 Hz and a treble band over 3300 Hz. The MP3BAS, MP3MED, and MP3TRE registers allow a 32-step gain control in each band according to Table 16.

Table 16. Bass, Medium, Treble Control

BAS4:0 or MED4:0 or TRE4:0	Gain (dB)
00000	- ∞
00001	-14
00010	-10
11110	+1
11111	+1.5

Frame Information

The MP3 frame header contains information on the audio data contained in the frame. These informations is made available in the MP3STA register for you information. MPVER and MPFS1:0 bits allow decoding of the sampling frequency according to Table 17. MPVER bit gives the MPEG version (2 or 1).

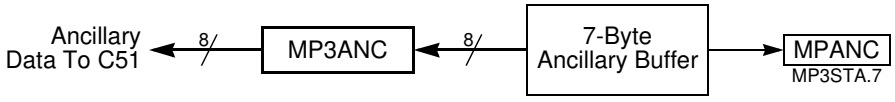
Table 17. MP3 Frame Frequency Sampling

MPVER	MPFS1	MPFS0	Fs (kHz)
0	0	0	22.05 (MPEG II)
0	0	1	24 (MPEG II)
0	1	0	16 (MPEG II)
0	1	1	Reserved
1	0	0	44.1 (MPEG I)
1	0	1	48 (MPEG I)
1	1	0	32 (MPEG I)
1	1	1	Reserved

Ancillary Data

MP3 frames also contain data bits called ancillary data. These data are made available in the MP3ANC register for each frame. As shown in Figure 12, the ancillary data are available by Bytes when MPANC flag in MP3STA register is set. MPANC flag is set when the ancillary buffer is not empty (at least one ancillary data is available) and is cleared only when there is no more ancillary data in the buffer. This flag can generate an interrupt as explained in Section “Interrupt”. When set, software must read all Bytes to empty the ancillary buffer.

Figure 12. Ancillary Data Block Diagram



Audio Output Interface

The product implements an audio output interface allowing the audio bitstream to be output in various formats. It is compatible with right and left justification PCM and I^2S formats and thanks to the on-chip PLL (see Section “Clock Controller”, page 10) allows connection of almost all of the commercial audio DAC families available on the market. The audio bitstream can be from 2 different types:

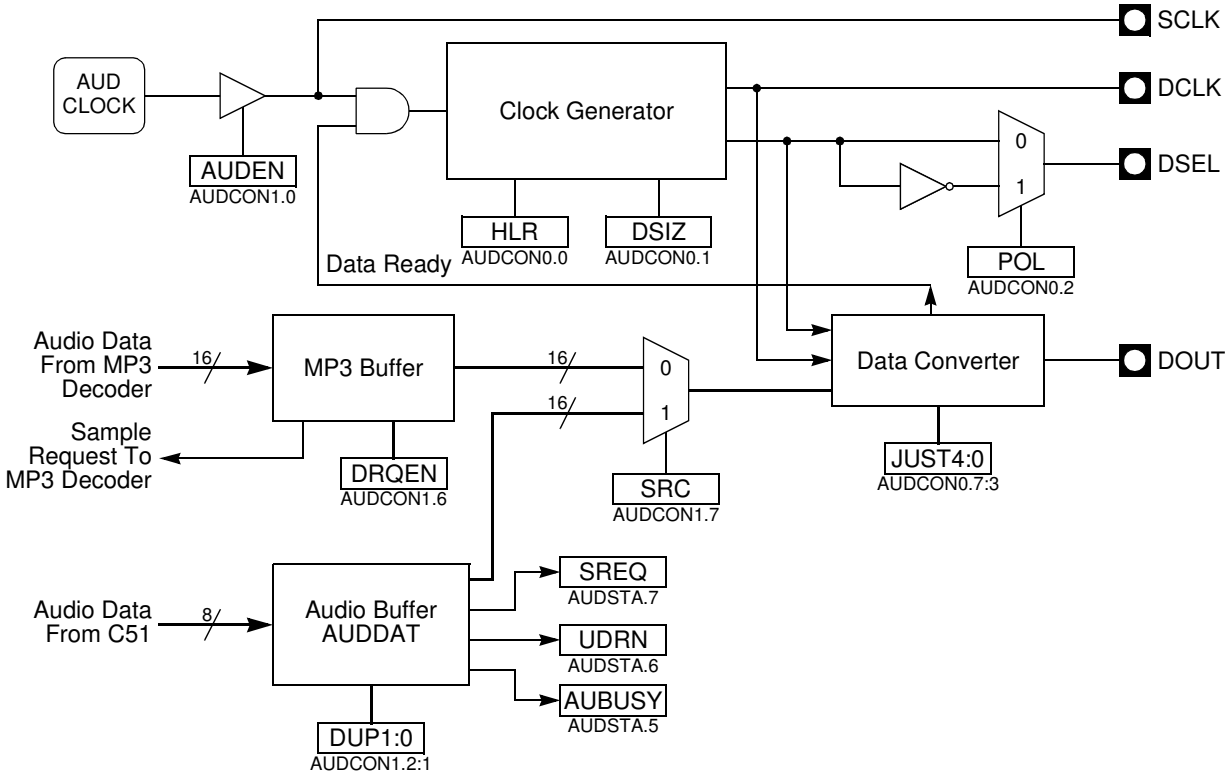
- The MP3 decoded bitstream coming from the MP3 decoder for playing songs.
- The audio bitstream coming from the MCU for outputting voice or sounds.

Description

The control unit core interfaces to the audio interface through five special function registers: AUDCON0 and AUDCON1, the Audio Control registers ; AUDSTA, the Audio Status register; AUDDAT, the Audio Data register; and AUDCLK, the Audio Clock Divider register.

Figure 13 shows the audio interface block diagram, blocks are detailed in the following sections.

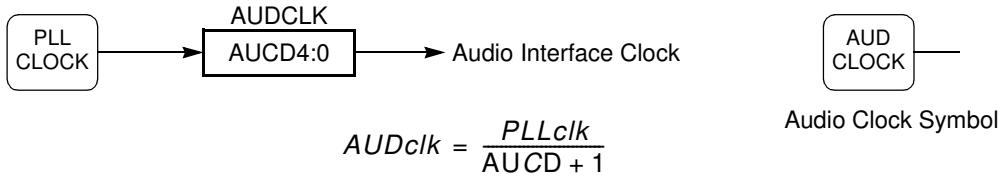
Figure 13. Audio Interface Block Diagram



Clock Generator

The audio interface clock is generated by division of the PLL clock. The division factor is given by AUCD4:0 bits in CLKAUD register. Figure 14 shows the audio interface clock generator and its calculation formula. The audio interface clock frequency depends on the incoming MP3 frames and the audio DAC used.

Figure 14. Audio Clock Generator and Symbol

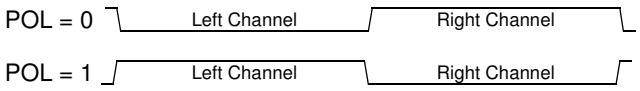


$$AUDclk = \frac{PLLclk}{AUCD + 1}$$

As soon as audio interface is enabled by setting AUDEN bit in AUDCON1 register, the master clock generated by the PLL is output on the SCLK pin which is the DAC system clock. This clock is output at 256 or 384 times the sampling frequency depending on the DAC capabilities. HLR bit in AUDCON0 register must be set according to this rate for properly generating the audio bit clock on the DCLK pin and the word selection clock on the DSEL pin. These clocks are not generated when no data is available at the data converter input.

For DAC compatibility, the bit clock frequency is programmable for outputting 16 bits or 32 bits per channel using the DSIZ bit in AUDCON0 register (see Section "Data Converter", page 17), and the word selection signal is programmable for outputting left channel on low or high level according to POL bit in AUDCON0 register as shown in Figure 15.

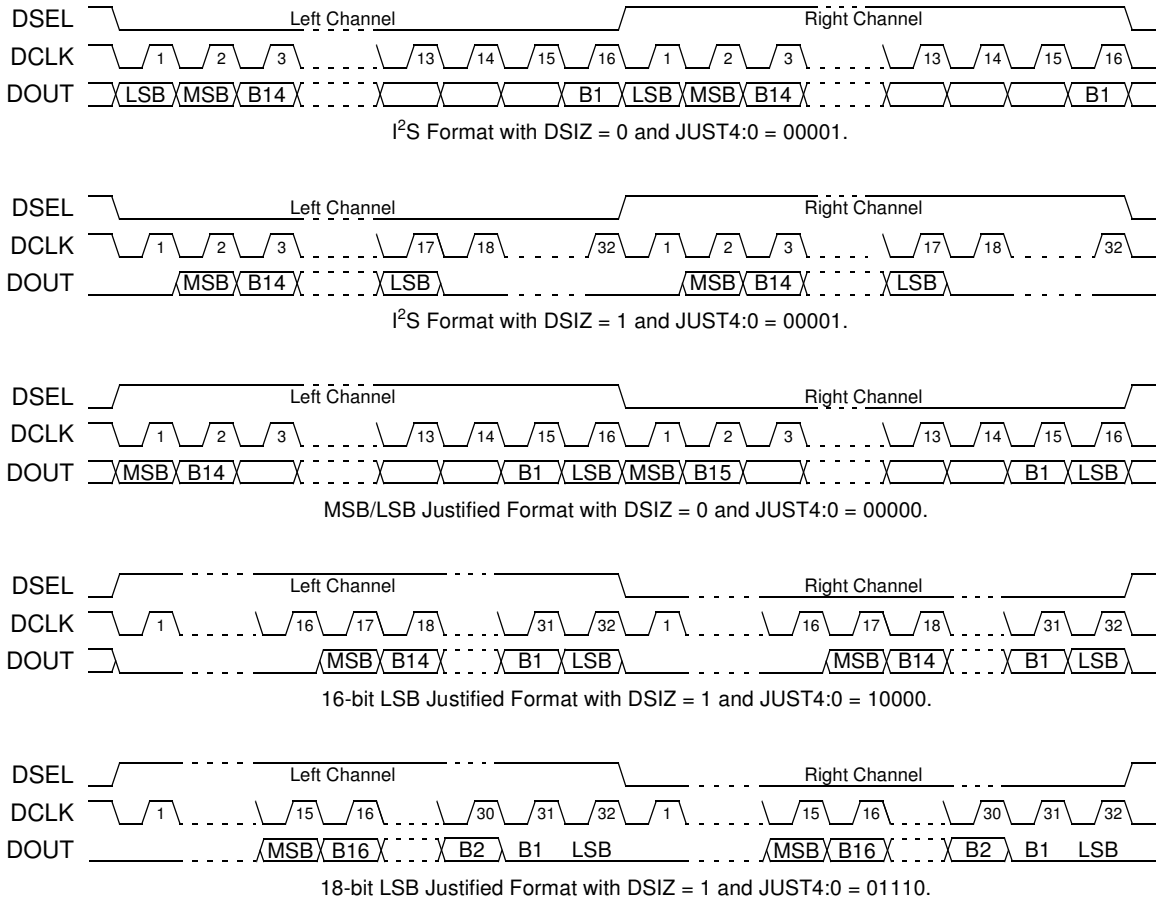
Figure 15. DSEL Output Polarity



Data Converter

The data converter block converts the audio stream input from the 16-bit parallel format to a serial format. For accepting all PCM formats and I^2S format, JUST4:0 bits in AUDCON0 register are used to shift the data output point. As shown in Figure 16, these bits allow MSB justification by setting JUST4:0 = 00000, LSB justification by setting JUST4:0 = 10000, I^2S Justification by setting JUST4:0 = 00001, and more than 16-bit LSB justification by filling the low significant bits with logic 0.

Figure 16. Audio Output Format



The data converter receives its audio stream from 2 sources selected by the SRC bit in AUDCON1 register. When cleared, the audio stream comes from the MP3 decoder (see Section “MP3 Decoder”, page 12) for song playing. When set, the audio stream is coming from the C51 core for voice or sound playing.

As soon as first audio data is input to the data converter, it enables the clock generator for generating the bit and word clocks.

Audio Buffer

In voice or sound playing mode, the audio stream comes from the C51 core through an audio buffer. The data is in 8-bit format and is sampled at 8 kHz. The audio buffer adapts the sample format and rate. The sample format is extended to 16 bits by filling the LSB to 00h. Rate is adapted to the DAC rate by duplicating the data using DUP1:0 bits in AUDCON1 register according to Table 18.

The audio buffer interfaces to the C51 core through three flags: the sample request flag (SREQ in AUDSTA register), the under-run flag (UNDR in AUDSTA register) and the busy flag (AUBUSY in AUDSTA register). SREQ and UNDR can generate an interrupt request as explained in Section "Interrupt Request", page 19. The buffer size is 8 Bytes large. SREQ is set when the samples number switches from 4 to 3 and reset when the samples number switches from 4 to 5; UNDR is set when the buffer becomes empty signaling that the audio interface ran out of samples; and AUBUSY is set when the buffer is full.

Table 18. Sample Duplication Factor

DUP1	DUP0	Factor
0	0	No sample duplication, DAC rate = 8 kHz (C51 rate).
0	1	One sample duplication, DAC rate = 16 kHz (2 x C51 rate).
1	0	2 samples duplication, DAC rate = 32 kHz (4 x C51 rate).
1	1	Three samples duplication, DAC rate = 48 kHz (6 x C51 rate).

MP3 Buffer

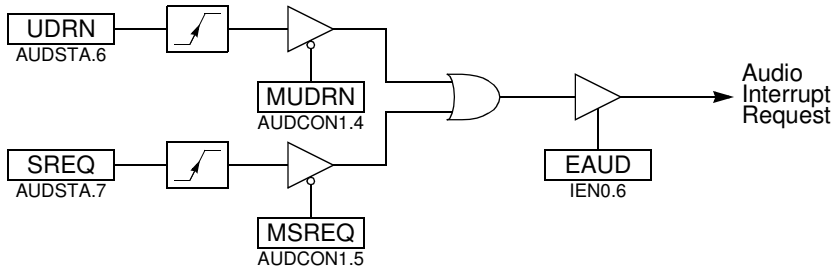
In song playing mode, the audio stream comes from the MP3 decoder through a buffer. The MP3 buffer is used to store the decoded MP3 data and interfaces to the decoder through a 16-bit data input and data request signal. This signal asks for data when the buffer has enough space to receive new data. Data request is conditioned by the DREQEN bit in AUDCON1 register. When set, the buffer requests data to the MP3 decoder. When cleared no more data is requested but data are output until the buffer is empty. This bit can be used to suspend the audio generation (pause mode).

Interrupt Request

The audio interrupt request can be generated by 2 sources when in C51 audio mode: a sample request when SREQ flag in AUDSTA register is set to logic 1, and an under-run condition when UDRN flag in AUDSTA register is set to logic 1. Both sources can be enabled separately by masking one of them using the MSREQ and MUDRN bits in AUDCON1 register. A global enable of the audio interface is provided by setting the EAUD bit in IEN0 register.

The interrupt is requested each time one of the 2 sources is set to one. The source flags are cleared by writing some data in the audio buffer through AUDDAT, but the global audio interrupt flag is cleared by hardware when the interrupt service routine is executed.

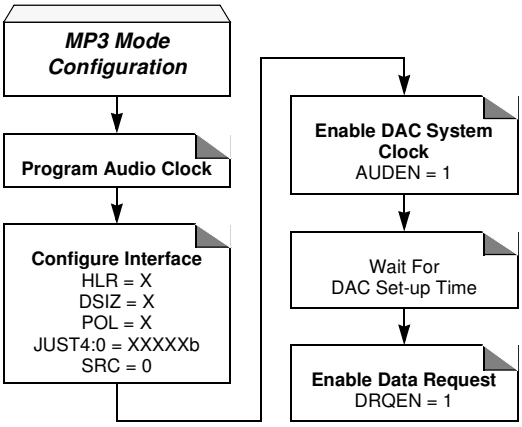
Figure 17. Audio Interface Interrupt System



MP3 Song Playing

In MP3 song playing mode, the operations to do are to configure the PLL and the audio interface according to the DAC selected. The audio clock is programmed to generate the 256·Fs or 384·Fs as explained in Section "Clock Generator", page 17. Figure 18 shows the configuration flow of the audio interface when in MP3 song mode.

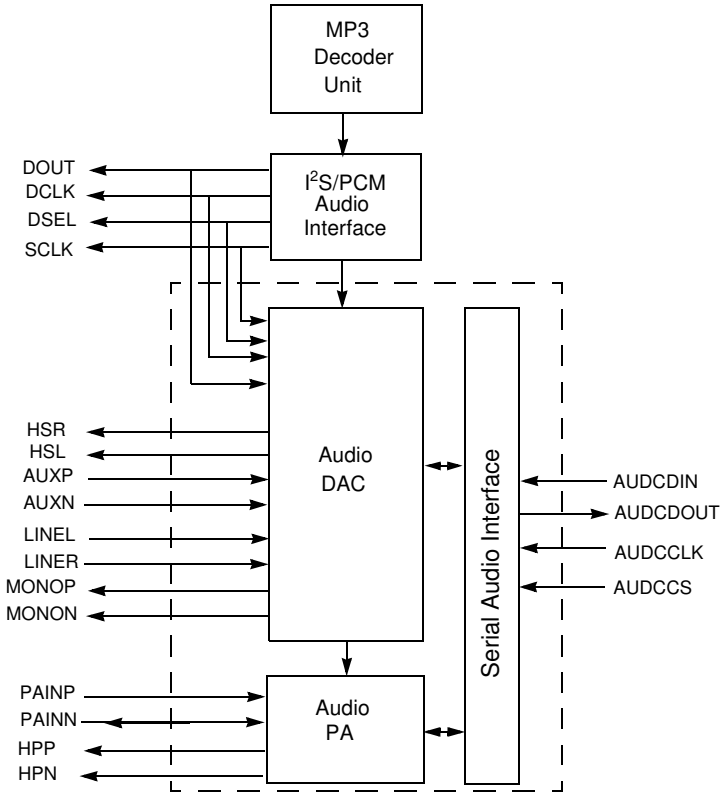
Figure 18. MP3 Mode Audio Configuration Flow



DAC and PA Interface

The AT83SND2CMP3 implements a stereo Audio Digital-to-Analog Converter and Audio Power Amplifier targeted for Li-Ion or Ni-Mh battery powered devices.

Figure 19. Audio Interface Block Diagram



DAC

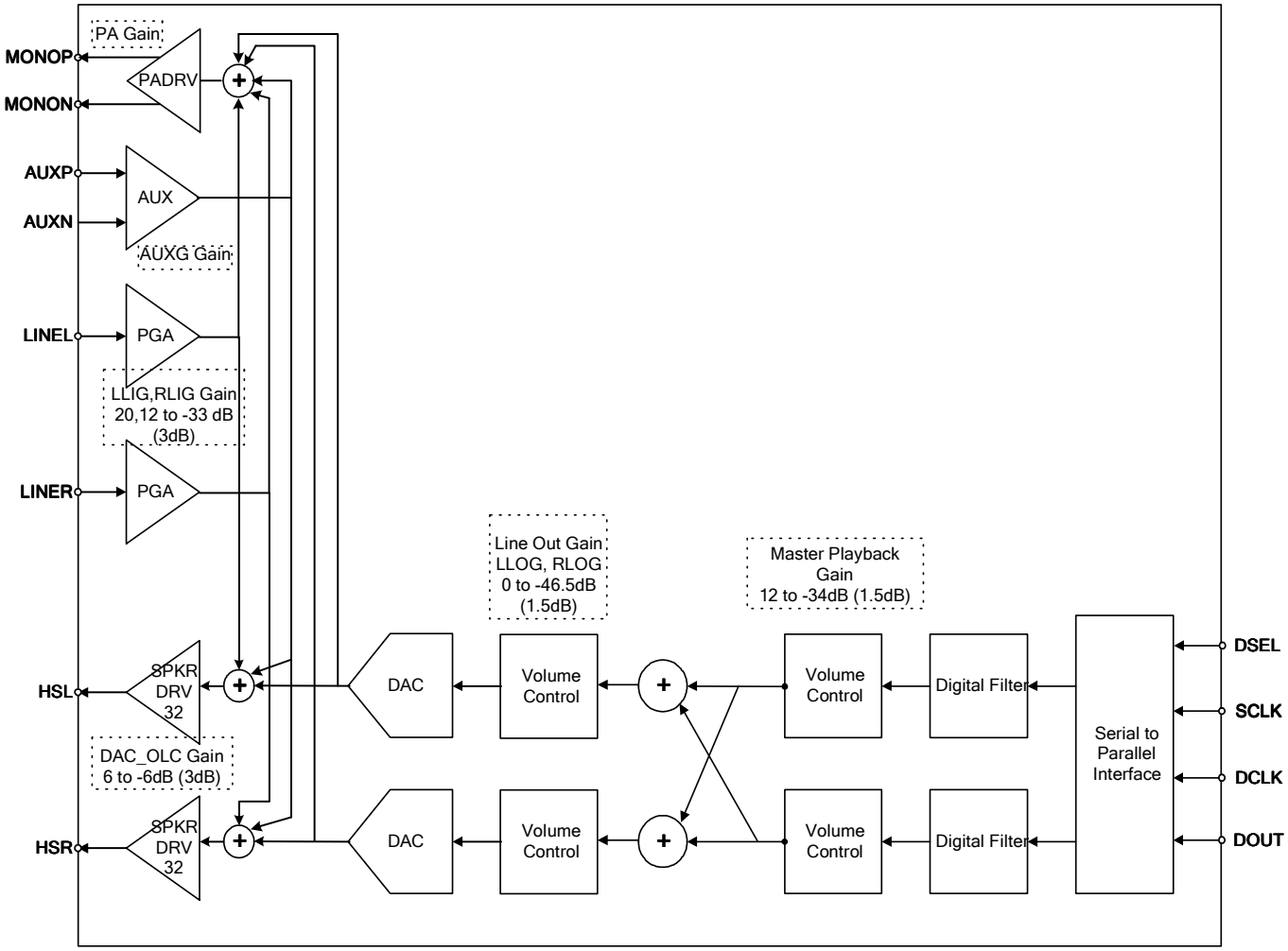
The Stereo DAC section is a complete high performance, stereo, audio digital-to-analog converter delivering 93 dB Dynamic Range. It comprises a multibit sigma-delta modulator with dither, continuous time analog filters and analog output drive circuitry. This architecture provides a high insensitivity to clock jitter. The digital interpolation filter increases the sample rate by a factor of 8 using 3 linear phase half-band filters cascaded, followed by a first order SINC interpolator with a factor of 8. This filter eliminates the images of baseband audio, remaining only the image at 64x the input sample rate, which is eliminated by the analog post filter. Optionally, a dither signal can be added that may reduce eventual noise tones at the output. However, the use of a multibit sigma-delta modulator already provides extremely low noise tones energy.

Master clock is 128 up to 512 times the input data rate allowing choice of input data rate up to 50 kHz, including standard audio rates of 48, 44.1, 32, 16 and 8 kHz. The DAC section is followed by a volume and mute control and can be simultaneously played back directly through a Stereo 32Ω Headset pair of drivers. The Stereo 32Ω Headset pair of drivers also includes a mixer of a LINEL and LINER pair of stereo inputs as well as a differential monaural auxiliary input (line level).

DAC Features

- **20 bit D/A Conversion**
- **72dB Dynamic Range, -75dB THD Stereo line-in or microphone interface with 20dB amplification**
- **93dB Dynamic Range, -80dB THD Stereo D/A conversion**
- **74dB Dynamic Range / -65dB THD for 20mW output power over 32 Ohm loads**
- **Stereo, Mono and Reverse Stereo Mixer**
- **Left/Right speaker short-circuit detection flag**
- **Differential mono auxiliary input amplifier and PA driver**
- **Audio sampling rates (Fs): 16, 22.05, 24, 32, 44.1 and 48 kHz.**

Figure 20. Stereo DAC functional diagram



Digital Signals Timing

Data Interface

To avoid noises at the output, the reset state is maintained until proper synchronism is achieved in the DAC serial interface:

- DSEL
- SCLK
- DCLK
- DOUT

The data interface allows three different data transfer modes:

Figure 21. 20 bit I2S justified mode

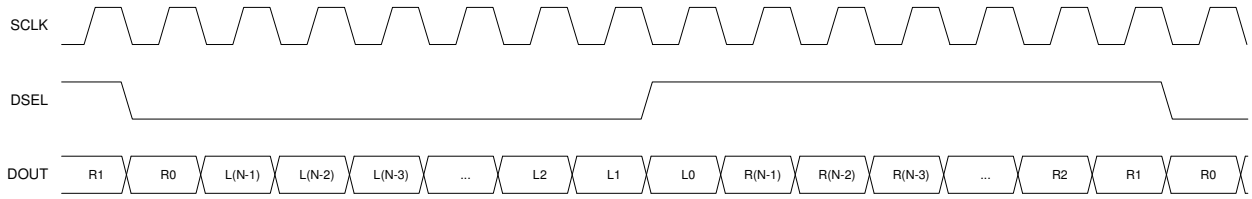


Figure 22. 20 bit MSB justified mode

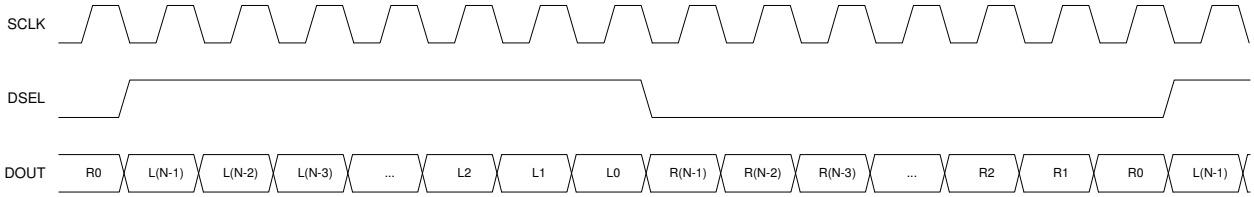
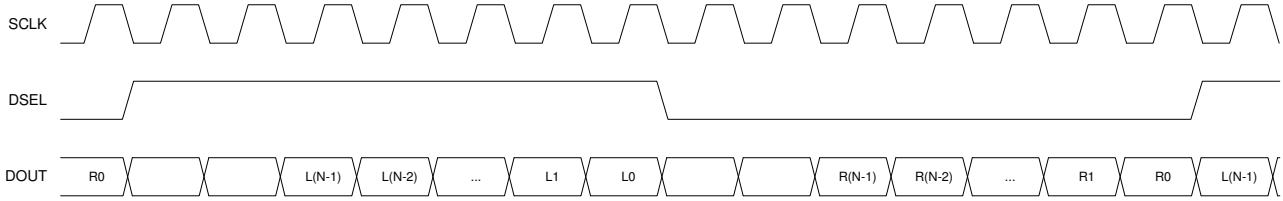


Figure 23. 20 bit LSB justified mode



The selection between modes is done using the DINTSEL 1:0 in DAC_MISC register (Table 40.) according with the following table:

DINTSEL 1:0	Format
00	I2S Justified
01	MSB Justified
1x	LSB Justified

The data interface always works in slave mode. This means that the DSEL and the DCLK signals are provided by microcontroller audio data interface.

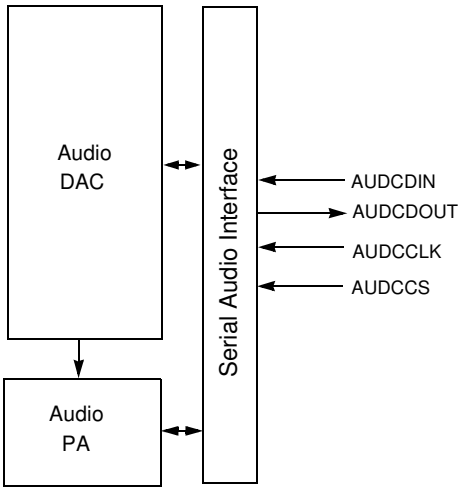
Serial Audio DAC Interface

The serial audio DAC interface is a Synchronous Peripheral Interface (SPI) in slave mode:

- AUDCDIN: is used to transfer data in series from the master to the slave DAC. It is driven by the master.
- AUDCDOUT: is used to transfer data in series from the slave DAC to the master. It is driven by the selected slave DAC.
- Serial Clock (AUDCCLK): it is used to synchronize the data transmission both in and out the devices through the AUDCDIN and AUDCDOUT lines.

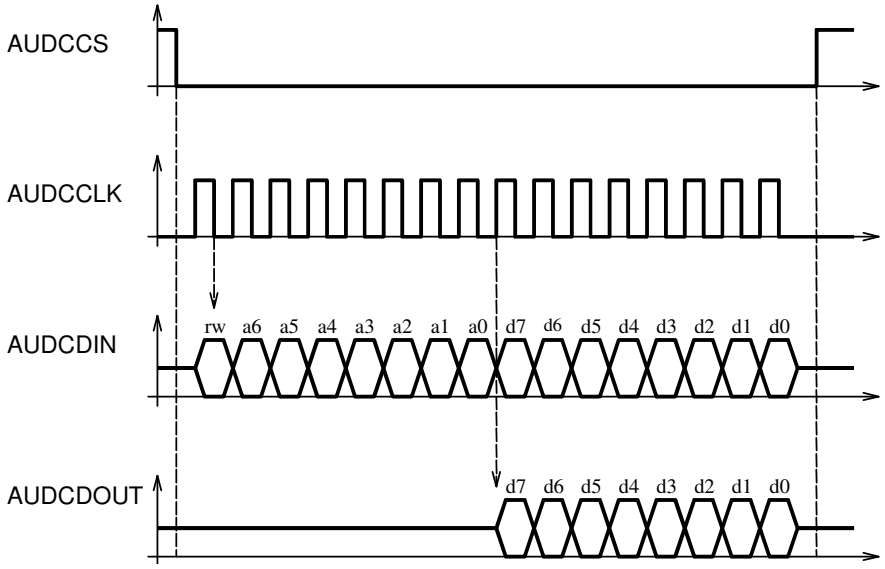
Note: Refer to Table 29. for DAC SPI Interface Description

Figure 24. Serial Audio Interface



Protocol is as following to access DAC registers:

Figure 25. Dac SPI Interface



DAC Interface SPI Protocol

On AUDCDIN, the first bit is a read/write bit. 0 indicates a write operation while 1 is for a read operation. The 7 following bits are used for the register address and the 8 last ones are the write data. For both address and data, the most significant bit is the first one.

In case of a read operation, AUDCDOUT provides the contents of the read register, MSB first.

The transfer is enabled by the AUDCCS signal active low. The interface is resetted at every rising edge of AUDCCS in order to come back to an idle state, even if the transfer does not succeed. The DAC Interface SPI is synchronized with the serial clock AUDC-

CLK. Falling edge latches AUDCDIN input and rising edge shifts AUDCDOUT output bits.

Note that the DLCK must run during any DAC SPI interface access (read or write).

Figure 26. DAC SPI Interface Timings

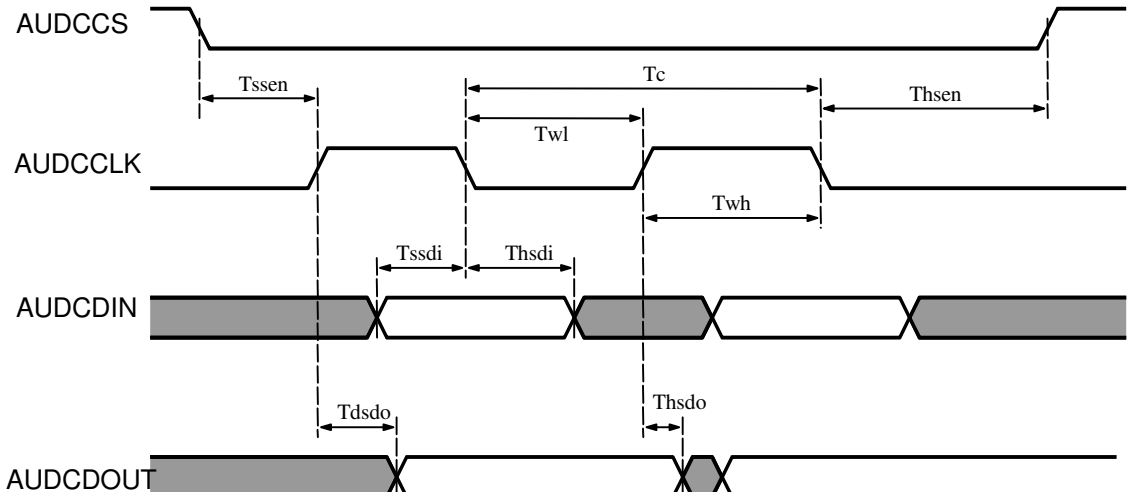


Table 19. Dac SPI Interface Timings

Timing parameter	Description	Min	Max
Tc	AUDCCLK min period	150 ns	-
Twl	AUDCCLK min pulse width low	50 ns	-
Twh	AUDCCLK min pulse width high	50 ns	-
Tssen	Setup time AUDCCS falling to AUDCCLK rising	50 ns	-
Thsen	Hold time AUDCCLK falling to AUDCCS rising	50 ns	-
Tssdi	Setup time AUDCDIN valid to AUDCCLK falling	20 ns	-
Thsdi	Hold time AUDCCLK falling to AUDCDIN not valid	20 ns	-
Tdsdo	Delay time AUDCCLK rising to AUDCDOUT valid	-	20 ns
Thsdo	Hold time AUDCCLK rising to AUDCDOUT not valid	0 ns	-

DAC Register Tables

Table 20. DAC Register Address

Address	Register	Name	Access	Reset state
00h	DAC_CTRL	Dac Control	Read/Write	00h
01h	DAC_LLIG	Dac Left Line in Gain	Read/Write	05h
02h	DAC_RLIG	Dac Right Line in Gain	Read/Write	05h
03h	DAC_LPMG	Dac Left Master Playback Gain	Read/Write	08h
04h	DAC_RPMG	Dac Right Master Playback Gain	Read/Write	08h
05h	DAC_LLOG	Dac Left Line Out Gain	Read/Write	00h
06h	DAC_RLOG	Dac Right Line Out Gain	Read/Write	00h
07h	DAC_OLC	Dac Output Level Control	Read/Write	22h
08h	DAC_MC	Dac Mixer Control	Read/Write	09h
09h	DAC_CSFC	Dac Clock and Sampling Frequency Control	Read/Write	00h
0Ah	DAC_MISC	Dac Miscellaneous	Read/Write	00h
0Ch	DAC_PRECH	Dac Precharge Control	Read/Write	00h
0Dh	DAC_AUXG	Dac Auxilary input gain Control	Read/Write	05h
10h	DAC_RST	Dac Reset	Read/Write	00h
11h	PA_CRTL	Power Amplifier Control	Read/Write	00h

DAC Gain

The DAC implements severals gain control: line-in (Table 21.), master playback (), line-out (Table 24.).

Table 21. Line-in gain

LLIG 4:0 RLIG 4:0	Gain (dB)
00000	20
00001	12
00010	9
00011	6
00100	3
00101	0
00110	-3
00111	-6
01000	-9
01001	-12
01010	-15
01011	-18
01100	-21

Table 21. Line-in gain (Continued)

01101	-24
01110	-27
01111	-30
10000	-33
10001	< -60

Table 22. Master Playback Gain

LMPG 5:0 RMPG 5:0	Gain (dB)
000000	12.0
000001	10.5
000010	9.0
000011	7.5
000100	6.0
000101	4.5
000110	3.0
000111	1.5
001000	0.0
001001	-1.5
001010	-3.0
001011	-4.5
001100	-6.0
001101	-7.5
001110	-9.0
001111	-10.5
010000	-12.0
010001	-13.5
010010	-15.0
010011	-16.5
010100	-18.0
010101	-19.5
010110	-21.0
010111	-22.5
011000	-24.0
011001	-25.5

Table 22. Master Playback Gain (Continued)

LMPG 5:0 RMPG 5:0	Gain (dB)
011010	-27.0
011011	-28.5
011100	-30.0
011101	-31.5
011110	-33.0
011111	-34.5
100000	mute

Table 23. Line-out Gain

LLOG 5:0 RLOG 5:0	Gain (dB)
000000	0.0
000001	-1.5
000010	-3.0
000011	-4.5
000100	-6.0
000101	-7.5
000110	-9.0
000111	-10.5
001000	-12.0
001001	-13.5
001010	-15.0
001011	-16.5
001100	-18.0
001101	-19.5
001110	-21.0
001111	-22.5
010000	-24.0
010001	-25.5
010010	-27.0
010011	-28.5
010100	-30.0
010101	-31.5
010110	-33.0

Table 23. Line-out Gain (Continued)

010111	-34.5
011000	-36.0
011001	-37.5
011010	-39.0
011011	-40.5
011100	-42.0
011101	-43.5
011110	-45.0
011111	-46.5
100000	mute

Table 24. DAC Output Level Control

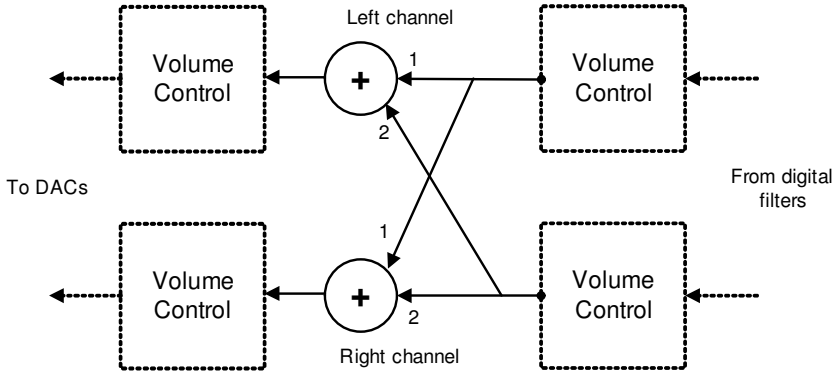
LOLC 2:0 ROLC 2:0	Gain (dB)
000	6
001	3
010	0
011	-3
100	-6

Digital Mixer Control

The Audio DAC features a digital mixer that allows the mixing and selection of multiple input sources.

The mixing / multiplexing functions are described in the following table according with the next figure:

Figure 27. Mixing / Multiplexing functions



Note: Whenever the two mixer inputs are selected, a –6 dB gain is applied to the output signal. Whenever only one input is selected, no gain is applied.

Signal	Description
LMSMIN1	Left Channel Mono/Stereo Mixer Left Mixed input enable – High to enable, Low to disable
LMSMIN2	Left Channel Mono/Stereo Mixer Right Mixed input enable – High to enable, Low to disable
RMSMIN1	Right Channel Mono/Stereo Mixer Left Mixed input enable – High to enable, Low to disable
RMSMIN2	Right Channel Mono/Stereo Mixer Right Mixed input enable – High to enable, Low to disable

Note: Refer to DAC_MC register Table 38. for signal description

Master Clock and Sampling Frequency Selection

The following table describes the different modes available for master clock and sampling frequency selection by setting OVRSEL bit in DAC_CSFC register (refer to Table 39.).

Table 25. Master Clock selection

OVRSEL	Master Clock
0	256 x FS
1	384 x FS

The selection of input sample size is done using the NBITS 1:0 in DAC_MISC register (refer to Table 40.) according to Table 26.

Table 26. Input Sample Size Selection

NBITS 1:0	Format
00	16 bits
01	18 bits
10	20 bits

The selection between modes is done using DINTSEL 1:0 in DAC_MISC register (refer to Table 40.) according to Table 27.

Table 27. Format Selection

DINTSEL 1:0	Format
00	I2S Justified
01	MSB Justified
1x	LSB Justified

De-emphasis and dither enable

The circuit features a de-emphasis filter for the playback channel. To enable the de-emphasis filtering, DEEMPEN must be set to high.

Likewise, the dither option (added in the playback channel) is enabled by setting the DITHEN signal to High.

Table 28. DAC Auxlilary Input Gain

AUXG 4:0	Gain (dB)
00000	20
00001	12
00010	9
00011	6
00100	3
00101	0
00110	-3
00111	-6
01000	-9
01001	-12
01010	-15
01011	-18
01100	-21
01101	-24
01110	-27
01111	-30
10000	-33
10001	<-60

Register

Table 29. AUXCON Register

AUXCON (S:90h) – Auxiliary Control Register

7	6	5	4	3	2	1	0
SDA	**SCL**	**-**	**AUDCDOUT**	**AUDCDIN**	**AUDCCLK**	**AUDCCS**	**KIN0**

Bit Number	Bit Mnemonic	Description
7	SDA	**TWI Serial Data** SDA is the bidirectional Two Wire data line.
6	SCL	**TWI Serial Clock** When TWI controller is in master mode, SCL outputs the serial clock to the slave peripherals. When TWI controller is in slave mode, SCL receives clock from the master controller.
5	-	Not used.
4	AUDCDOUT	**Audio Dac SPI Data Output**.
3	AUDCDIN	**Audio Dac SPI Data Input**
2	AUDCCLK	**Audio Dac SPI clock**
1	AUDCCS	**Audio Dac Chip select** Set to deselect DAC Clear to select DAC
0	KIN0	**Keyboard Input Interrupt**.

Reset Value = 1111 1111b

Table 30. Dac Control Register Register - DAC_CTRL (00h)

7	6	5	4	3	2	1	0
ONPADRV	ONAUXIN	ONDACR	ONDACL	ONLNOR	ONLNOL	ONLNIR	ONLNIL

Bit Number	Bit Mnemonic	Description
7	ONPADRV	Differential mono PA driver Clear to power down. Set to power up.
6	ONAUXIN	Differential mono auxiliary input amplifier Clear to power down. Set to power up.
5	ONDACR	Right channel DAC Clear to power down. Set to power up.
4	ONDACL	Left channel DAC Clear to power down. Set to power up.
3	ONLNOR	Right channel line out driver Clear to power down. Set to power up.
2	ONLNOL	Left channel line out driver Clear to power down. Set to power up.
1	ONLNIR	Right channel line in amplifier Clear to power down. Set to power up.
0	ONLNIL	Left channel line in amplifier Clear to power down. Set to power up.

Reset Value = 00000000b

Table 31. DAC Left Line In Gain Register - DAC_LLIG (01h)

7	6	5	4	3	2	1	0
-	-	-	LLIG4	LLIG3	LLIG2	LLIG1	LLIG0

Bit Number	Bit Mnemonic	Description
7:5	-	Not used
4:0	LLIG 4:0	Left channel line in analog gain selector

Reset Value = 00000101b

Table 32. DAC Right Line In Gain Register - DAC_RLIG (02h)

7	6	5	4	3	2	1	0
-	-	-	RLIG4	RLIG3	RLIG2	RLIG1	RLIG0

Bit Number	Bit Mnemonic	Description
7:5	-	Not used
4:0	RLIG 4:0	Right channel line in analog gain selector

Reset Value = 0000101b

Table 33. DAC Left Master Playback Gain Register - DAC_LMPG (03h)

7	6	5	4	3	2	1	0
-	-	LMPG5	LMPG4	LMPG3	LMPG2	LMPG1	LMPG0

Bit Number	Bit Mnemonic	Description
7:6	-	Not used
5:0	LMPG 5:0	Left channel master playback digital gain selector

Reset Value = 00001000b

Table 34. DAC Right Master Playback Gain Register - DAC_RMPG (04h)

7	6	5	4	3	2	1	0
-	-	RMPG5	RMPG4	RMPG3	RMPG2	RMPG1	RMPG0

Bit Number	Bit Mnemonic	Description
7:6	-	Not used
5:0	RMPG 5:0	Right channel master playback digital gain selector

Reset Value = 00001000b

Table 35. DAC Left Line Out Gain Register - DAC_LLOG (05h)

7	6	5	4	3	2	1	0
-	-	LLOG5	LLOG4	LLOG3	LLOG2	LLOG1	LLOG0

Bit Number	Bit Mnemonic	Description
7:6	-	Not used
5:0	LLOG 5:0	Left channel line out digital gain selector

Reset Value = 00000000b

Table 36. DAC Rigth Line Out Gain Register - DAC_RLOG (06h)

7	6	5	4	3	2	1	0
-	-	RLOG5	RLOG4	RLOG3	RLOG2	RLOG1	RLOG0

Bit Number	Bit Mnemonic	Description
7:6	-	Not used
5:0	RLOG 5:0	Right channel line out digital gain selector

Reset Value = 00000000b

Table 37. DAC Output Level Control Register - DAC_OLC (07h)

7	6	5	4	3	2	1	0
RSHORT	ROLC2	RLOC1	RLOC0	LSHORT	LOLC2	LOLC1	LOLC0

Bit Number	Bit Mnemonic	Description
7	RSHORT	Right channel short circuit indicator (persistent; after being set, bit is not cleared automatically even after the short circuit is eliminated; must be cleared by reset cycle or direct register write operation)
6:4	ROLC 2:0	Right channel output level control selector
3	LSHORT	Left channel short circuit indicator (persistent; after being set, bit is not cleared automatically even after the short circuit is eliminated; must be cleared by reset cycle or direct register write operation)
2:0	LOLC 2:0	Left channel output level control selector

Reset Value = 00100010b

Table 38. Dac Mixer Control Register - DAC_MC (08h)

7	6	5	4	3	2	1	0
-	-	INVR	INVL	RMSMIN2	RMSMIN1	LMSMIN2	LMSMIN1

Bit Number	Bit Mnemonic	Description
7:6	-	Not used
5	INVR	Right channel mixer output invert Set to enable. Clear to disable.
4	INVL	Left channel mixer output invert. Set to enable. Clear to disable.
3	RMSMIN2	Right Channel Mono/Stereo Mixer Right Mixed input enable Set to enable. Clear to disable.
2	RMSMIN1	Right Channel Mono/Stereo Mixer Left Mixed input enable Set to enable. Clear to disable.
1	LMSMIN2	Left Channel Mono/Stereo Mixer Right Mixed input enable Set to enable. Clear to disable.
0	LMSMIN1	Left Channel Mono/Stereo Mixer Left Mixed input enable Set to enable. Clear to disable.

Reset Value = 00001001b

Table 39. DAC Mixer Control Register - DAC_CSFC (09h)

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

Bit Number	Bit Mnemonic	Description
7:5	-	Not used
4	OVRSEL	Master clock selector Clear for 256 x Fs. Set for 384 x Fs.
3:0	-	Not Used

Reset Value = 00000000b

Table 40. Dac Miscellaneous Register - DAC_ MISC (0Ah)

7	6	5	4	3	2	1	0
-	-	DINTSEL1	DINTSEL0	DITHEN	DEEMPEN	NBITS1	NBITS0

Bit Number	Bit Mnemonic	Description
7	-	Not used
6	-	Not used
5:4	DINTSEL1:0	I2S data format selector
3	DITHEN	Dither enable (Clear this bit to disable, set to enable)
2	DEEMPEN	De-emphasis enable (clear this bit to disable, set to enable)
1:0	NBITS 1:0	Data interface word length

Reset Value = 00000010b

Table 41. DAC Precharge Control Register - DAC_ PRECH (0Ch)

7	6	5	4	3	2	1	0
PRCHARGEPADRV	- PRCHARGEAUXIN	- PRCHARGELNOR	PRCHARGELNOL	PRCHARGELNIL	PRCHARGELNIL	PRCHARGE	ONMSTR

Bit Number	Bit Mnemonic	Description
7	PRCHARGEPADRV	Differential mono PA driver pre-charge. Set to charge.
6	PRCHARGEAUXIN	Differential mono auxiliary input pre-charge. Set to charge.
5	PRCHARGELNOR	Right channel line out pre-charge. Set to charge.
4	PRCHARGELNOL	Left channel line out pre-charge. Set to charge.
3	PRCHARGELNIR	Right channel line in pre-charge. Set to charge.
2	PRCHARGELNIL	Left channel line in pre-charge Set to charge.
1	PRCHARGE	Master pre-charge Set to charge.
0	ONMSTR	Master power on control Clear to power down. Set to to power up.

Reset Value = 00000000b

Table 42. DAC Auxilary input gain Register - DAC_ AUXG (0Dh)l

7	6	5	4	3	2	1	0
-	-	-	AUXG4	AUXG3	AUXG2	AUXG1	AUXG0

Bit Number	Bit Mnemonic	Description
7:5	-	Not used
4:0	AUXG 4:0	Differential mono auxiliary input analog gain selector

Reset Value = 0000101b

DAC Reset Register - DAC_ RST (10h)

7	6	5	4	3	2	1	0
-	-	-	-	-	RESMASK	RESFILZ	RSTZ

Bit Number	Bit Mnemonic	Description
7:3	-	Not Used.
2	RESMASK	Active high reset mask of the audio codec
1	RESFILZ	Active low reset of the audio codec filter
0	RSTZ	Active low reset of the audio codec

Reset Value = 00000000b

Note: Refer to Audio DAC Startup sequence.

Power Amplifier

High quality mono output is provided. The DAC output is connected through a buffer stage to the input of the Audio Power Amplifier, using two coupling capacitors The mono buffer stage also includes a mixer of the LINEL and LINER inputs as well as a differential monaural auxiliary input (line level) which can be, for example, the output of a voice CODEC output driver in mobile phones.

In the full power mode, the Power Amplifier is capable of driving an 8Ω Loudspeaker at maximum power of 440mW, making it suitable as a handsfree speaker driver in Wireless Handset Application.

The Low Power Mode is designed to be switched from the handsfree mode to the normal earphone/speaker mode of a telephone handset.

The audio power amplifier is not internally protected against short-circuit. The user should avoid any short-circuit on the load.

PA Features

- **0.44W on 8Ω Load**
- **Low Power Mode for Earphone**
- **Programmable Gain (-22 to +20 dB)**
- **Fully Differential Structure, Input and Output**

Table 43. PA Gain

APAGAIN 3:0	Gain (db)
0000	-22
0001	20
0010	17
0011	14
0100	11
0101	8
0110	5
0111	2
1000	-1
1001	-4
1010	-7
1011	-10
1100	-13
1101	-16
1110	-19
1111	-22

Table 44. PA Operating Mode

APAON	APAPRECH	Operating Mode
0	0	Stand-By
0	1	Input Capacitors Precharge
1	0	Active Mode
1	1	Forbidden State

Table 45. PA Low Power Mode

APALP	Power Mode
0	Low power mode
1	High power mode

Audio Supplies and Start-up

In operating mode AUDVBAT (supply of the audio power amplifier) must be between 3V and 5,5V.

AUDVDD, HSVDD and VDD must be inferior or equal to AUDVBAT.

A typical application is AUDVBAT connected to a battery and AUDVDD, HSVDD and VDD supplied by regulators.

AUDVBAT must be present at the same time or before AUDVDD, HSVDD and VDD.

AUDRST must be active low (0) until the voltages are not etablished and reach the proper values.

To avoid noise issues, it is recommended to use ceramic decoupling capacitors for each supply closed to the package. The track of the supplies must be optimized to minimize the resistance especially on AUDVBAT where all the current from the power amplifier comes from.

Note: Refer to the application diagram.

Audio DAC Start-up Sequence

In order to minimize any audio output noise during the start-up, the following sequence should be applied.

Example of power-on: Path DAC to Headset Output

- Desassert the Reset: write 07h at address 10h.
- All precharge and Master on: write FFh at address 0Ch.
- Line Out On: write 30h at address 00h.
- Delay 500 ms.
- Precharge off: write 0Ch at address 01h.
- Delay 1 ms.
- Line Out on, DAC On: write 3Ch at address 00h.

Example of power-off: Path DAC to Headset Output

- DAC off: write 30h at address 00h.
- Master off: write 00h at address 0Ch.
- Delay 1 ms.
- All off: write 00h at address 00h

Example Start I2S

- Start DCLK.
- RSTMASK=1.
- RESFILZ=0 and RSTZ=0.
- RESFILZ=1 and RSTZ=1.
- RSTMASK=0.
- Delay 5 ms.
- ONDACL=1 and ONDACR=1.
- Program all DAC settings: audio format, gains...

Example Stop I2S:

- DAC off: ONDACL=0 and ONDACR=0.
- Stop I2S and DLCK.

Audio PA Sequence

PA Power-On Sequence

To avoid an audible ‘click’ at start-up, the input capacitors have to be pre-charged before the Power Amplifier.

PA Power-Off Sequence

To avoid an audible ‘click’ at power-off, the gain should be set to the minimum gain (-22dB) before setting the Power Amplifier.

Precharge Control

The power up of the circuit can be performed independently for several blocks. The sequence flow starts by setting to High the block specific fastcharge control bit and subsequently the associated power control bit. Once the power control bit is set to High, the fast charging starts. This action begins a user controlled fastcharge cycle. When the fastcharge period is over, the user must reset the associated fastcharge bit and the block is ready for use. If a power control bit is cleared a new power up sequence is needed.

The several blocks with independent power control are identified in Table 46. The table describes the power on control and fastcharge bits for each block.

Table 46. Precharge and Power Control

Powered up block	Power on control bit	Precharge Control Bit
Vref & Vcm generator	ONMSTR	PRCHARGE (reg 12; bit 1)
Left line in amplifier	ONLNIL	PRCHARGELNIL
Right line in amplifier	ONLNIR	PRCHARGELNIR
Left line out amplifier	ONLNOL	PRCHARGELNOL
Right line out amplifier	ONLNOR	PRCHARGELNOR
Left D-to-A converter	ONDACL	Not needed
Right D-to-A converter	ONDACR	Not needed
Auxiliary input amplifier	ONAUXIN	PRCHARGEAUXIN
PA Driver output	ONPADRV	PRCHARGEPADRV

Note: Note that all block can be precharged simultaneously.

Register

Table 47. PA Control Register - PA_CTRL (11h)l

7	6	5	4	3	2	1	0
-	APAON	APAPRECH	APALP	APAGAIN3	APAGAIN2	APAGAIN1	APAGAIN0

Bit Number	Bit Mnemonic	Description
7	-	Not used
6	APAON	Audio power amplifier on bit
5	APAPRECH	Audio power amplifier precharge bit
4	APALP	Audio power amplifier low power bit
3:0	APAGAIN3:0	Audio power amplifier gain

Reset Value = 00000000b

Universal Serial Bus

The product implements a USB device controller supporting full speed data transfer. In addition to the default control endpoint 0, it provides 2 other endpoints, which can be configured in control, bulk, interrupt or isochronous modes:

- Endpoint 0: 32-Byte FIFO, default control endpoint
- Endpoint 1, 2: 64-Byte Ping-pong FIFO,

This allows the firmware to be developed conforming to most USB device classes, for example:

- USB Mass Storage Class Bulk-only Transport, Revision 1.0 - September 31, 1999
- USB Human Interface Device Class, Version 1.1 - April 7, 1999
- USB Device Firmware Upgrade Class, Revision 1.0 - May 13, 1999

USB Mass Storage Class Bulk-Only Transport

Within the Bulk-only framework, the Control endpoint is only used to transport class-specific and standard USB requests for device set-up and configuration. One Bulk-out endpoint is used to transport commands and data from the host to the device. One Bulk in endpoint is used to transport status and data from the device to the host.

The following AT83SND2CMP3 configuration adheres to those requirements:

- Endpoint 0: 32 Bytes, Control In-Out
- Endpoint 1: 64 Bytes, Bulk-in
- Endpoint 2: 64 Bytes, Bulk-out

USB Device Firmware Upgrade (DFU)

The USB Device Firmware Update (DFU) protocol can be used to upgrade the on-chip Flash memory of the AT83SND2CMP3. This allows installing product enhancements and patches to devices that are already in the field. 2 different configurations and descriptor sets are used to support DFU functions. The Run-Time configuration co-exist with the usual functions of the device, which is USB Mass Storage for AT83SND2CMP3. It is used to initiate DFU from the normal operating mode. The DFU configuration is used to perform the firmware update after device re-configuration and USB reset. It excludes any other function. Only the default control pipe (endpoint 0) is used to support DFU services in both configurations.

The only possible value for the MaxPacketSize in the DFU configuration is 32 Bytes, which is the size of the FIFO implemented for endpoint 0.

Description

The USB device controller provides the hardware that the AT83SND2CMP3 needs to interface a USB link to a data flow stored in a double port memory.

It requires a 48 MHz reference clock provided by the clock controller as detailed in Section "", page 44. This clock is used to generate a 12 MHz Full Speed bit clock from the received USB differential data flow and to transmit data according to full speed USB device tolerance. Clock recovery is done by a Digital Phase Locked Loop (DPLL) block.

The Serial Interface Engine (SIE) block performs NRZI encoding and decoding, bit stuffing, CRC generation and checking, and the serial-parallel data conversion.

The Universal Function Interface (UFI) controls the interface between the data flow and the Dual Port RAM, but also the interface with the C51 core itself.

Figure 30 shows how to connect the AT83SND2CMP3 to the USB connector. D+ and D- pins are connected through 2 termination resistors. Value of these resistors is detailed in the section “DC Characteristics”.

Figure 28. USB Device Controller Block Diagram

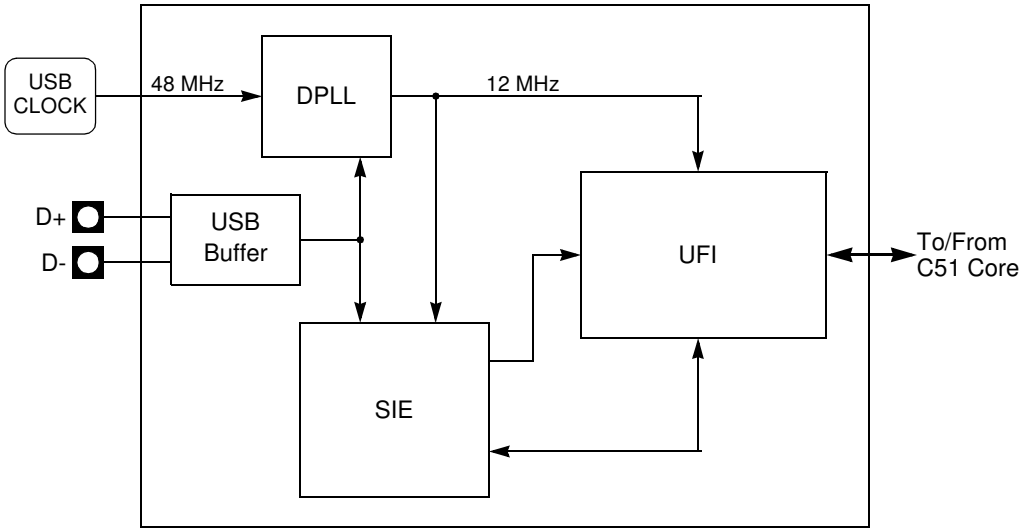
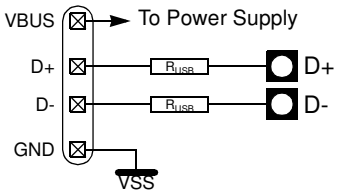


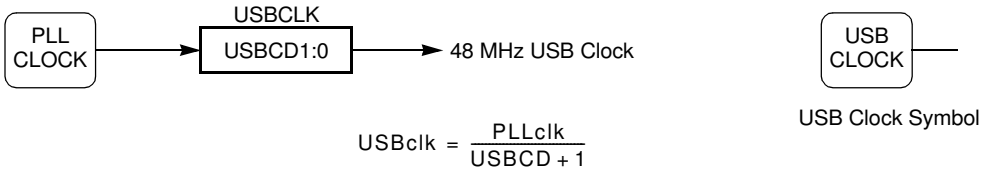
Figure 29. USB Connection



Clock Controller

The USB controller clock is generated by division of the PLL clock. The division factor is given by USBCD1:0 bits in USBCLK register. Figure 30 shows the USB controller clock generator and its calculation formula. The USB controller clock frequency must always be 48 MHz.

Figure 30. USB Clock Generator and Symbol



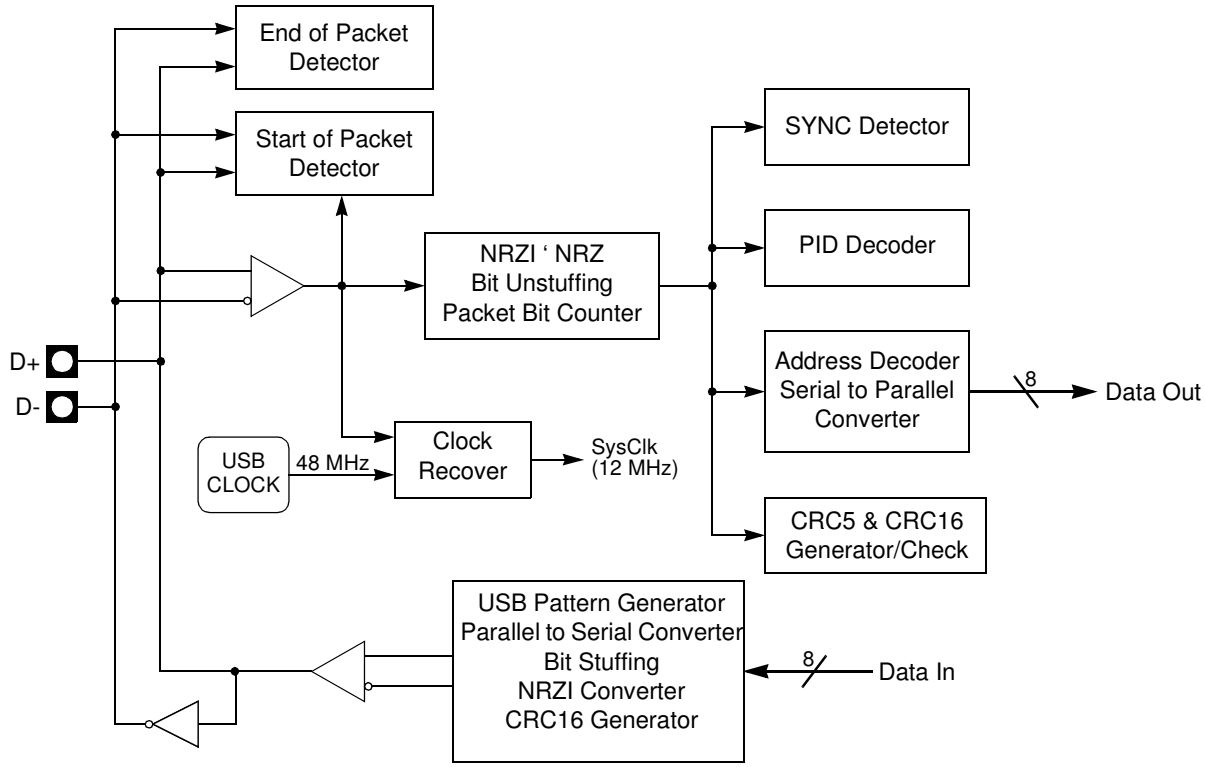
$$USBclk = \frac{PLLclk}{USBCD + 1}$$

Serial Interface Engine (SIE)

The SIE performs the following functions:

- NRZI data encoding and decoding.
- Bit stuffing and unstuffing.
- CRC generation and checking.
- ACKs and NACKs automatic generation.
- TOKEN type identifying.
- Address checking.
- Clock recovery (using DPLL).

Figure 31. SIE Block Diagram



Function Interface Unit (UFI)

The Function Interface Unit provides the interface between the AT83SND2CMP3 and the SIE. It manages transactions at the packet level with minimal intervention from the device firmware, which reads and writes the endpoint FIFOs.

Figure 33 shows typical USB IN and OUT transactions reporting the split in the hardware (UFI) and software (C51) load.

Figure 32. UFI Block Diagram

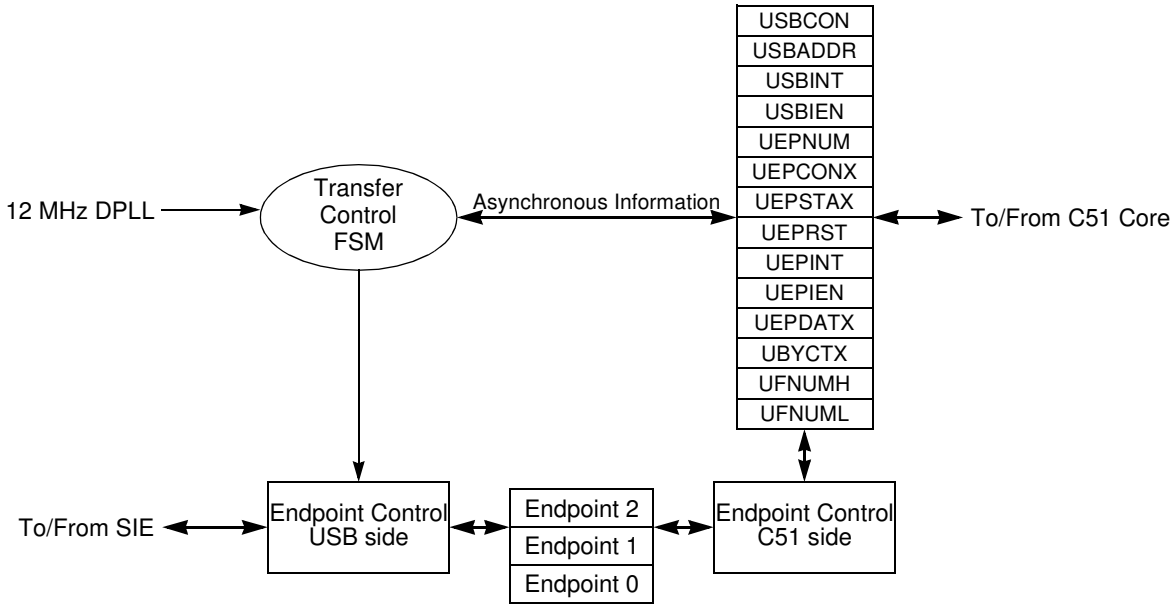
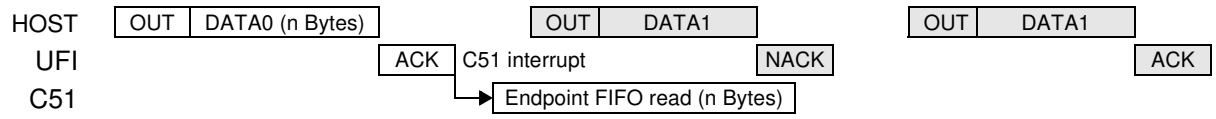


Figure 33. USB Typical Transaction Load

OUT Transactions:



IN Transactions:

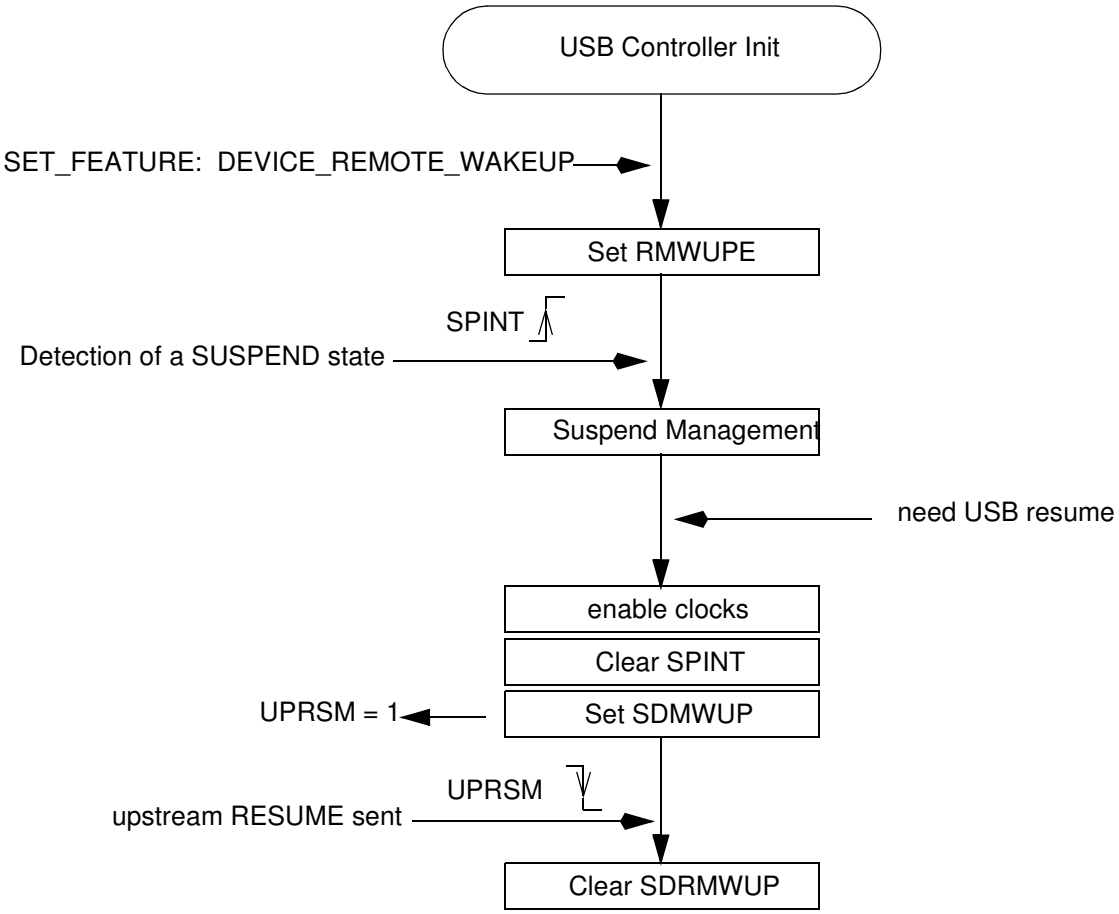
Upstream Resume

A USB device can be allowed by the Host to send an upstream resume for Remote Wake-up purpose.

When the USB controller receives the SET_FEATURE request: DEVICE_REMOTE_WAKEUP, the firmware should set to 1 the RMWUPE bit in the USBCON register to enable this functionality. RMWUPE value should be 0 in the other cases.

If the device is in SUSPEND mode, the USB controller can send an upstream resume by clearing first the SPINT bit in the USBINT register and by setting then to 1 the SDRMWUP bit in the USBCON register. The USB controller sets to 1 the UPRSM bit in the USBCON register. All clocks must be enabled first. The Remote Wake is sent only if the USB bus was in Suspend state for at least 5ms. When the upstream resume is completed, the UPRSM bit is reset to 0 by hardware. The firmware should then clear the SDRMWUP bit.

Figure 34. Example of REMOTE WAKEUP Management



USB Interrupt System

Interrupt System Priorities

Figure 35. USB Interrupt Control System

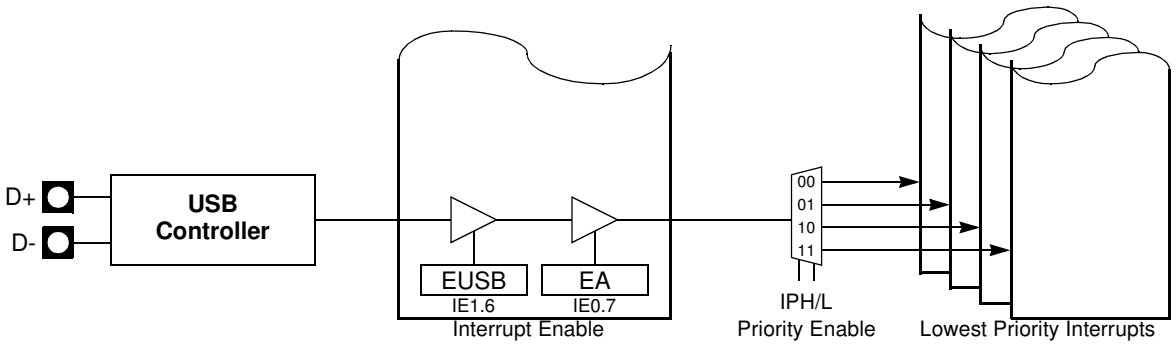


Table 1. Priority Levels

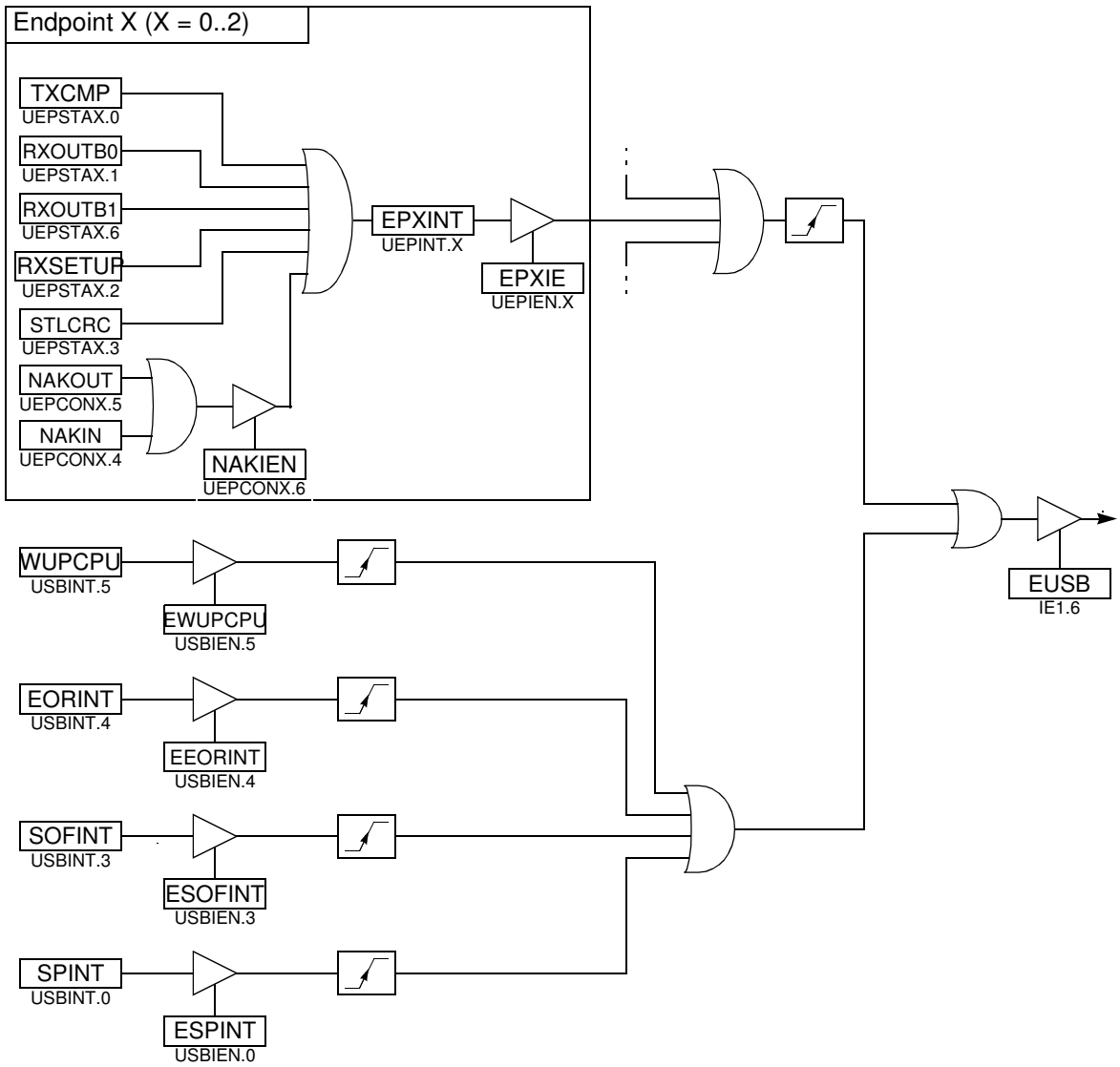
IPHUSB	IPLUSB	USB Priority Level
0	0	0.....Lowest
0	1	1
1	0	2
1	1	3.....Highest

USB Interrupt Control System

As shown in Figure 36, many events can produce a USB interrupt:

- TXCMPL: Transmitted In Data. This bit is set by hardware when the Host accept a In packet.
- RXOUTB0: Received Out Data Bank 0. This bit is set by hardware when an Out packet is accepted by the endpoint and stored in bank 0.
- RXOUTB1: Received Out Data Bank 1 (only for Ping-pong endpoints). This bit is set by hardware when an Out packet is accepted by the endpoint and stored in bank 1.
- RXSETUP: Received Setup. This bit is set by hardware when an SETUP packet is accepted by the endpoint.
- STLCRC: STALLED (only for Control, Bulk and Interrupt endpoints). This bit is set by hardware when a STALL handshake has been sent as requested by STALLRQ, and is reset by hardware when a SETUP packet is received.
- SOFINT: Start of Frame Interrupt . This bit is set by hardware when a USB start of frame packet has been received.
- WUPCPU: Wake-Up CPU Interrupt. This bit is set by hardware when a USB resume is detected on the USB bus, after a SUSPEND state.
- SPINT: Suspend Interrupt. This bit is set by hardware when a USB suspend is detected on the USB bus.

Figure 36. USB Interrupt Control Block Diagram



MultiMedia Card Controller

The AT83SND2CMP3 implements a MultiMedia Card (MMC) controller. The MMC is used to store MP3 encoded audio files in removable Flash memory cards that can be easily plugged or removed from the application.

Card Concept

The basic MultiMedia Card concept is based on transferring data via a minimum number of signals.

Card Signals

The communication signals are:

- CLK: with each cycle of this signal a one bit transfer on the command and data lines is done. The frequency may vary from zero to the maximum clock frequency.
- CMD: is a bi-directional command channel used for card initialization and data transfer commands. The CMD signal has 2 operation modes: open-drain for initialization mode and push-pull for fast command transfer. Commands are sent from the MultiMedia Card bus master to the card and responses from the cards to the host.
- DAT: is a bi-directional data channel. The DAT signal operates in push-pull mode. Only one card or the host is driving this signal at a time.

Card Registers

Within the card interface five registers are defined: OCR, CID, CSD, RCA and DSR. These can be accessed only by the corresponding commands.

The 32-bit Operation Conditions Register (OCR) stores the V_{DD} voltage profile of the card. The register is optional and can be read only.

The 128-bit wide CID register carries the card identification information (Card ID) used during the card identification procedure.

The 128-bit wide Card-Specific Data register (CSD) provides information on how to access the card contents. The CSD defines the data format, error correction type, maximum data access time, data transfer speed, and whether the DSR register can be used. The 16-bit Relative Card Address register (RCA) carries the card address assigned by the host during the card identification. This address is used for the addressed host-card communication after the card identification procedure.

The 16-bit Driver Stage Register (DSR) can be optionally used to improve the bus performance for extended operating conditions (depending on parameters like bus length, transfer rate or number of cards).

Bus Concept

The MultiMedia Card bus is designed to connect either solid-state mass-storage memory or I/O-devices in a card format to multimedia applications. The bus implementation allows the coverage of application fields from low-cost systems to systems with a fast data transfer rate. It is a single master bus with a variable number of slaves. The MultiMedia Card bus master is the bus controller and each slave is either a single mass storage card (with possibly different technologies such as ROM, OTP, Flash etc.) or an I/O-card with its own controlling unit (on card) to perform the data transfer.

The MultiMedia Card bus also includes power connections to supply the cards.

The bus communication uses a special protocol (MultiMedia Card bus protocol) which is applicable for all devices. Therefore, the payload data transfer between the host and the cards can be bi-directional.

Bus Lines

The MultiMedia Card bus architecture requires all cards to be connected to the same set of lines. No card has an individual connection to the host or other devices, which reduces the connection costs of the MultiMedia Card system.

The bus lines can be divided into three groups:

- Power supply: V_{SS1} and V_{SS2}, V_{DD} – used to supply the cards.
- Data transfer: MCMD, MDAT – used for bi-directional communication.
- Clock: MCLK – used to synchronize data transfer across the bus.

Bus Protocol

After a power-on reset, the host must initialize the cards by a special message-based MultiMedia Card bus protocol. Each message is represented by one of the following tokens:

- Command: a command is a token which starts an operation. A command is transferred serially from the host to the card on the MCMD line.
- Response: a response is a token which is sent from an addressed card (or all connected cards) to the host as an answer to a previously received command. It is transferred serially on the MCMD line.
- Data: data can be transferred from the card to the host or vice-versa. Data is transferred serially on the MDAT line.

Card addressing is implemented using a session address assigned during the initialization phase, by the bus controller to all currently connected cards. Individual cards are identified by their CID number. This method requires that every card will have an unique CID number. To ensure uniqueness of CIDs the CID register contains 24 bits (MID and OID fields) which are defined by the MMCA. Every card manufacturers is required to apply for an unique MID (and optionally OID) number.

MultiMedia Card bus data transfers are composed of these tokens. One data transfer is a bus operation. There are different types of operations. Addressed operations always contain a command and a response token. In addition, some operations have a data token, the others transfer their information directly within the command or response structure. In this case no data token is present in an operation. The bits on the MDAT and the MCMD lines are transferred synchronous to the host clock.

2 types of data transfer commands are defined:

- Sequential commands: These commands initiate a continuous data stream, they are terminated only when a stop command follows on the MCMD line. This mode reduces the command overhead to an absolute minimum.
- Block-oriented commands: These commands send a data block succeeded by CRC bits. Both read and write operations allow either single or multiple block transmission. A multiple block transmission is terminated when a stop command follows on the MCMD line similarly to the stream read.

Figure 37 through Figure 41 show the different types of operations, on these figures, grayed tokens are from host to card(s) while white tokens are from card(s) to host.

Figure 37. Sequential Read Operation

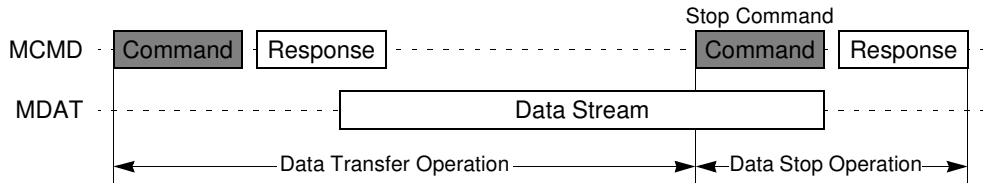
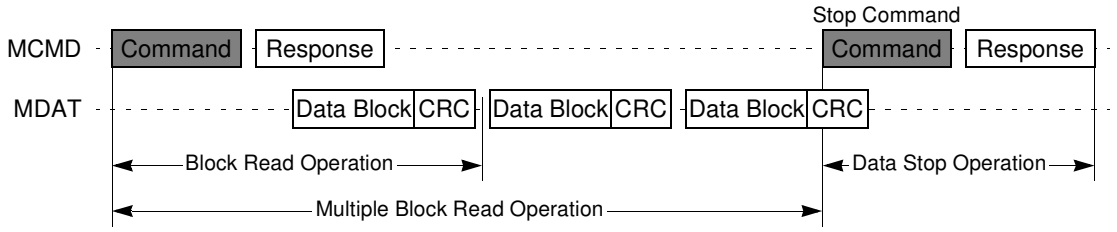


Figure 38. (Multiple) Block Read Operation



As shown in Figure 39 and Figure 40 the data write operation uses a simple busy signalling of the write operation duration on the data line (MDAT).

Figure 39. Sequential Write Operation

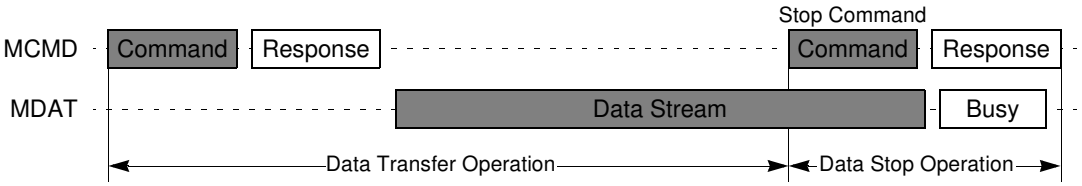


Figure 40. Multiple Block Write Operation

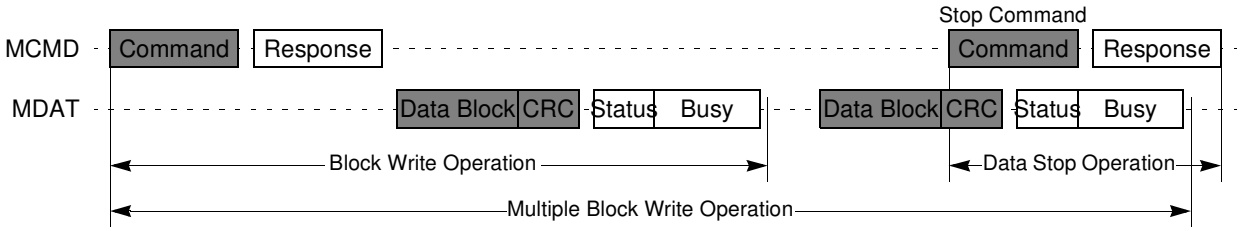
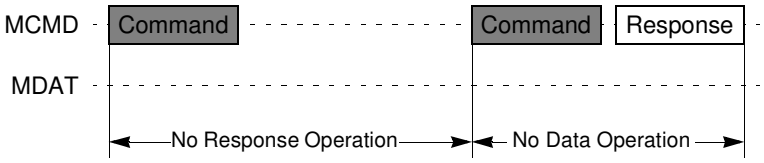


Figure 41. No Response and No Data Operation



Command Token Format

As shown in Figure 42, commands have a fixed code length of 48 bits. Each command token is preceded by a Start bit: a low level on MCMD line and succeeded by an End bit: a high level on MCMD line. The command content is preceded by a Transmission bit: a high level on MCMD line for a command token (host to card) and succeeded by a 7 - bit CRC so that transmission errors can be detected and the operation may be repeated. Command content contains the command index and address information or parameters.

Figure 42. Command Token Format

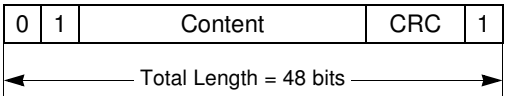


Table 48. Command Token Format

Bit Position	47	46	45:40	39:8	7:1	0
Width (Bits)	1	1	6	32	7	1
Value	'0'	'1'	-	-	-	'1'
Description	Start bit	Transmission bit	Command Index	Argument	CRC7	End bit

Response Token Format

There are five types of response tokens (R1 to R5). As shown in Figure 43, responses have a code length of 48 bits or 136 bits. A response token is preceded by a Start bit: a low level on MCMD line and succeeded by an End bit: a high level on MCMD line. The command content is preceded by a Transmission bit: a low level on MCMD line for a response token (card to host) and succeeded (R1,R2,R4,R5) or not (R3) by a 7 - bit CRC.

Response content contains mirrored command and status information (R1 response), CID register or CSD register (R2 response), OCR register (R3 response), or RCA register (R4 and R5 response).

Figure 43. Response Token Format

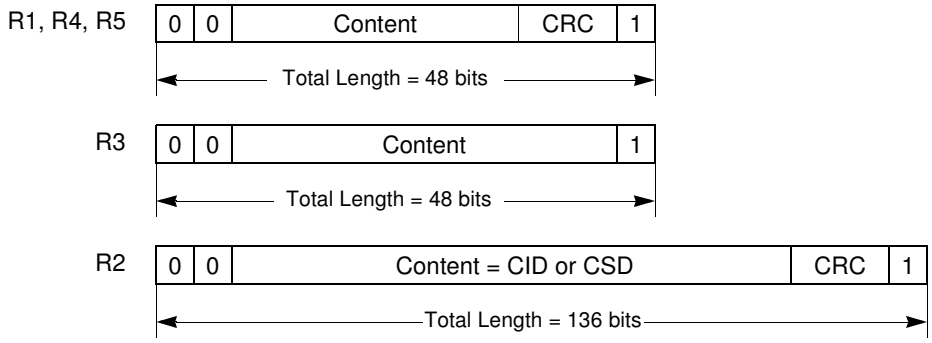


Table 49. R1 Response Format (Normal Response)

Bit Position	47	46	45:40	39:8	7:1	0
Width (bits)	1	1	6	32	7	1
Value	'0'	'0'	-	-	-	'1'
Description	Start bit	Transmission bit	Command Index	Card Status	CRC7	End bit

Table 50. R2 Response Format (CID and CSD registers)

Bit Position	135	134	[133:128]	[127:1]	0
Width (bits)	1	1	6	32	1
Value	'0'	'0'	'111111'	-	'1'
Description	Start bit	Transmission bit	Reserved	Argument	End bit

Table 51. R3 Response Format (OCR Register)

Bit Position	47	46	[45:40]	[39:8]	[7:1]	0
Width (bits)	1	1	6	32	7	1
Value	'0'	'0'	'111111'	-	'1111111'	'1'
Description	Start bit	Transmission bit	Reserved	OCR register	Reserved	End bit

Table 52. R4 Response Format (Fast I/O)

Bit Position	47	46	[45:40]	[39:8]	[7:1]	0
Width (bits)	1	1	6	32	7	1
Value	'0'	'0'	'100111'	-	-	'1'
Description	Start bit	Transmission bit	Command Index	Argument	CRC7	End bit

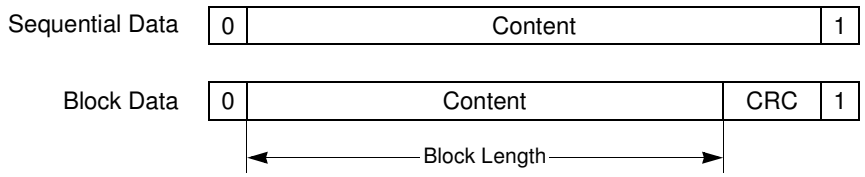
Table 53. R5 Response Format

Bit Position	47	46	[45:40]	[39:8]	[7:1]	0
Width (bits)	1	1	6	32	7	1
Value	'0'	'0'	'101000'	-	-	'1'
Description	Start bit	Transmission bit	Command Index	Argument	CRC7	End bit

Data Packet Format

There are 2 types of data packets: stream and block. As shown in Figure 44, stream data packets have an indeterminate length while block packets have a fixed length depending on the block length. Each data packet is preceded by a Start bit: a low level on MCMD line and succeeded by an End bit: a high level on MCMD line. Due to the fact that there is no predefined end in stream packets, CRC protection is not included in this case. The CRC protection algorithm for block data is a 16-bit CCITT polynomial.

Figure 44. Data Token Format



Clock Control

The MMC bus clock signal can be used by the host to turn the cards into energy saving mode or to control the data flow (to avoid under-run or over-run conditions) on the bus. The host is allowed to lower the clock frequency or shut it down.
There are a few restrictions the host must follow:

- The bus frequency can be changed at any time (under the restrictions of maximum data transfer frequency, defined by the cards, and the identification frequency defined by the specification document).
- It is an obvious requirement that the clock must be running for the card to output data or response tokens. After the last MultiMedia Card bus transaction, the host is

required, to provide 8 (eight) clock cycles for the card to complete the operation before shutting down the clock. Following is a list of the various bus transactions:

- A command with no response. 8 clocks after the host command End bit.
- A command with response. 8 clocks after the card command End bit.
- A read data transaction. 8 clocks after the End bit of the last data block.
- A write data transaction. 8 clocks after the CRC status token.
- The host is allowed to shut down the clock of a “busy” card. The card will complete the programming operation regardless of the host clock. However, the host must provide a clock edge for the card to turn off its busy signal. Without a clock edge the card (unless previously disconnected by a deselect command-CMD7) will force the MDAT line down, forever.

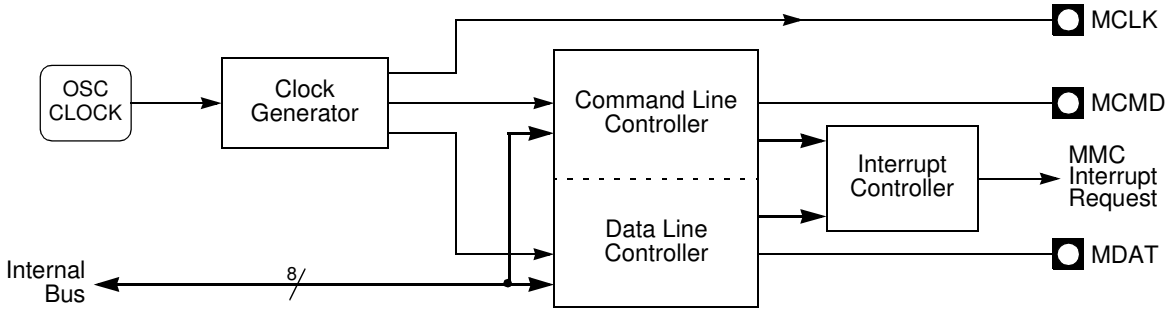
Description

The MMC controller interfaces to the C51 core through the following eight special function registers:

MMCON0, MMCON1, MMCON2, the three MMC control registers; MMSTA, the MMC status register ; MMINT, the MMC interrupt register; MMMSK, the MMC interrupt mask register; MMCMD, the MMC command register; MMDAT, the MMC data register; and MMCLK, the MMC clock register.

As shown in Figure 45, the MMC controller is divided in four blocks: the clock generator that handles the MCLK (formally the MMC CLK) output to the card, the command line controller that handles the MCMD (formally the MMC CMD) line traffic to or from the card, the data line controller that handles the MDAT (formally the MMC DAT) line traffic to or from the card, and the interrupt controller that handles the MMC controller interrupt sources. These blocks are detailed in the following sections.

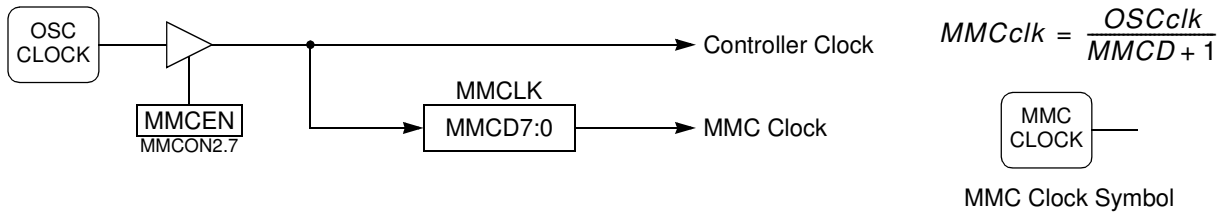
Figure 45. MMC Controller Block Diagram



Clock Generator

The MMC clock is generated by division of the oscillator clock (F_{OSC}) issued from the Clock Controller block as detailed in Section "Oscillator", page 10. The division factor is given by MMCD7:0 bits in MMCLK register, a value of 0x00 stops the MMC clock. Figure 46 shows the MMC clock generator and its output clock calculation formula.

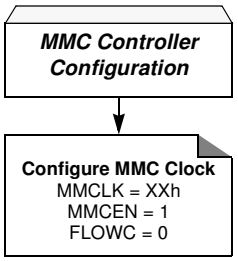
Figure 46. MMC Clock Generator and Symbol



As soon as MMCEN bit in MMCON2 is set, the MMC controller receives its system clock. The MMC command and data clock is generated on MCLK output and sent to the command line and data line controllers. Figure 47 shows the MMC controller configuration flow.

As exposed in Section “Clock Control”, page 55, MMCD7:0 bits can be used to dynamically increase or reduce the MMC clock.

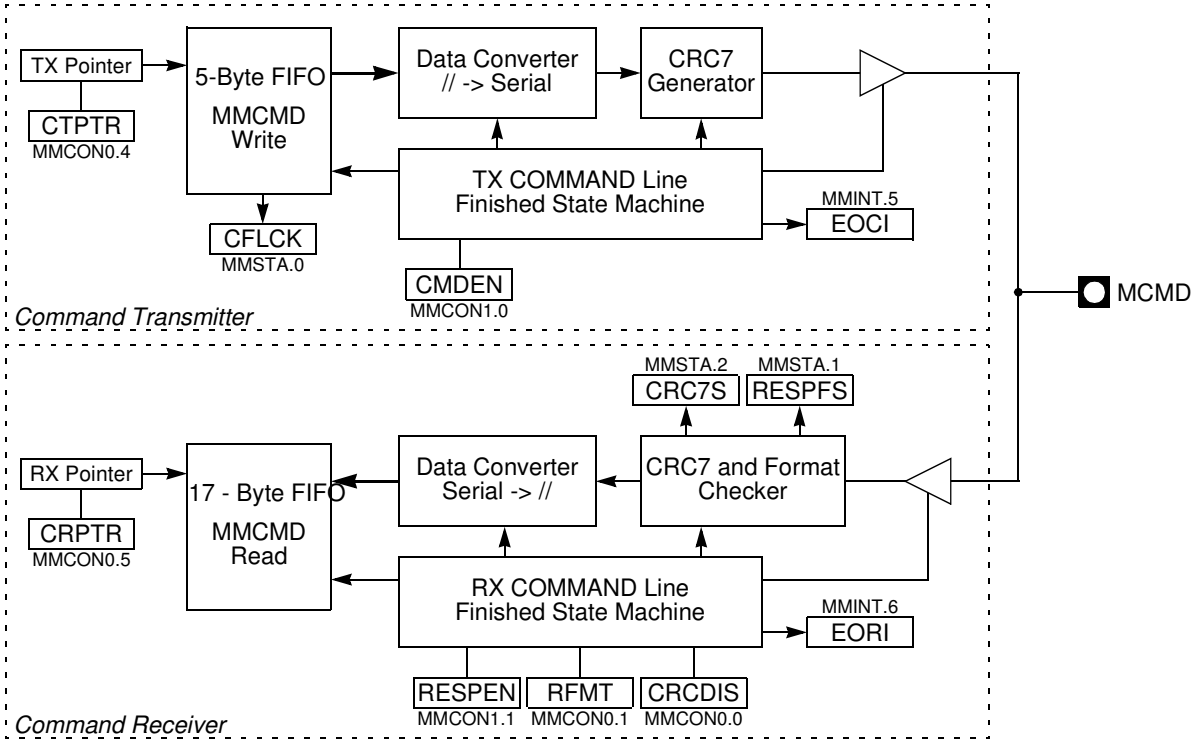
Figure 47. Configuration Flow



Command Line Controller

As shown in Figure 48, the command line controller is divided in 2 channels: the command transmitter channel that handles the command transmission to the card through the MCMD line and the command receiver channel that handles the response reception from the card through the MCMD line. These channels are detailed in the following sections.

Figure 48. Command Line Controller Block Diagram



Command Transmitter

For sending a command to the card, user must load the command index (1 Byte) and argument (4 Bytes) in the command transmit FIFO using the MMCMD register. Before starting transmission by setting and clearing the CMDEN bit in MMCON1 register, user must first configure:

- RESPEN bit in MMCON1 register to indicate whether a response is expected or not.
- RFMT bit in MMCON0 register to indicate the response size expected.
- CRCDIS bit in MMCON0 register to indicate whether the CRC7 included in the response will be computed or not. In order to avoid CRC error, CRCDIS may be set for response that do not include CRC7.

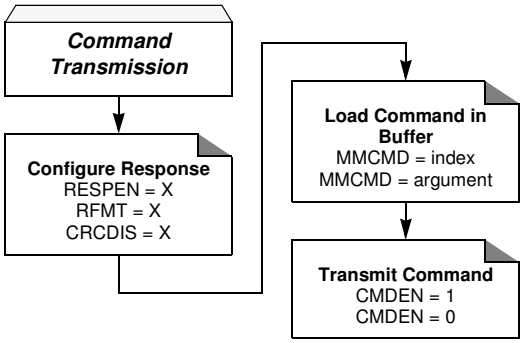
Figure 49 summarizes the command transmission flow.

As soon as command transmission is enabled, the CFLCK flag in MMSTA is set indicating that write to the FIFO is locked. This mechanism is implemented to avoid command overrun.

The end of the command transmission is signalled to you by the EOCI flag in MMINT register becoming set. This flag may generate an MMC interrupt request as detailed in Section "Interrupt", page 66. The end of the command transmission also resets the CFLCK flag.

User may abort command loading by setting and clearing the CTPTR bit in MMCON0 register which resets the write pointer to the transmit FIFO.

Figure 49. Command Transmission Flow



Command Receiver

The end of the response reception is signalled to you by the EORI flag in MMINT register. This flag may generate an MMC interrupt request as detailed in Section "Interrupt", page 66. When this flag is set, 2 other flags in MMSTA register: RESPFS and CRC7S give a status on the response received. RESPFS indicates if the response format is correct or not: the size is the one expected (48 bits or 136 bits) and a valid End bit has been received, and CRC7S indicates if the CRC7 computation is correct or not. These Flags are cleared when a command is sent to the card and updated when the response has been received.

User may abort response reading by setting and clearing the CRPTR bit in MMCON0 register which resets the read pointer to the receive FIFO.

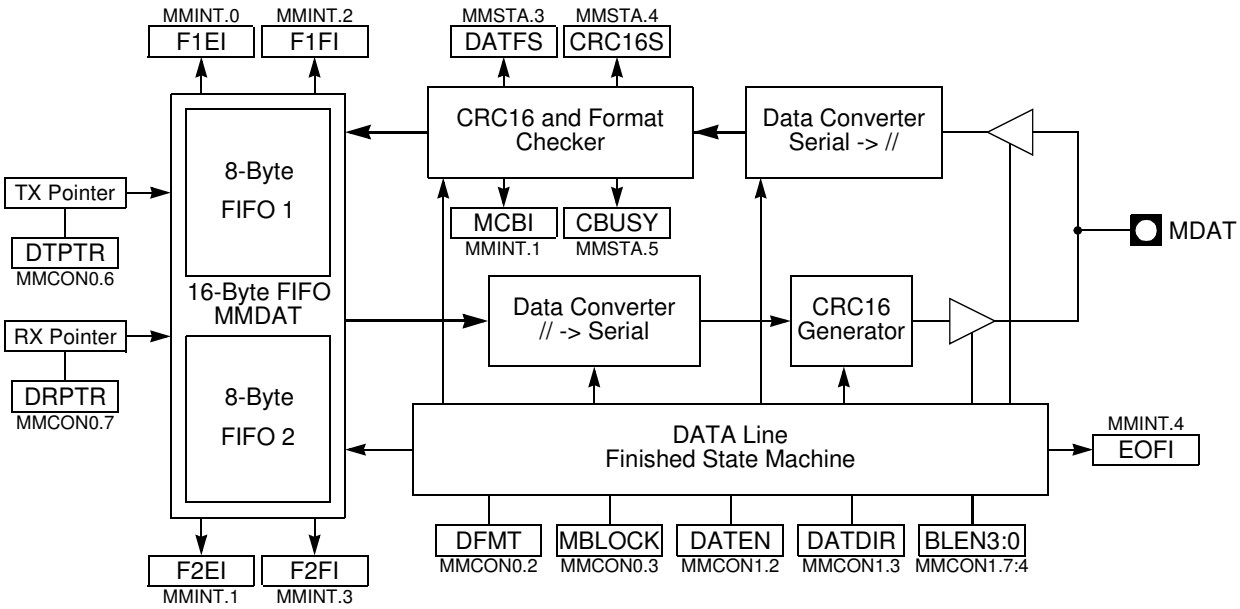
According to the MMC specification delay between a command and a response (formally N_{CR} parameter) can not exceed 64 MMC clock periods. To avoid any locking of the MMC controller when card does not send its response (e.g. physically removed from the bus), user must launch a time-out period to exit from such situation. In case of time-out user may reset the command controller and its internal state machine by setting and clearing the CCR bit in MMCON2 register.

This time-out may be disarmed when receiving the response.

Data Line Controller

The data line controller is based on a 16-Byte FIFO used both by the data transmitter channel and by the data receiver channel.

Figure 50. Data Line Controller Block Diagram



FIFO Implementation

The 16-Byte FIFO is based on a dual 8-Byte FIFOs managed using 2 pointers and four flags indicating the status full and empty of each FIFO.

Pointers are not accessible to user but can be reset at any time by setting and clearing DRPTR and DTPTR bits in MMCON0 register. Resetting the pointers is equivalent to abort the writing or reading of data.

F1EI and F2EI flags in MMINT register signal when set that respectively FIFO1 and FIFO2 are empty. F1FI and F2FI flags in MMINT register signal when set that respectively FIFO1 and FIFO2 are full. These flags may generate an MMC interrupt request as detailed in Section “Interrupt”.

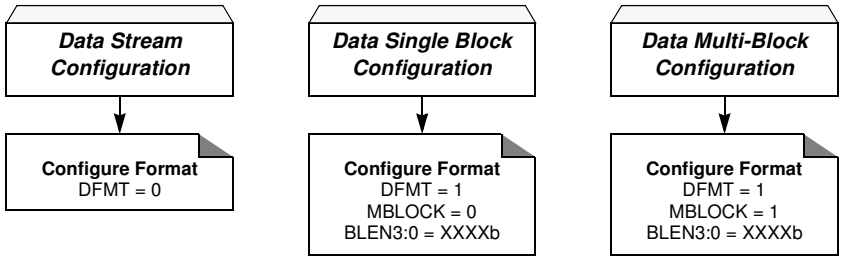
Data Configuration

Before sending or receiving any data, the data line controller must be configured according to the type of the data transfer considered. This is achieved using the Data Format bit: DFMT in MMCON0 register. Clearing DFMT bit enables the data stream format while setting DFMT bit enables the data block format. In data block format, user must also configure the single or multi-block mode by clearing or setting the MBLOCK bit in MMCON0 register and the block length using BLEN3:0 bits in MMCON1 according to Table 54. Figure 51 summarizes the data modes configuration flows.

Table 54. Block Length Programming

BLEN3:0	Block Length (Byte)
BLEN = 0000 to 1011	Length = 2^{BLEN}: 1 to 2048
> 1011	Reserved: do not program BLEN3:0 > 1011

Figure 51. Data Controller Configuration Flows



Data Transmitter

Configuration

For transmitting data to the card user must first configure the data controller in transmission mode by setting the DATDIR bit in MMCON1 register.

Figure 52 summarizes the data stream transmission flows in both polling and interrupt modes while Figure 53 summarizes the data block transmission flows in both polling and interrupt modes, these flows assume that block length is greater than 16 data.

Data Loading

Data is loaded in the FIFO by writing to MMDAT register. Number of data loaded may vary from 1 to 16 Bytes. Then if necessary (more than 16 Bytes to send) user must wait that one FIFO becomes empty (F1EI or F2EI set) before loading 8 new data.

Data Transmission

Transmission is enabled by setting and clearing DATEN bit in MMCON1 register.
Data is transmitted immediately if the response has already been received, or is delayed after the response reception if its status is correct. In both cases transmission is delayed if a card sends a busy state on the data line until the end of this busy condition.

According to the MMC specification, the data transfer from the host to the card may not start sooner than 2 MMC clock periods after the card response was received (formally N_{WR} parameter). To address all card types, this delay can be programmed using DATD1:0 bits in MMCON2 register from 3 MMC clock periods when DATD1:0 bits are cleared to 9 MMC clock periods when DATD1:0 bits are set, by step of 2 MMC clock periods.

End of Transmission

The end of a data frame (block or stream) transmission is signalled to you by the EOFI flag in MMINT register. This flag may generate an MMC interrupt request as detailed in Section "Interrupt", page 66.

In data stream mode, EOFI flag is set, after reception of the End bit. This assumes user has previously sent the STOP command to the card, which is the only way to stop stream transfer.

In data block mode, EOFI flag is set, after reception of the CRC status token (see Figure 43). 2 other flags in MMSTA register: DATFS and CRC16S report a status on the frame sent. DATFS indicates if the CRC status token format is correct or not, and CRC16S indicates if the card has found the CRC16 of the block correct or not.

Busy Status

As shown in Figure 43 the card uses a busy token during a block write operation. This busy status is reported to you by the CBUSY flag in MMSTA register and by the MCBI flag in MMINT which is set every time CBUSY toggles, i.e. when the card enters and exits its busy state. This flag may generate an MMC interrupt request as detailed in Section "Interrupt", page 66.

Figure 52. Data Stream Transmission Flows

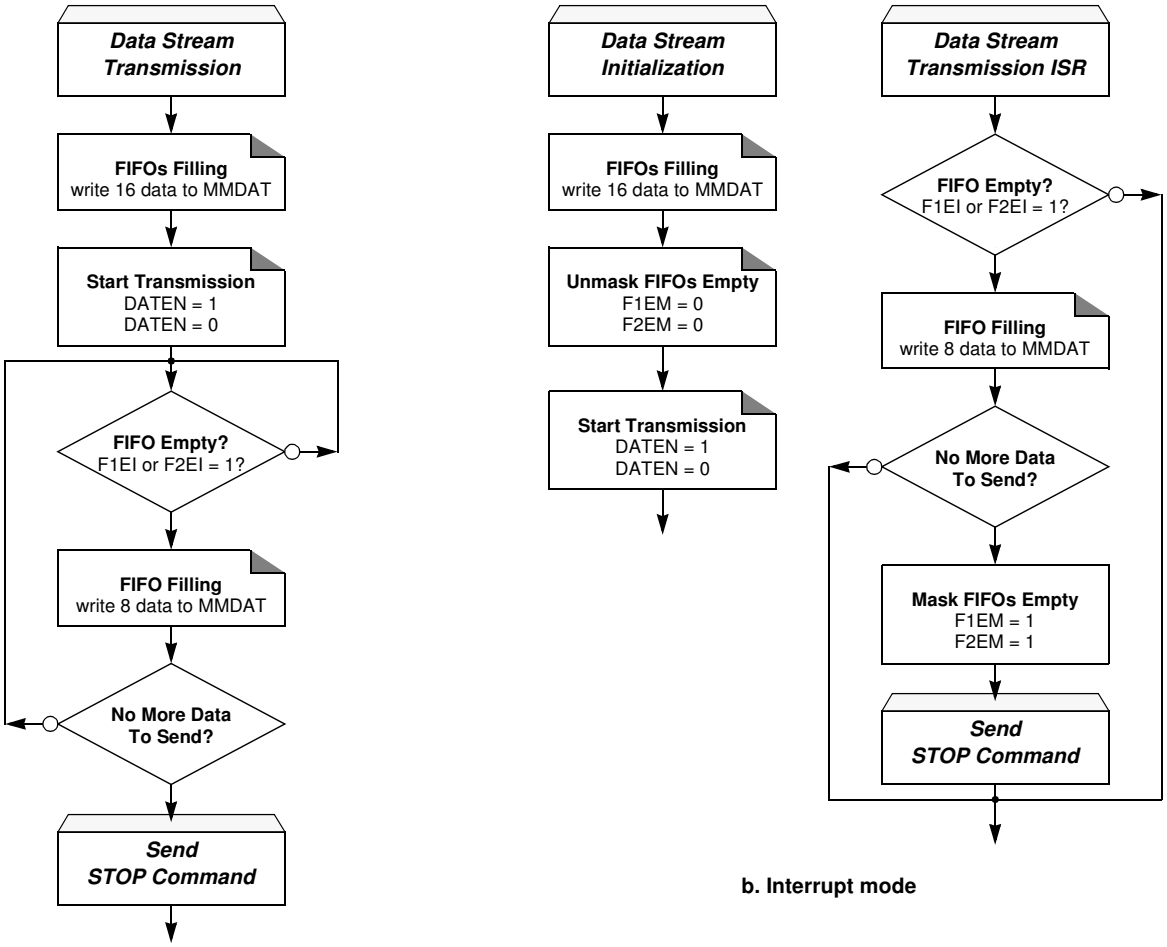
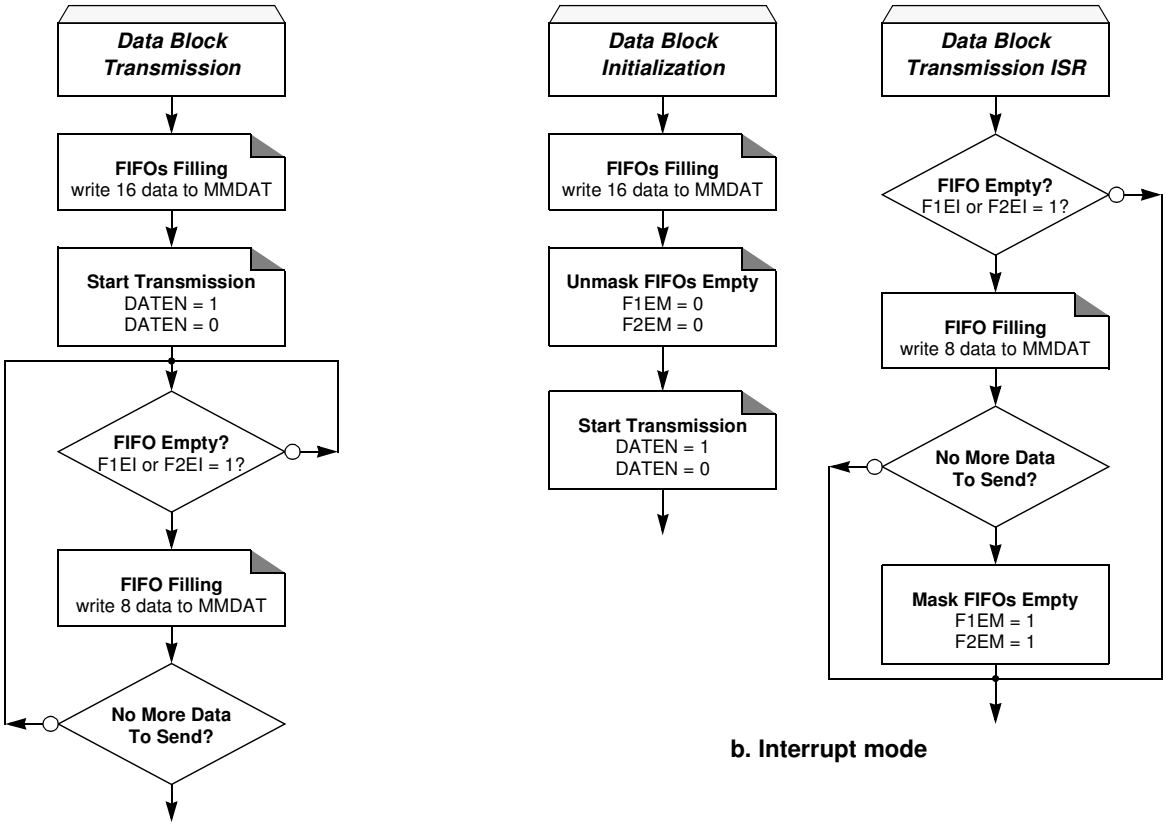


Figure 53. Data Block Transmission Flows



a. Polling mode

b. Interrupt mode

Data Receiver

Configuration

To receive data from the card you must first configure the data controller in reception mode by clearing the DATDIR bit in MMCON1 register.

Figure 54 summarizes the data stream reception flows in both polling and interrupt modes while Figure 55 summarizes the data block reception flows in both polling and interrupt modes, these flows assume that block length is greater than 16 Bytes.

Data Reception

The end of a data frame (block or stream) reception is signalled to you by the EOFI flag in MMINT register. This flag may generate an MMC interrupt request as detailed in Section "Interrupt", page 66. When this flag is set, 2 other flags in MMSTA register: DATFS and CRC16S give a status on the frame received. DATFS indicates if the frame format is correct or not: a valid End bit has been received, and CRC16S indicates if the CRC16 computation is correct or not. In case of data stream CRC16S has no meaning and stays cleared.

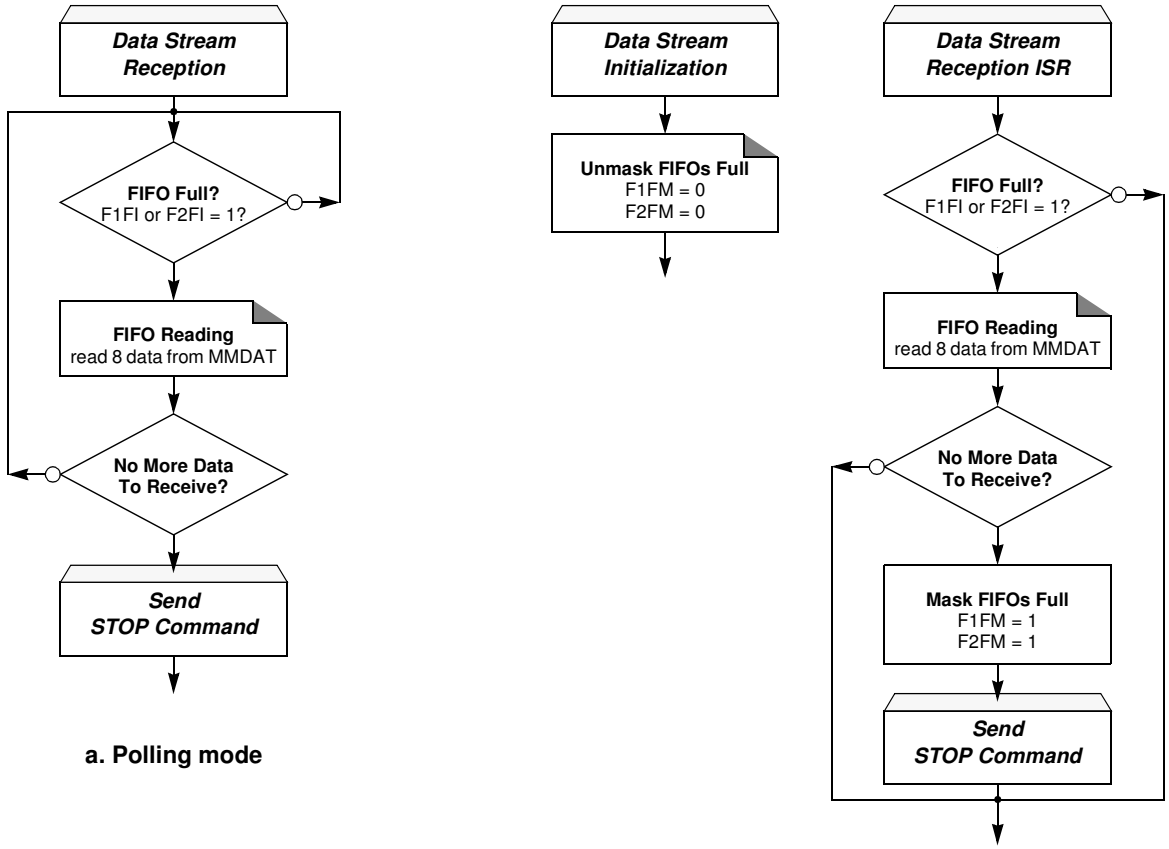
According to the MMC specification data transmission from the card starts after the access time delay (formally N_{AC} parameter) beginning from the End bit of the read command. To avoid any locking of the MMC controller when card does not send its data (e.g. physically removed from the bus), you must launch a time-out period to exit from such situation. In case of time-out you may reset the data controller and its internal state machine by setting and clearing the DCR bit in MMCON2 register.

This time-out may be disarmed after receiving 8 data (F1FI flag set) or after receiving end of frame (EOFI flag set) in case of block length less than 8 data (1, 2 or 4).

Data Reading

Data is read from the FIFO by reading to MMDAT register. Each time one FIFO becomes full (F1FI or F2FI set), user is requested to flush this FIFO by reading 8 data.

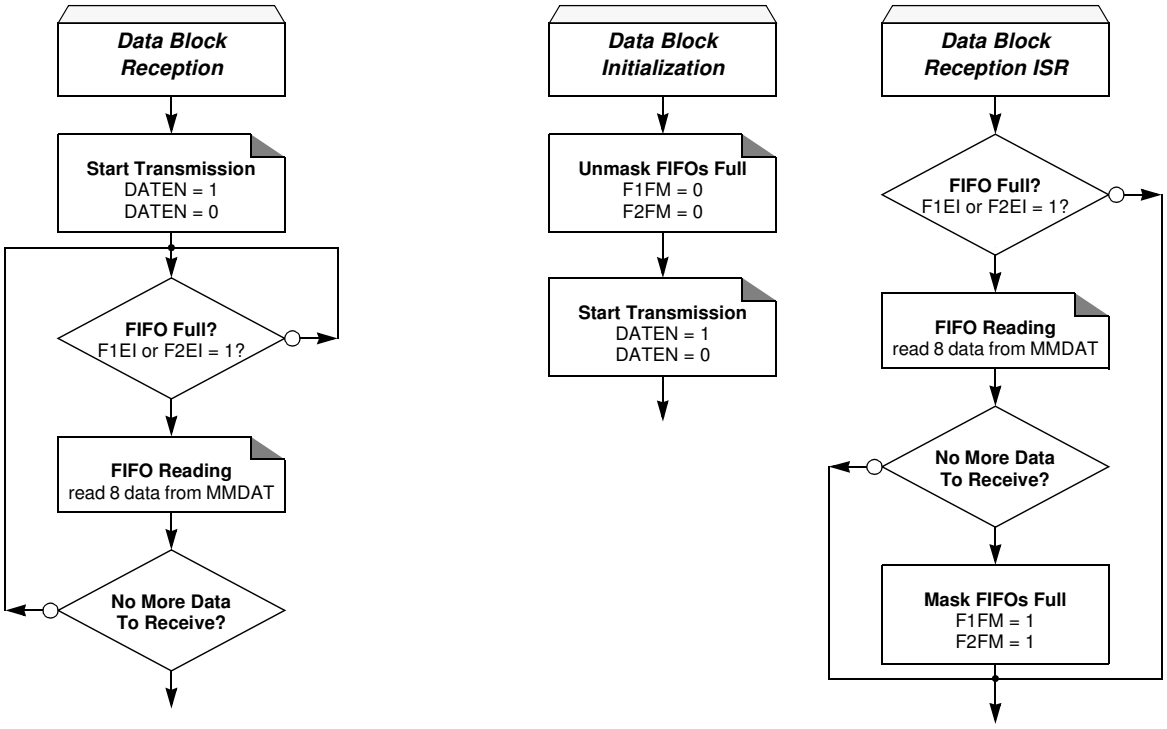
Figure 54. Data Stream Reception Flows



a. Polling mode

b. Interrupt mode

Figure 55. Data Block Reception Flows



a. Polling mode **b. Interrupt mode**

Flow Control

To allow transfer at high speed without taking care of CPU oscillator frequency, the FLOWC bit in MMCON2 allows control of the data flow in both transmission and reception.

During transmission, setting the FLOWC bit has the following effects:

- MMCLK is stopped when both FIFOs become empty: F1EI and F2EI set.
- MMCLK is restarted when one of the FIFOs becomes full: F1EI or F2EI cleared.

During reception, setting the FLOWC bit has the following effects:

- MMCLK is stopped when both FIFOs become full: F1FI and F2FI set.
- MMCLK is restarted when one of the FIFOs becomes empty: F1FI or F2FI cleared.

As soon as the clock is stopped, the MMC bus is frozen and remains in its state until the clock is restored by writing or reading data in MMDAT.

Interrupt

Description

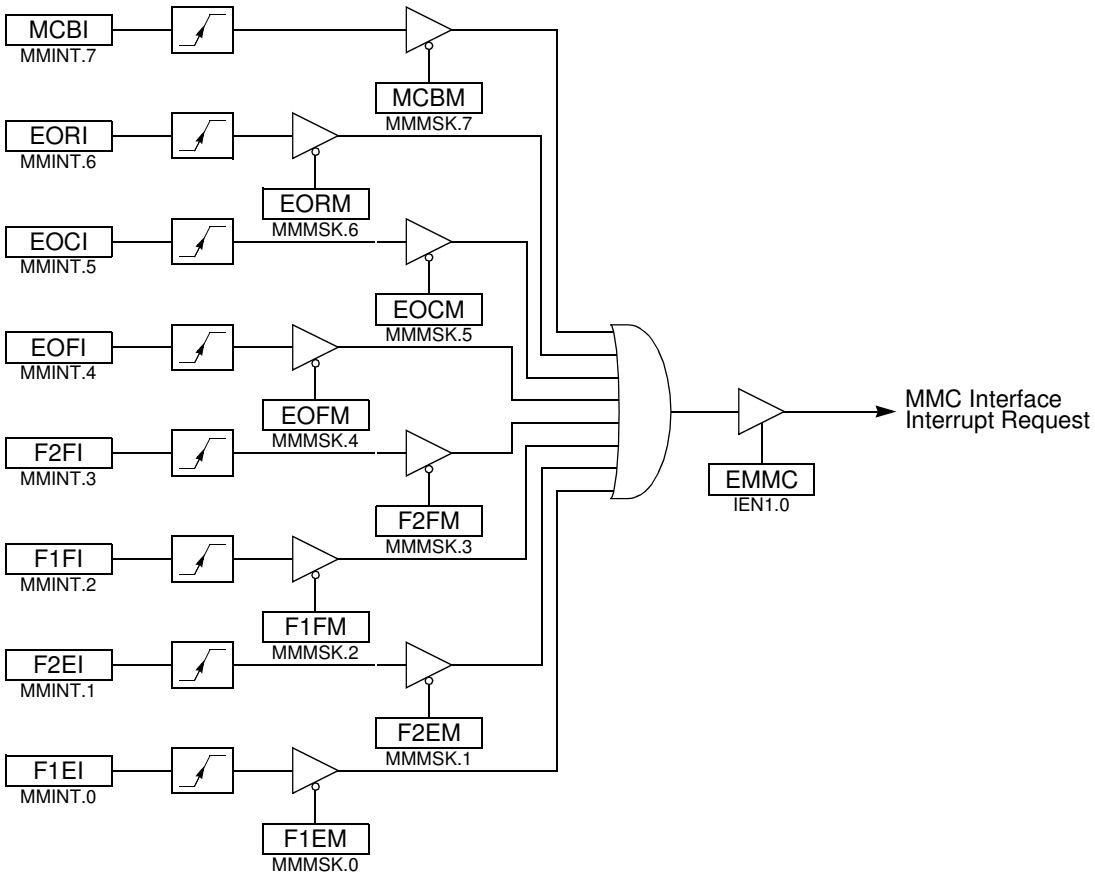
As shown in Figure 56, the MMC controller implements eight interrupt sources reported in MCBI, EORI, EOCI, EOFI, F2FI, F1FI, and F2EI flags in MMCINT register. These flags are detailed in the previous sections.

All these sources are maskable separately using MCBM, EORM, EOCM, EOFM, F2FM, F1FM, and F2EM mask bits respectively in MMMSK register.

The interrupt request is generated each time an unmasked flag is set, and the global MMC controller interrupt enable bit is set (EMMC in IEN1 register).

Reading the MMINT register automatically clears the interrupt flags (acknowledgment). This implies that register content must be saved and tested interrupt flag by interrupt flag to be sure not to forget any interrupts.

Figure 56. MMC Controller Interrupt System



Serial I/O Port

The serial I/O port in the AT83SND2CMP3 provides both synchronous and asynchronous communication modes. It operates as a Synchronous Receiver and Transmitter in one single mode (Mode 0) and operates as an Universal Asynchronous Receiver and Transmitter (UART) in three full-duplex modes (Modes 1, 2 and 3). Asynchronous modes support framing error detection and multiprocessor communication with automatic address recognition.

Mode Selection

SM0 and SM1 bits in SCON register are used to select a mode among the single synchronous and the three asynchronous modes according to Table 55.

Table 55. Serial I/O Port Mode Selection

SM0	SM1	Mode	Description	Baud Rate
0	0	0	Synchronous Shift Register	Fixed/Variable
0	1	1	8-bit UART	Variable
1	0	2	9-bit UART	Fixed
1	1	3	9-bit UART	Variable

Baud Rate Generator

Depending on the mode and the source selection, the baud rate can be generated from either the Timer 1 or the Internal Baud Rate Generator. The Timer 1 can be used in Modes 1 and 3 while the Internal Baud Rate Generator can be used in Modes 0, 1 and 3.

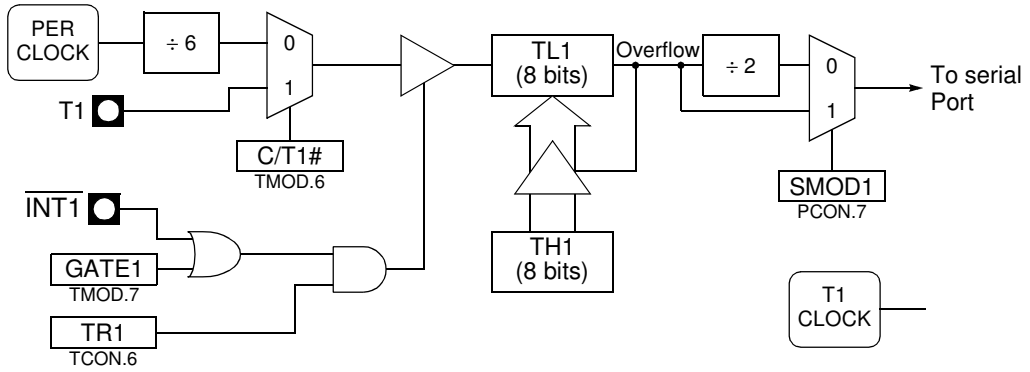
The addition of the Internal Baud Rate Generator allows freeing of the Timer 1 for other purposes in the application. It is highly recommended to use the Internal Baud Rate Generator as it allows higher and more accurate baud rates than Timer 1.

Baud rate formulas depend on the modes selected and are given in the following mode sections.

Timer 1

When using Timer 1, the Baud Rate is derived from the overflow of the timer. As shown in Figure 57 Timer 1 is used in its 8-bit auto-reload mode (detailed in Section "Mode 2 (8-bit Timer with Auto-Reload)", page 53). SMOD1 bit in PCON register allows doubling of the generated baud rate.

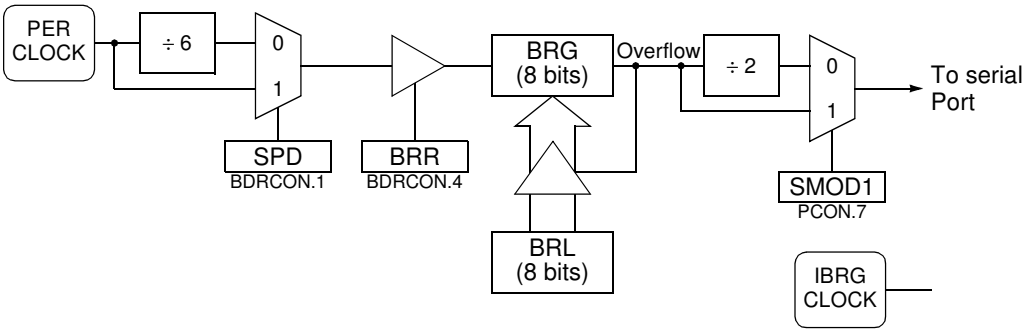
Figure 57. Timer 1 Baud Rate Generator Block Diagram



Internal Baud Rate Generator

When using the Internal Baud Rate Generator, the Baud Rate is derived from the overflow of the timer. As shown in Figure 58 the Internal Baud Rate Generator is an 8-bit auto-reload timer fed by the peripheral clock or by the peripheral clock divided by 6 depending on the SPD bit in BDRCON register. The Internal Baud Rate Generator is enabled by setting BBR bit in BDRCON register. SMOD1 bit in PCON register allows doubling of the generated baud rate.

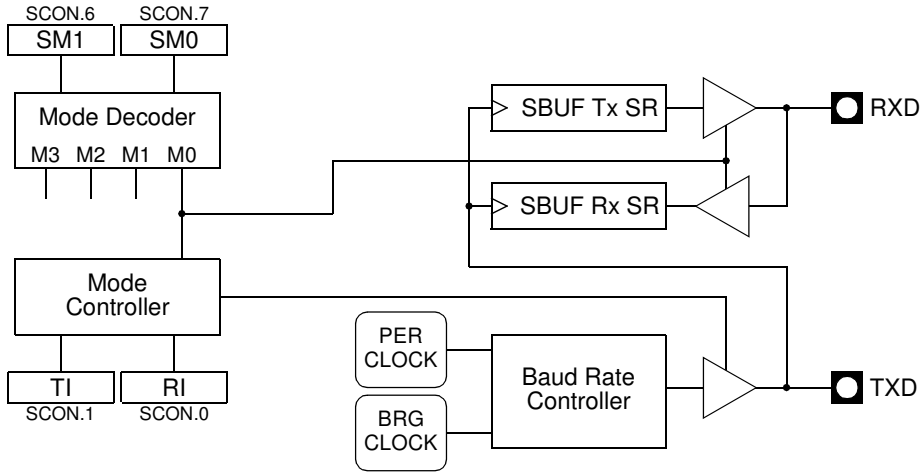
Figure 58. Internal Baud Rate Generator Block Diagram



Synchronous Mode (Mode 0)

Mode 0 is a half-duplex, synchronous mode, which is commonly used to expand the I/0 capabilities of a device with shift registers. The transmit data (TXD) pin outputs a set of eight clock pulses while the receive data (RXD) pin transmits or receives a Byte of data. The 8-bit data are transmitted and received least-significant bit (LSB) first. Shifts occur at a fixed Baud Rate (see Section "Baud Rate Selection (Mode 0)", page 69). Figure 59 shows the serial port block diagram in Mode 0.

Figure 59. Serial I/O Port Block Diagram (Mode 0)

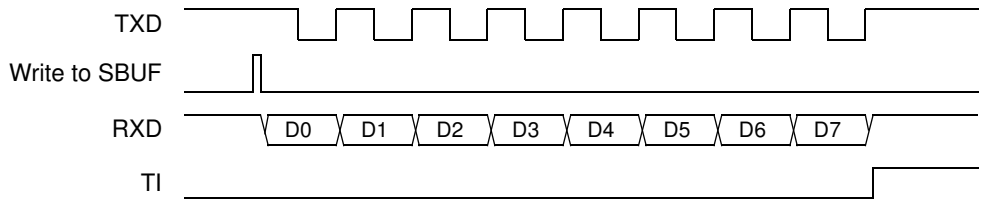


Transmission (Mode 0)

To start a transmission mode 0, write to SCON register clearing bits SM0, SM1.

As shown in Figure 60, writing the Byte to transmit to SBUF register starts the transmission. Hardware shifts the LSB (D0) onto the RXD pin during the first clock cycle composed of a high level then low level signal on TXD. During the eighth clock cycle the MSB (D7) is on the RXD pin. Then, hardware drives the RXD pin high and asserts TI to indicate the end of the transmission.

Figure 60. Transmission Waveforms (Mode 0)

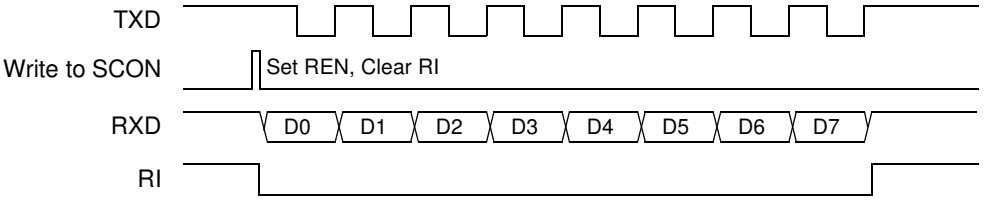


Reception (Mode 0)

To start a reception in mode 0, write to SCON register clearing SM0, SM1 and RI bits and setting the REN bit.

As shown in Figure 61, Clock is pulsed and the LSB (D0) is sampled on the RXD pin. The D0 bit is then shifted into the shift register. After eight samplings, the MSB (D7) is shifted into the shift register, and hardware asserts RI bit to indicate a completed reception. Software can then read the received Byte from SBUF register.

Figure 61. Reception Waveforms (Mode 0)



Baud Rate Selection (Mode 0)

In mode 0, the baud rate can be either, fixed or variable.

As shown in Figure 62, the selection is done using M0SRC bit in BDRCON register. Figure 63 gives the baud rate calculation formulas for each baud rate source.

Figure 62. Baud Rate Source Selection (mode 0)

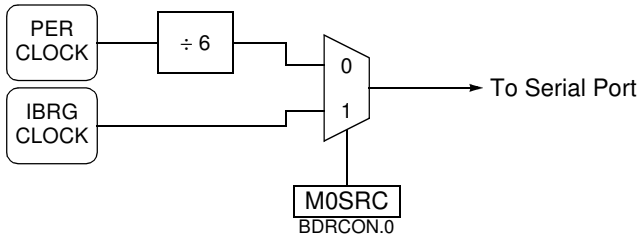


Figure 63. Baud Rate Formulas (Mode 0)

$$\text{Baud\_Rate} = \frac{F_{PER}}{6}$$

a. Fixed Formula

$$\text{Baud\_Rate} = \frac{2^{SMOD1} \cdot F_{PER}}{6^{(1-SPD)} \cdot 32 \cdot (256 - BRL)}$$

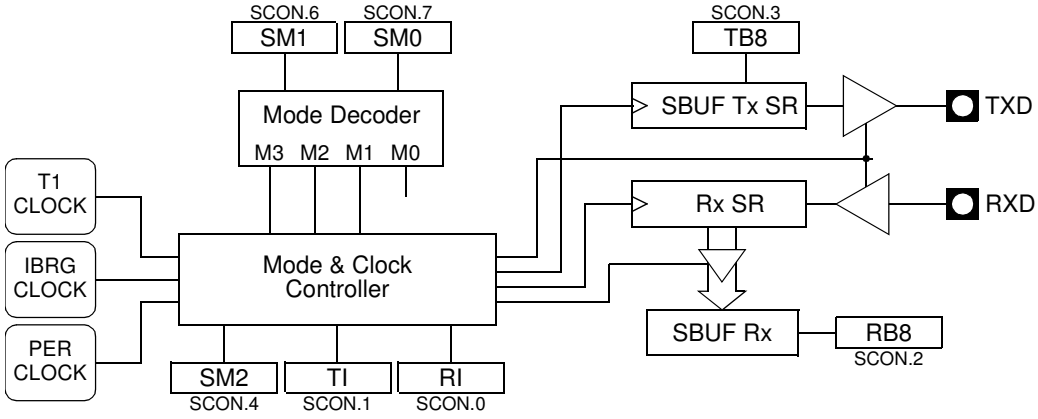
$$BRL = 256 - \frac{2^{SMOD1} \cdot F_{PER}}{6^{(1-SPD)} \cdot 32 \cdot \text{Baud\_Rate}}$$

b. Variable Formula

Asynchronous Modes (Modes 1, 2 and 3)

The Serial Port has one 8-bit and 2 9-bit asynchronous modes of operation. Figure 64 shows the Serial Port block diagram in such asynchronous modes.

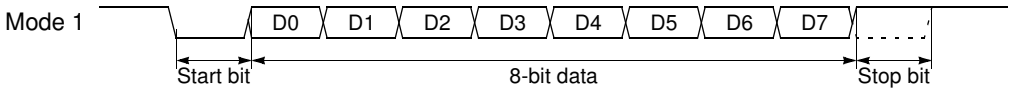
Figure 64. Serial I/O Port Block Diagram (Modes 1, 2 and 3)



Mode 1

Mode 1 is a full-duplex, asynchronous mode. The data frame (see Figure 65) consists of 10 bits: one start, eight data bits and one stop bit. Serial data is transmitted on the TXD pin and received on the RXD pin. When a data is received, the stop bit is read in the RB8 bit in SCON register.

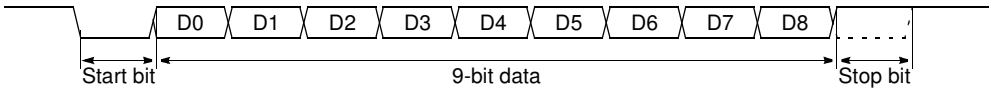
Figure 65. Data Frame Format (Mode 1)



Modes 2 and 3

Modes 2 and 3 are full-duplex, asynchronous modes. The data frame (see Figure 66) consists of 11 bits: one start bit, eight data bits (transmitted and received LSB first), one programmable ninth data bit and one stop bit. Serial data is transmitted on the TXD pin and received on the RXD pin. On receive, the ninth bit is read from RB8 bit in SCON register. On transmit, the ninth data bit is written to TB8 bit in SCON register. Alternatively, you can use the ninth bit can be used as a command/data flag.

Figure 66. Data Frame Format (Modes 2 and 3)



Transmission (Modes 1, 2 and 3)

To initiate a transmission, write to SCON register, set the SM0 and SM1 bits according to Table 55, and set the ninth bit by writing to TB8 bit. Then, writing the Byte to be transmitted to SBUF register starts the transmission.

Reception (Modes 1, 2 and 3)

To prepare for reception, write to SCON register, set the SM0 and SM1 bits according to Table 55, and set the REN bit. The actual reception is then initiated by a detected high-to-low transition on the RXD pin.

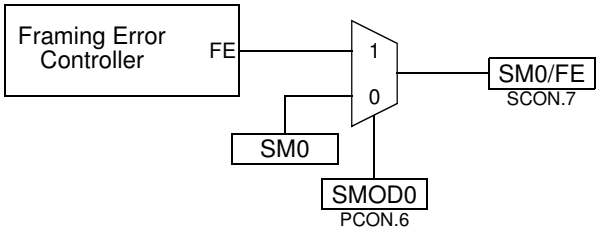
Framing Error Detection (Modes 1, 2 and 3)

Framing error detection is provided for the three asynchronous modes. To enable the framing bit error detection feature, set SMOD0 bit in PCON register as shown in Figure 67.

When this feature is enabled, the receiver checks each incoming data frame for a valid stop bit. An invalid stop bit may result from noise on the serial lines or from simultaneous transmission by 2 devices. If a valid stop bit is not found, the software sets FE bit in SCON register.

Software may examine FE bit after each reception to check for data errors. Once set, only software or a chip reset clear FE bit. Subsequently received frames with valid stop bits cannot clear FE bit. When the framing error detection feature is enabled, RI rises on stop bit instead of the last data bit as detailed in Figure 73.

Figure 67. Framing Error Block Diagram



Baud Rate Selection (Modes 1 and 3)

In modes 1 and 3, the Baud Rate is derived either from the Timer 1 or the Internal Baud Rate Generator and allows different baud rate in reception and transmission.
As shown in Figure 68 the selection is done using RBCK and TBCK bits in BDRCON register.
Figure 69 gives the baud rate calculation formulas for each baud rate source while Table 56 details Internal Baud Rate Generator configuration for different peripheral clock frequencies and giving baud rates closer to the standard baud rates.

Figure 68. Baud Rate Source Selection (Modes 1 and 3)

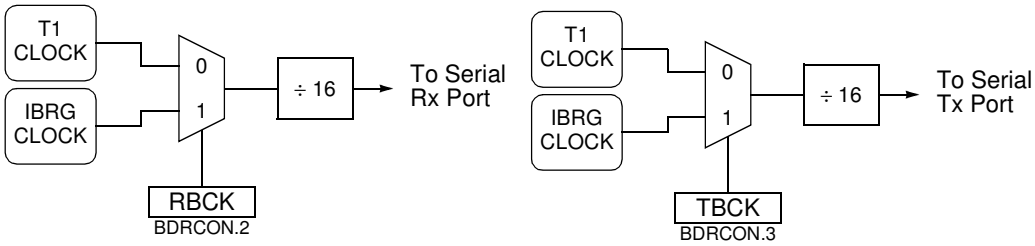


Figure 69. Baud Rate Formulas (Modes 1 and 3)

$$\text{Baud\_Rate} = \frac{2^{SMOD1} \cdot F_{PER}}{6^{(1-SPD)} \cdot 32 \cdot (256 - BRL)}$$

$$BRL = 256 - \frac{2^{SMOD1} \cdot F_{PER}}{6^{(1-SPD)} \cdot 32 \cdot \text{Baud\_Rate}}$$

a. IBRG Formula

$$\text{Baud\_Rate} = \frac{2^{SMOD1} \cdot F_{PER}}{6 \cdot 32 \cdot (256 - TH1)}$$

$$TH1 = 256 - \frac{2^{SMOD1} \cdot F_{PER}}{192 \cdot \text{Baud\_Rate}}$$

b. T1 Formula

Table 56. Internal Baud Rate Generator Value

Baud Rate	F_{PER} = 6 MHz(1)				F_{PER} = 8 MHz(1)				F_{PER} = 10 MHz(1)			
	SPD	SMOD1	BRL	Error %	SPD	SMOD1	BRL	Error %	SPD	SMOD1	BRL	Error %
115200	-	-	-	-	-	-	-	-	-	-	-	-
57600	-	-	-	-	1	1	247	3.55	1	1	245	1.36
38400	1	1	246	2.34	1	1	243	0.16	1	1	240	1.73
19200	1	1	236	2.34	1	1	230	0.16	1	1	223	1.36
9600	1	1	217	0.16	1	1	204	0.16	1	1	191	0.16
4800	1	1	178	0.16	1	1	152	0.16	1	1	126	0.16

Baud Rate	F_{PER} = 12 MHz(2)				F_{PER} = 16 MHz(2)				F_{PER} = 20 MHz(2)			
	SPD	SMOD1	BRL	Error %	SPD	SMOD1	BRL	Error %	SPD	SMOD1	BRL	Error %
115200	-	-	-	-	1	1	247	3.55	1	1	245	1.36
57600	1	1	243	0.16	1	1	239	2.12	1	1	234	1.36
38400	1	1	236	2.34	1	1	230	0.16	1	1	223	1.36
19200	1	1	217	0.16	1	1	204	0.16	1	1	191	0.16
9600	1	1	178	0.16	1	1	152	0.16	1	1	126	0.16
4800	1	1	100	0.16	1	1	48	0.16	1	0	126	0.16

Notes: 1. These frequencies are achieved in X1 mode, $F_{PER} = F_{OSC} \div 2$.
2. These frequencies are achieved in X2 mode, $F_{PER} = F_{OSC}$.

Baud Rate Selection (Mode 2)

In mode 2, the baud rate can only be programmed to 2 fixed values: 1/16 or 1/32 of the peripheral clock frequency.

As shown in Figure 70 the selection is done using SMOD1 bit in PCON register.

Figure 71 gives the baud rate calculation formula depending on the selection.

Figure 70. Baud Rate Generator Selection (Mode 2)

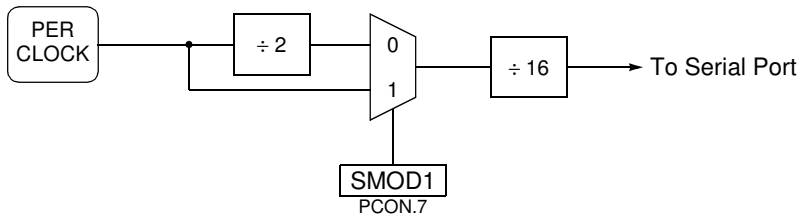


Figure 71. Baud Rate Formula (Mode 2)

$$\text{Baud\_Rate} = \frac{2^{SMOD1} \cdot F_{PER}}{32}$$

Multiprocessor Communication (Modes 2 and 3)

Modes 2 and 3 provide a ninth-bit mode to facilitate multiprocessor communication. To enable this feature, set SM2 bit in SCON register. When the multiprocessor communication feature is enabled, the serial Port can differentiate between data frames (ninth bit clear) and address frames (ninth bit set). This allows the AT83SND2CMP3 to function as a slave processor in an environment where multiple slave processors share a single serial line.

When the multiprocessor communication feature is enabled, the receiver ignores frames with the ninth bit clear. The receiver examines frames with the ninth bit set for an address match. If the received address matches the slaves address, the receiver hardware sets RB8 and RI bits in SCON register, generating an interrupt.

The addressed slave's software then clears SM2 bit in SCON register and prepares to receive the data Bytes. The other slaves are unaffected by these data Bytes because they are waiting to respond to their own addresses.

Automatic Address Recognition

The automatic address recognition feature is enabled when the multiprocessor communication feature is enabled (SM2 bit in SCON register is set).

Implemented in hardware, automatic address recognition enhances the multiprocessor communication feature by allowing the Serial Port to examine the address of each incoming command frame. Only when the Serial Port recognizes its own address, the receiver sets RI bit in SCON register to generate an interrupt. This ensures that the CPU is not interrupted by command frames addressed to other devices.

If desired, the automatic address recognition feature in mode 1 may be enabled. In this configuration, the stop bit takes the place of the ninth data bit. Bit RI is set only when the received command frame address matches the device's address and is terminated by a valid stop bit.

To support automatic address recognition, a device is identified by a given address and a broadcast address.

Note: The multiprocessor communication and automatic address recognition features cannot be enabled in mode 0 (i.e, setting SM2 bit in SCON register in mode 0 has no effect).

Given Address

Each device has an individual address that is specified in SADDR register; the SADEN register is a mask Byte that contains don't care bits (defined by zeros) to form the device's given address. The don't care bits provide the flexibility to address one or more slaves at a time. The following example illustrates how a given address is formed.
To address a device by its individual address, the SADEN mask Byte must be 1111 1111b.

For example:

```
SADDR = 0101 0110b
SADEN = 1111 1100b
Given = 0101 01XXb
```

The following is an example of how to use given addresses to address different slaves:

```
Slave A:SADDR = 1111 0001b
  SADEN = 1111 1010b
  Given = 1111 0X0Xb
Slave B:SADDR = 1111 0011b
  SADEN = 1111 1001b
  Given = 1111 0XX1b
Slave C:SADDR = 1111 0011b
  SADEN = 1111 1101b
  Given = 1111 00X1b
```

The SADEN Byte is selected so that each slave may be addressed separately.
For slave A, bit 0 (the LSB) is a don't-care bit; for slaves B and C, bit 0 is a 1. To communicate with slave A only, the master must send an address where bit 0 is clear (e.g. 1111 0000B).

For slave A, bit 1 is a 0; for slaves B and C, bit 1 is a don't care bit. To communicate with slaves A and B, but not slave C, the master must send an address with bits 0 and 1 both set (e.g. 1111 0011B).

To communicate with slaves A, B and C, the master must send an address with bit 0 set, bit 1 clear, and bit 2 clear (e.g. 1111 0001B).

Broadcast Address

A broadcast address is formed from the logical OR of the SADDR and SADEN registers with zeros defined as don't-care bits, e.g.:

```
SADDR = 0101 0110b
SADEN = 1111 1100b
(SADDR | SADEN)=1111 111Xb
```

The use of don't-care bits provides flexibility in defining the broadcast address, however in most applications, a broadcast address is FFh.

The following is an example of using broadcast addresses:

```
Slave A:SADDR = 1111 0001b
  SADEN = 1111 1010b
  Given = 1111 1X11b,

Slave B:SADDR = 1111 0011b
  SADEN = 1111 1001b
  Given = 1111 1X11b,

Slave C:SADDR = 1111 0010b
  SADEN = 1111 1101b
  Given = 1111 1111b,
```

For slaves A and B, bit 2 is a don't care bit; for slave C, bit 2 is set. To communicate with all of the slaves, the master must send the address FFh.

To communicate with slaves A and B, but not slave C, the master must send the address FBh.

Reset Address

On reset, the SADDR and SADEN registers are initialized to 00h, i.e. the given and broadcast addresses are XXXX XXXXb (all don't care bits). This ensures that the Serial Port is backwards compatible with the 80C51 microcontrollers that do not support automatic address recognition.

Interrupt

The Serial I/O Port handles 2 interrupt sources that are the “end of reception” (RI in SCON) and “end of transmission” (TI in SCON) flags. As shown in Figure 72 these flags are combined together to appear as a single interrupt source for the C51 core. Flags must be cleared by software when executing the serial interrupt service routine.
The serial interrupt is enabled by setting ES bit in IEN0 register. This assumes interrupts are globally enabled by setting EA bit in IEN0 register.

Depending on the selected mode and weather the framing error detection is enabled or disabled, RI flag is set during the stop bit or during the ninth bit as detailed in Figure 73.

Figure 72. Serial I/O Interrupt System

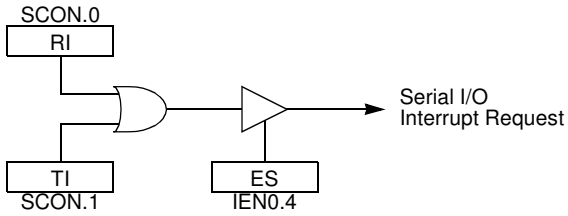
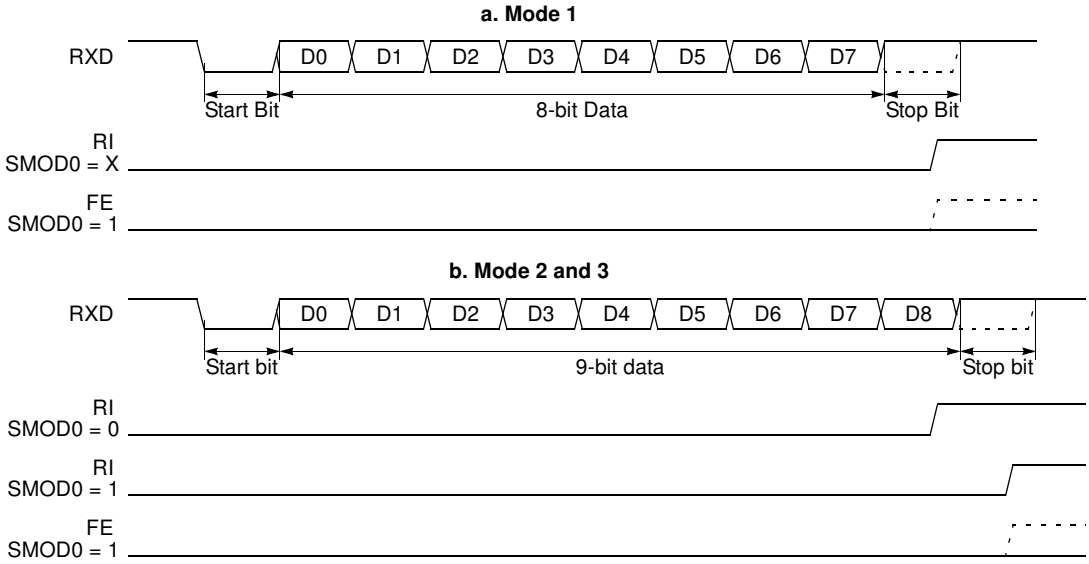


Figure 73. Interrupt Waveforms



Keyboard Interface

The AT83SND2CMP3 implement a keyboard interface allowing the connection of a keypad. It is based on one input with programmable interrupt capability on both high or low level. This input allows exit from idle and power down modes.

Description

The keyboard interfaces with the C51 core through 2 special function registers: KBCON, the keyboard control register; and KBSTA, the keyboard control and status register.

An interrupt enable bit (EKB in IEN1 register) allows global enable or disable of the keyboard interrupt (see Figure 74). As detailed in Figure 75 this keyboard input has the capability to detect a programmable level according to KINL0 bit value in KBCON register. Level detection is then reported in interrupt flag KINF0 in KBSTA register.

A keyboard interrupt is requested each time this flag is set. This flag can be masked by software using KINM0 bits in KBCON register and is cleared by reading KBSTA register.

Figure 74. Keyboard Interface Block Diagram

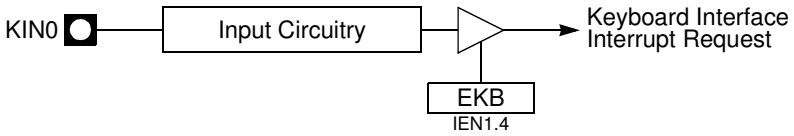
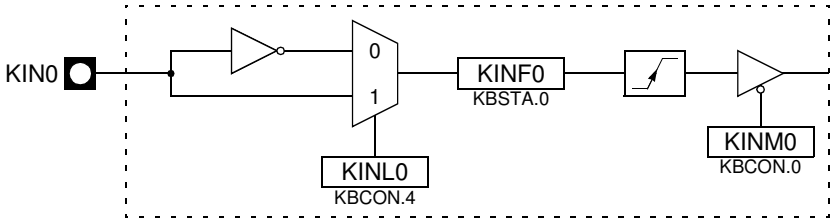


Figure 75. Keyboard Input Circuitry



Power Reduction Mode

KIN0 inputs allow exit from idle and power-down modes as detailed in section “Power Management”, page 46. To enable this feature, KPDE bit in KBSTA register must be set to logic 1.

Due to the asynchronous keypad detection in power down mode (all clocks are stopped), exit may happen on parasitic key press. In this case, no key is detected and software must enter power down again.

Electrical Characteristics

Absolute Maximum Rating

Storage Temperature ..... -65 to +150°C Voltage on any other Pin to V_{SS} ..... -0.3 to +4.0 V I_{OL} per I/O Pin ..... 5 mA Power Dissipation ..... 1 W Operating Conditions Ambient Temperature Under Bias..... -40 to +85°C V_{DD} ..... 2.7 to 3.3V	*NOTICE: 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.

DC Characteristics

Digital Logic

Table 57. Digital DC Characteristics
V_{DD} = 2.7 to 3.3 V, T_A = -40 to +85°C

Symbol	Parameter	Min	Typ(1)	Max	Units	Test Conditions
V_{IL}	Input Low Voltage	-0.5		$0.2 \cdot V_{DD}$ - 0.1	V	
V_{IH1}(2)	Input High Voltage (except RST, X1)	$0.2 \cdot V_{DD}$ + 1.1		V_{DD}	V	
V_{IH2}	Input High Voltage (RST, X1)	$0.7 \cdot V_{DD}$		V_{DD} + 0.5	V	
V_{OL1}	Output Low Voltage (except P0, MCMD, MDAT, MCLK, SCLK, DCLK, DSEL, DOUT)			0.45	V	I_{OL}= 1.6 mA
V_{OL2}	Output Low Voltage (P0, MCMD, MDAT, MCLK, SCLK, DCLK, DSEL, DOUT)			0.45	V	I_{OL}= 3.2 mA
V_{OH1}	Output High Voltage (P1, P2, P3, P4 and P5)	V_{DD} - 0.7			V	I_{OH}= -30 µA
V_{OH2}	Output High Voltage (P0, P2 address mode, MCMD, MDAT, MCLK, SCLK, DCLK, DSEL, DOUT, D+, D-)	V_{DD} - 0.7			V	I_{OH}= -3.2 mA
I_{IL}	Logical 0 Input Current (P1, P2, P3, P4 and P5)			-50	µA	V_{IN}= 0.45 V
I_{LI}	Input Leakage Current (P0, MCMD, MDAT, MCLK, SCLK, DCLK, DSEL, DOUT)			10	µA	0.45< V_{IN}< V_{DD}
I_{TL}	Logical 1 to 0 Transition Current (P1, P2, P3, P4 and P5)			-650	µA	V_{IN}= 2.0 V
R_{RST}	Pull-Down Resistor	50	90	200	kΩ	
C_{IO}	Pin Capacitance		10		pF	T_A= 25°C
V_{RET}	V_{DD} Data Retention Limit			1.8	V	
I_{DD}						

Table 57. Digital DC Characteristics
V_{DD} = 2.7 to 3.3 V, T_A = -40 to +85°C

Symbol	Parameter	Min	Typ(1)	Max	Units	Test Conditions
I_{DD}	AT83SND2CMP3 Operating Current			X1 / X2 mode 7/ 11.5 9/ 14.5 10.5 / 18	mA	V_{DD} < 3.3 V 12 MHz 16 MHz 20 MHz
I_{DL}	AT83SND2CMP3 Idle Mode Current			X1 / X2 mode 6.3 / 9.1 7.4 / 11.3 8.5 / 14	mA	V_{DD} < 3.3 V 12 MHz 16 MHz 20 MHz
I_{PD}	AT83SND2CMP3 Power-Down Mode Current		20	500	µA	V_{RET} < V_{DD} < 3.3 V

Notes: 1. Typical values are obtained using V_{DD}= 3 V and T_A= 25°C. They are not tested and there is no guarantee on these values.

Table 58. Typical Reference Design AT83SND2CMP3 Power Consumption

Player Mode	I_{DD}	Test Conditions
Stop	10 mA	AT83SND2CMP3 at 16 MHz, X2 mode, V_{DD}= 3 V No song playing. This consumption does not include AUDVBAT current.
Playing	37 mA	AT83SND2CMP3 at 16 MHz, X2 mode, V_{DD}= 3 V MP3 Song with Fs= 44.1 KHz, at any bit rates (Variable Bit Rate) This consumption does not include AUDVBAT current.

I_{DD}, I_{DL} and I_{PD} Test Conditions

Figure 76. I_{DD} Test Condition, Active Mode

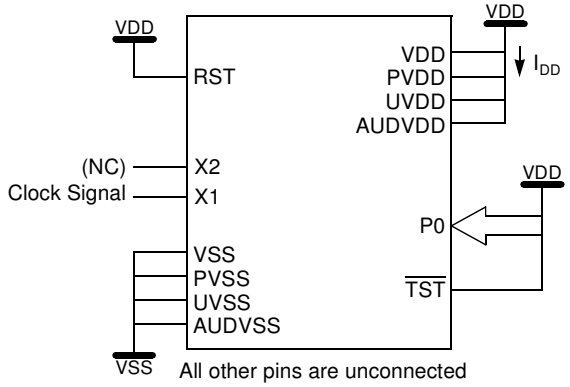


Figure 77. I_{DL} Test Condition, Idle Mode

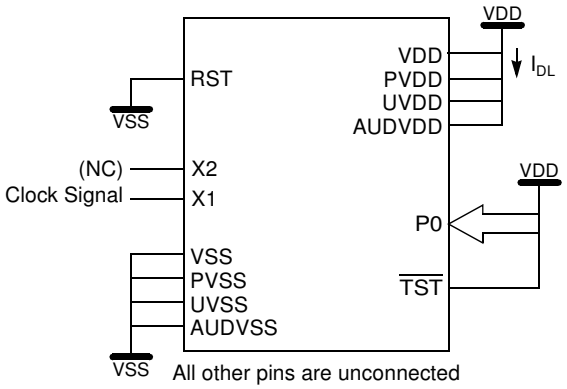
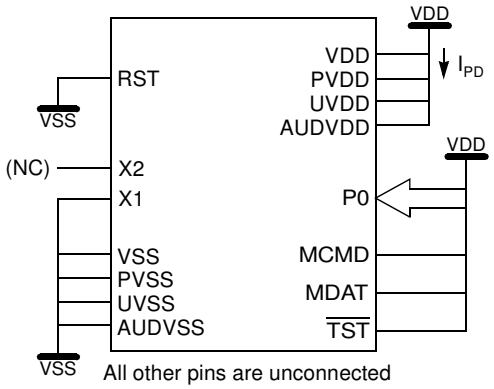


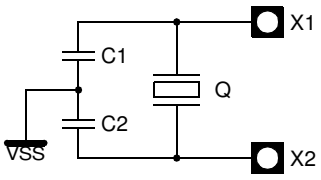
Figure 78. I_{PD} Test Condition, Power-Down Mode



Oscillator & Crystal

Schematic

Figure 79. Crystal Connection



Note: For operation with most standard crystals, no external components are needed on X1 and X2. It may be necessary to add external capacitors on X1 and X2 to ground in special cases (max 10 pF). X1 and X2 may not be used to drive other circuits.

Parameters

Table 59. Oscillator & Crystal Characteristics

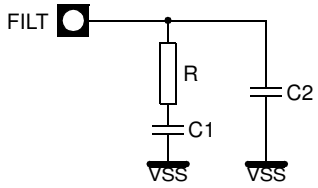
V_{DD} = 2.7 to 3.3 V, T_A = -40 to +85°C

Symbol	Parameter	Min	Typ	Max	Unit
C_{X1}	Internal Capacitance (X1 - VSS)		10		pF
C_{X2}	Internal Capacitance (X2 - VSS)		10		pF
C_L	Equivalent Load Capacitance (X1 - X2)		5		pF
DL	Drive Level			50	µW
F	Crystal Frequency			20	MHz
RS	Crystal Series Resistance			40	Ω
CS	Crystal Shunt Capacitance			6	pF

Phase Lock Loop

Schematic

Figure 80. PLL Filter Connection



Parameters

Table 60. PLL Filter Characteristics

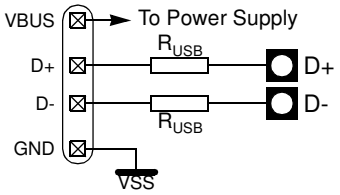
V_{DD} = 2.7 to 3.3 V, T_A = -40 to +85°C

Symbol	Parameter	Min	Typ	Max	Unit
R	Filter Resistor		100		Ω
C1	Filter Capacitance 1		10		nF
C2	Filter Capacitance 2		2.2		nF

USB Connection

Schematic

Figure 81. USB Connection



Parameters

Table 61. USB Termination Characteristics

V_{DD} = 2.7 to 3.3 V, T_A = -40 to +85°C

Symbol	Parameter	Min	Typ	Max	Unit
R_{USB}	USB Termination Resistor		27		Ω

DAC and PA

Electrical Specifications

PA

AUDVBAT = 3.6V, TA = 25°C unless otherwise noted.

High power mode, 100nF capacitor connected between CBP and AUDVSS, 470nF input capacitors, Load = 8 ohms.

Figure 82. PA Specification

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
AUDVBAT	Supply Voltage		3.2	-	5.5	V
I_{DD}	Quiescent Current	Inputs shorted, no load	-	6	8	mA
I_{DDstby}	Standby Current	Capacitance	-	-	2	µA
V_{CBP}	DC Reference		-	AUDVBAT/2	-	V
VOS	Output differential offset	full gain	-20	0	20	mV
Z_{IN}	Input impedance	Active state	12K	20k	30k	W
Z_{LFP}	Output load	Full Power mode	6	8	32	W
Z_{LLP}	Output load	Low-Power mode	100	150	300	W
C_L	Capacitive load		-	-	100	pF
PSRR	Power supply rejection ratio	200 – 2kHz Differential output	-	60	-	dB
BW	Output Frequency bandwidth	1KHz reference frequency 3dB attenuation. 470nF input coupling capacitors	50	-	20000	Hz
t_{UP}	Output setup time	Off to on mode. Voltage already settled. Input capacitors precharged	-	-	10	ms
V_N	Output noise	Max gain, A weighted	-	120	500	μV_{RMS}
THD_{HP}	Output distortion	High power mode, V_{DD} = 3.2V, 1KHz, Pout=100mW, gain=0dB	-	50	-	dB
THD_{LP}	Output distortion	Low power mode, VDD = 3.2V , 1KHz, Vout= 100mVpp, Max gain, load 8 ohms in serie with 200 ohms	-	1	-	%
G_{ACC}	Overall Gain accuracy		-2	0	2	dB
G_{STEP}	Gain Step Accuracy		-0.7	0	0.7	dB

Figure 83. Maximum Dissipated Power Versus Power Supply

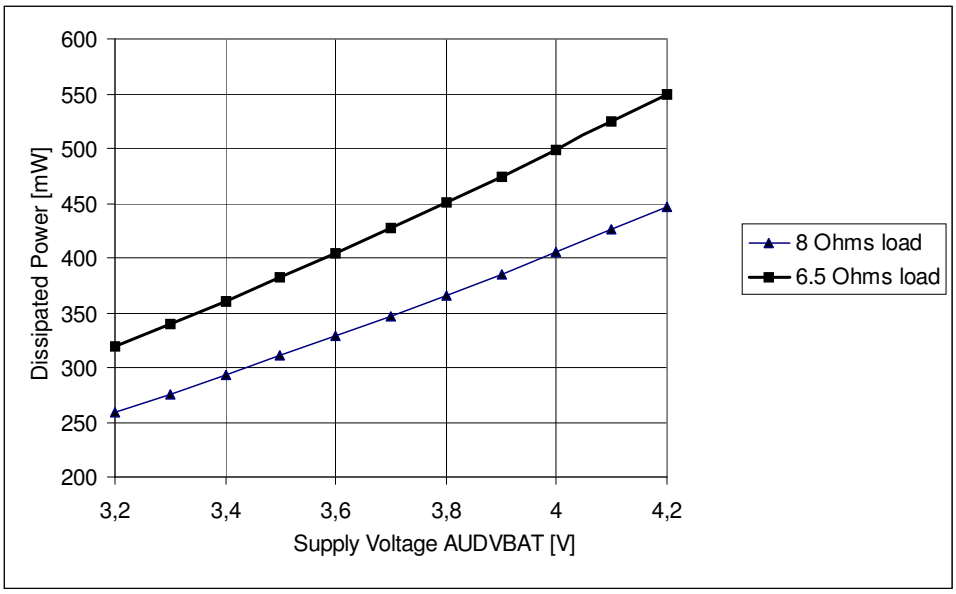
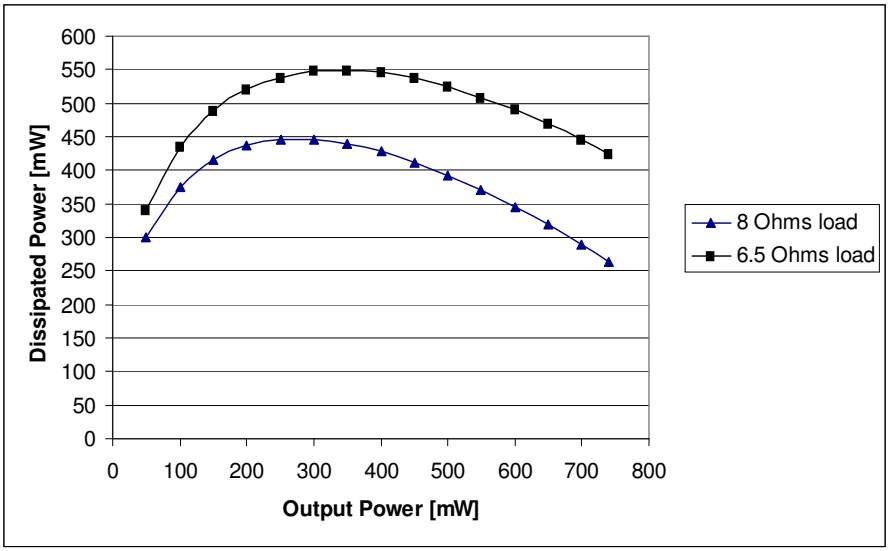


Figure 84. Dissipated Power vs Output Power, AUDVBAT = 3.2V



DAC

AUDVDD, **HSVDD** = 2.8 V, Ta=25°C, typical case, unless otherwise noted

All noise and distortion specifications are measured in the 20 Hz to 0.425xFs and A-weighted filtered.

Full Scale levels scale proportionally with the analog supply voltage.

Figure 85. Audio DAC Specification

OVERALL	**MIN**	**TYP**	**MAX**	**UNITS**
Operating Temperature	-40	+25	+125	°C
Analog Supply Voltage (**AUDVDD, HSVDD**)	2.7	2.8	3.3	V

OVERALL	MIN	TYP	MAX	UNITS
Digital Supply Voltage (**VDD**)	2.4	2.8	3.3	V
Audio Amplifier Supply (**AUDVBAT**)	3.2	-	5.5	V
DIGITAL INPUTS/OUTPUTS				
Resolution	20			Bits
Logic Family	CMOS			
Logic Coding	2's Complement			
ANALOG PERFORMANCE – DAC to Line-out/Headphone Output				
Output level for full scale input (for **AUDVDD, HSVDD** = 2.8 V)		1.65		Vpp
Output common mode voltage		0.5x**HSVDD**		V
Output load resistance (on **HSL**, **HSR**) - Headphone load - Line load	16	32 10		Ohm kOhm
Output load capacitance (on **HSL**, **HSR**) - Headphone load - Line load		30 30	1000 150	pF pF
Signal to Noise Ratio (–1dBFS @ 1kHz input and 0dB Gain) - Line and Headphone loads	87	92		dB
Total Harmonic Distortion (–1dBFS @ 1kHz input and 0dB Gain) - Line Load - Headphone Load - Headphone Load (16 Ohm)		-80 -65 -40	-76 -60	dB dB dB
Dynamic Range (measured with -60 dBFS @ 1kHz input, extrapolated to full-scale) - Line Load - Headphone Load	88 70	93 74		dB dB
Interchannel mismatch		0.1	1	dB
Left-channel to right-channel crosstalk (@ 1kHz)		-90	-80	dB
Output Power Level Control Range	-6	-	6	dB
Output Power Level Control Step		3		dB
PSRR - 1kHz - 20kHz		55 50		dB dB
Maximum output slope at power up (100 to 220F coupling capacitor)			3	V/s
ANALOG PERFORMANCE – Line-in/Microphone Input to Line-out/Headphone Output				

OVERALL	MIN	TYP	MAX	UNITS
Input level for full scale output - 0dBFS Level @ **AUDVDD, HSVDD** = 2.8 V and 0 dB gain @ **AUDVDD, HSVDD** = 2.8 V and 20 dB gain		1.65 583 0.165 58.3		Vpp mVrms Vpp mVrms
Input common mode voltage		0.5x**AUDVDD**		V
Input impedance	7	10		kOhm
Signal to Noise Ratio -1 dBFS @ 1kHz input and 0 dB gain -21 dBFS @ 1kHz input and 20 dB gain	81	85 71		dB
Dynamic Range (extrapolated to full scale level) -60 dBFS @ 1kHz input and 0 dB gain -60 dBFS @ 1kHz input and 20 dB gain	82	86 72		dB
Total Harmonic Distortion –1dBFS @ 1kHz input and 0 dB gain –1dBFS @ 1kHz input and 20 dB gain		-80 -75	-76 -68	dB
Interchannel mismatch		0.1	1	dB
Left-channel to right-channel crosstalk (@ 1kHz)		-90	-80	dB
ANALOG PERFORMANCE – Differential mono input amplifier				
Differential input level for full scale output - 0dBFS Level @ **AUDVDD, HSVDD** = 2.8 V and 0 dB gain		1.65 583		Vppdif mVrms
Input common mode voltage		0.5x**AUDVDD**		V
Input impedance	7	10		kOhm
Signal to Noise Ratio (-1 dBFS @ 1kHz input and 0 dB gain)	76	80		dB
Total Harmonic Distortion (–1dBFS @ 1kHz input and 0 dB gain)		-85	-81	dB
ANALOG PERFORMANCE – PA Driver				
Differential output level for full scale input (for **AUDVDD, HSVDD** = 3 V)		3.3		Vppdif
Output common mode voltage		0.5x**HSVDD**		V
Output load		10	30	kOhm pF
Signal to Noise Ratio (–1dBFS @ 1kHz input and 0dB Gain)	76	80		dB
Total Harmonic Distortion (–1dBFS @ 1kHz input and 0dB Gain)		-75	-71	dB
MASTER CLOCK				
Master clock Maximum Long Term Jitter			1.5	ns_{pp}

OVERALL	MIN	TYP	MAX	UNITS
DIGITAL FILTER PERFORMANCE				
Frequency response (10 Hz to 20 kHz)		+/- 0.1		dB
Deviation from linear phase (10 Hz to 20 kHz)		+/- 0.1		deg
Passband 0.1 dB corner		0.4535		Fs
Stopband	0.5465			Fs
Stopband Attenuation	65			dB
DE-EMPHASIS FILTER PERFORMANCE (for 44.1kHz Fs)				
	Frequency		Gain	Margin
Pass band Transition band Stop Band	0Hz to 3180Hz 3180Hz to 10600Hz 10600Hz to 20kHz		-1dB Logarithm decay -10.45dB	1dB 1dB 1dB
Power Performance				
Current consumption from Audio Analog supply **AVDD**, **HSVDD** in power on		9.5		mA
Current consumption from Audio Analog supply **AVDD**, **HSVDD** in power down			10	µA
Power on Settling Time - From full Power Down to Full Power Up (AUDVREF and AUDVCM decoupling capacitors charge) - Linein amplifier (Line-in coupling capacitors charge) - Driver amplifier (out driver DC blocking capacitors charge)		 500 50 500		 ms ms ms

Digital Filters Transfer Function

Figure 86. Channel Filter

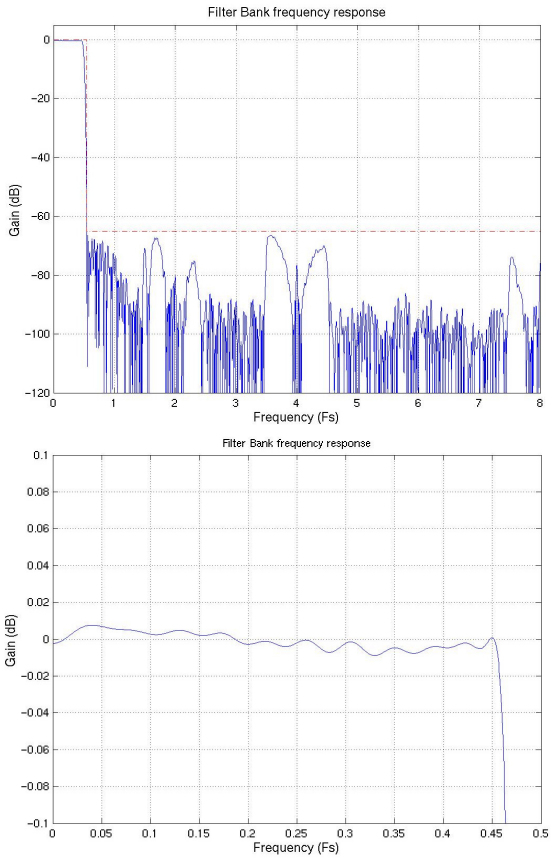
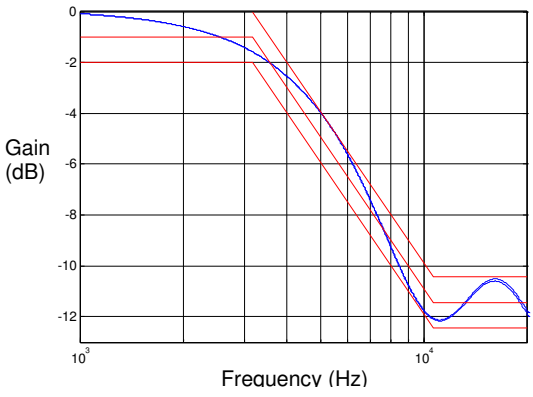


Figure 87. De-emphasis Filter



Audio DAC and PA Connection **Figure 88.** DAC and PA Connection

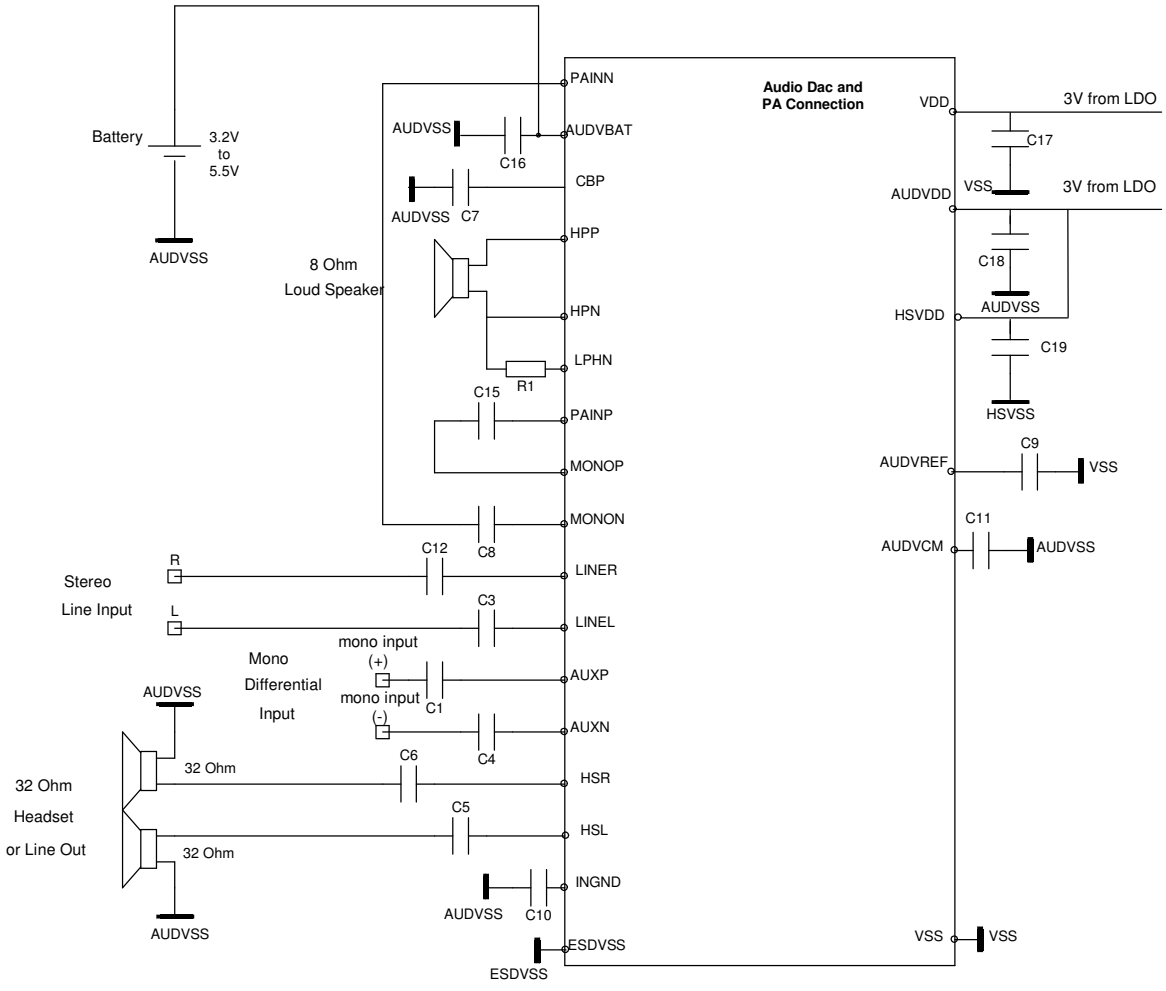


Table 62. DAC and PA Characteristics

Symbol	Parameter	Typ	Unit
C1	Capacitance	470	nF
C3	Capacitance	470	nF
C4	Capacitance	470	nF
C5	Capacitance	100	µF
C6	Capacitance	100	µF
C7	Capacitance	100	nF
C8	Capacitance	470	nF
C9	Capacitance	100n	µF
C10	Capacitance	10	µF
C11	Capacitance	10	µF
C12	Capacitance	470	nF
C15	Capacitance	470	nF
C16	Capacitance	22	µF
C17	Capacitance	100	nF
C18	Capacitance	100	nF
C19	Capacitance	100	nF
R1	Resistor	200	Ω

MMC Interface

Definition of symbols

Table 63. MMC Interface Timing Symbol Definitions

Signals	
C	Clock
D	Data In
O	Data Out

Conditions	
H	High
L	Low
V	Valid
X	No Longer Valid

Timings

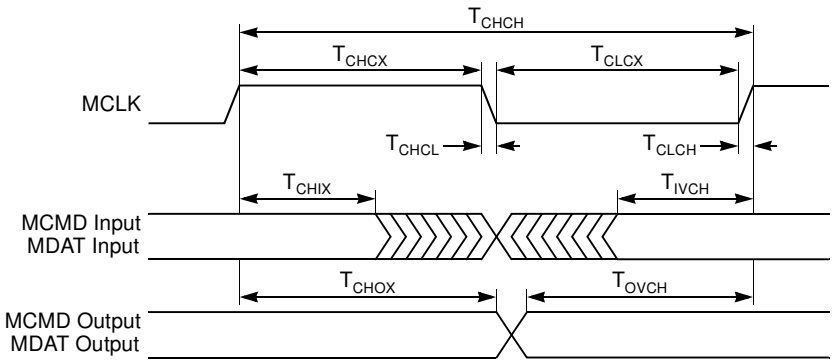
Table 64. MMC Interface AC timings

V_{DD} = 2.7 to 3.3 V, T_A = -40 to +85°C, CL ≤ 100pF (10 cards)

Symbol	Parameter	Min	Max	Unit
T_{CHCH}	Clock Period	50		ns
T_{CHCX}	Clock High Time	10		ns
T_{CLCX}	Clock Low Time	10		ns
T_{CLCH}	Clock Rise Time		10	ns
T_{CHCL}	Clock Fall Time		10	ns
T_{DVCH}	Input Data Valid to Clock High	3		ns
T_{CHDX}	Input Data Hold after Clock High	3		ns
T_{CHOX}	Output Data Hold after Clock High	5		ns
T_{OVCH}	Output Data Valid to Clock High	5		ns

Waveforms

Figure 89. MMC Input-Output Waveforms



Audio Interface

Definition of symbols

Table 65. Audio Interface Timing Symbol Definitions

Signals	
C	Clock
O	Data Out
S	Data Select

Conditions	
H	High
L	Low
V	Valid
X	No Longer Valid

Timings

Table 66. Audio Interface AC timings

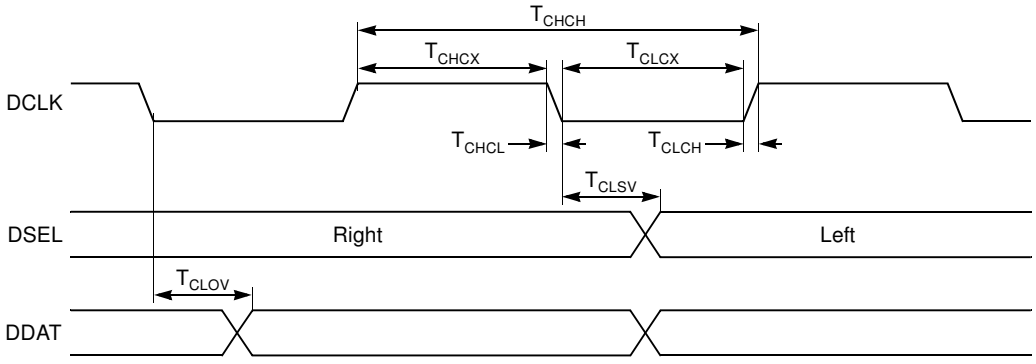
V_{DD} = 2.7 to 3.3 V, T_A = -40 to +85°C, CL≤ 30pF

Symbol	Parameter	Min	Max	Unit
T_{CHCH}	Clock Period		325.5[1]	ns
T_{CHCX}	Clock High Time	30		ns
T_{CLCX}	Clock Low Time	30		ns
T_{CLCH}	Clock Rise Time		10	ns
T_{CHCL}	Clock Fall Time		10	ns
T_{CLSV}	Clock Low to Select Valid		10	ns
T_{CLOV}	Clock Low to Data Valid		10	ns

Note: 1. 32-bit format with Fs= 48 KHz.

Waveforms

Figure 90. Audio Interface Waveforms



External Clock Drive and Logic Level References

Definition of symbols

Table 67. External Clock Timing Symbol Definitions

Signals	
C	Clock

Conditions	
H	High
L	Low
X	No Longer Valid

Timings

Table 68. External Clock AC Timings

V_{DD} = 2.7 to 3.3 V, T_A = -40 to +85°C

Symbol	Parameter	Min	Max	Unit
T_{CLCL}	Clock Period	50		ns
T_{CHCX}	High Time	10		ns
T_{CLCX}	Low Time	10		ns
T_{CLCH}	Rise Time	3		ns
T_{CHCL}	Fall Time	3		ns
T_{CR}	Cyclic Ratio in X2 mode	40	60	%

Waveforms

Figure 91. External Clock Waveform

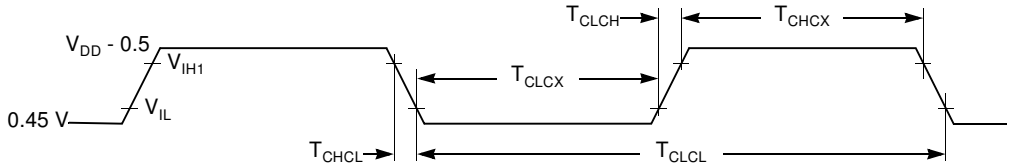
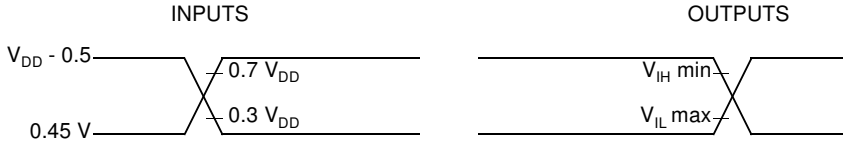


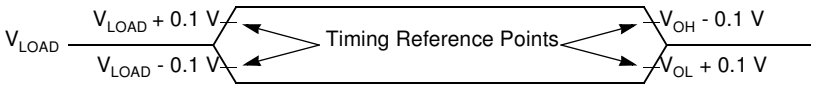
Figure 92. AC Testing Input/Output Waveforms



Note: 1. During AC testing, all inputs are driven at V_{DD} -0.5 V for a logic 1 and 0.45 V for a logic 0.

2. Timing measurements are made on all outputs at V_{IH} min for a logic 1 and V_{IL} max for a logic 0.

Figure 93. Float Waveforms



Note: For timing purposes, a port pin is no longer floating when a 100 mV change from load voltage occurs and begins to float when a 100 mV change from the loading V_{OH}/V_{OL} level occurs with $I_{OL}/I_{OH}= \pm 20$ mA.

Ordering Information

Table Possible order entries

Part Number	Supply Voltage	Temperature Range	Max Frequency	Package	Packing	Product Marking	RoHS Compliant	Firmware Version
AT83SND2MP3A1-7FTUL	3V	Industrial & Green	40 MHz	BGA100	Tray	83C51SND2CMP 3A1-ULA	Yes	2.40

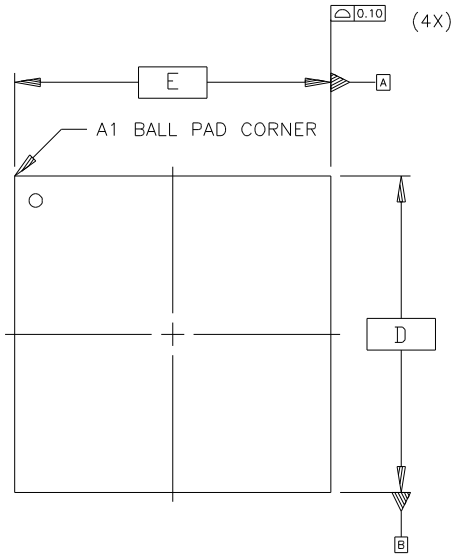
Table Obsolete part numbers

Part Number	Supply Voltage	Temperature Range	Max Frequency	Package	Packing	Product Marking	RoHS Compliant	Firmware Version
AT83SND2MP3-7FTIL	3V	Industrial	40 MHz	BGA100	Tray	83C51SND2C-IL	No	2.00
AT83SND2MP3-7FTJL	3V	Industrial & ROHS	40 MHz	BGA100	Tray	83C51SND2C-JL	Yes	2.00
AT83SND2CDVX-7FTIL	3V	Industrial	40 MHz	BGA100	Tray	83C51SND2C-IL	No	2.07
AT83SND2CDVX-7FTJL	3V	Industrial & ROHS	40 MHz	BGA100	Tray	83C51SND2C-JL	Yes	2.07

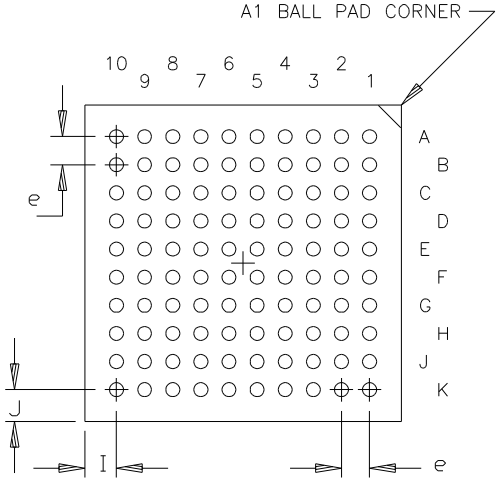
Package Information

CTBGA100

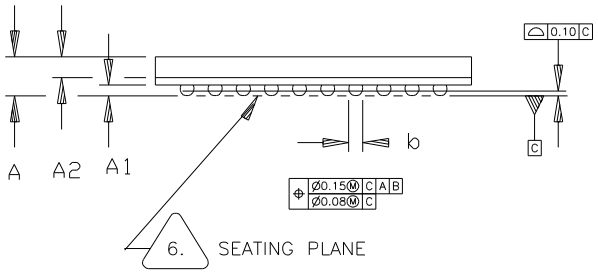
100 CTBGA/ TFBGA



TOP VIEW



BOTTOM VIEW



SIDE VIEW

	mm		
	MIN	NOM	MAX
A			1.20
A1	0.25	0.30	0.35
A2	0.55	0.60	0.65
e	0.80 BSC		
E/D	8.95	9.00	9.05
I/J	0.90 REF		
b	0.25		0.45

NOTES: CTBGA PACKAGE FAMILY

4. REFERENCE SPECIFICATIONS:
   A. AAWW SPEC #001−0531−2234: PACKING OPERATION PROCEDURE.
   B. AAWW SPEC #001−0519−2062: MARKING.

3. PRIMARY DATUM C AND SEATING PLANE ARE DEFINED BY THE SPHERICAL CROWNS OF THE SOLDER BALLS.

2. DIMENSION IS MEASURED AT THE MAXIMUM SOLDER BALL DIAMETER, PARALLEL TO PRIMARY DATUM C.

1. ALL DIMENSIONS AND TOLERANCES CONFORM TO ASME Y14.5M−1994.

Document Revision History

Changes from 7524A-07/05 to 7524B-05/06

1. Added AT83SND2CDVX part number.

Changes from 7524B-05/06 to 7524C - 06/07

1. Added AT83SND2CMP3A1 part number.

Changes from7524C - 06/07 to 7524D - 07/07

1. Updated Package drawing, CTBGA100.

Table of Contents

Atmel Corporation

2325 Orchard Parkway
San Jose, CA 95131, USA
Tel: 1(408) 441-0311
Fax: 1(408) 487-2600

Regional Headquarters

Europe
Atmel Sarl
Route des Arsenaux 41
Case Postale 80
CH-1705 Fribourg
Switzerland
Tel: (41) 26-426-5555
Fax: (41) 26-426-5500

Asia
Room 1219
Chinachem Golden Plaza
77 Mody Road Tsimshatsui
East Kowloon
Hong Kong
Tel: (852) 2721-9778
Fax: (852) 2722-1369

Japan
9F, Tonetsu Shinkawa Bldg.
1-24-8 Shinkawa
Chuo-ku, Tokyo 104-0033
Japan
Tel: (81) 3-3523-3551
Fax: (81) 3-3523-7581

Atmel Operations

Memory
2325 Orchard Parkway
San Jose, CA 95131, USA
Tel: 1(408) 441-0311
Fax: 1(408) 436-4314

Microcontrollers
2325 Orchard Parkway
San Jose, CA 95131, USA
Tel: 1(408) 441-0311
Fax: 1(408) 436-4314

La Chantrerie
BP 70602
44306 Nantes Cedex 3, France
Tel: (33) 2-40-18-18-18
Fax: (33) 2-40-18-19-60

ASIC/ASSP/Smart Cards
Zone Industrielle
13106 Rousset Cedex, France
Tel: (33) 4-42-53-60-00
Fax: (33) 4-42-53-60-01

1150 East Cheyenne Mtn. Blvd.
Colorado Springs, CO 80906, USA
Tel: 1(719) 576-3300
Fax: 1(719) 540-1759

Scottish Enterprise Technology Park
Maxwell Building
East Kilbride G75 0QR, Scotland
Tel: (44) 1355-803-000
Fax: (44) 1355-242-743

RF/Automotive
Theresienstrasse 2
Postfach 3535
74025 Heilbronn, Germany
Tel: (49) 71-31-67-0
Fax: (49) 71-31-67-2340

1150 East Cheyenne Mtn. Blvd.
Colorado Springs, CO 80906, USA
Tel: 1(719) 576-3300
Fax: 1(719) 540-1759

Biometrics/Imaging/Hi-Rel MPU/
High Speed Converters/RF Datacom
Avenue de Rochepleine
BP 123
38521 Saint-Egreve Cedex, France
Tel: (33) 4-76-58-30-00
Fax: (33) 4-76-58-34-80

Literature Requests
www.atmel.com/literature

Printed on recycled paper.

7524D–MP3–07/07