

PCM Audio Recorder v1.0 –Documentation

1. Overview

1.1. Introduction

The MASF PCM-Audio-Recorder Download-Software provides the MASF with the ability to record and play back non-compressed PCM audio data.

The integrated A/D-converter, the I2S input interface (SDI) or S/PDIF input interface can be used for continuous audio sample input. The integrated D/A-converter, the I2S output interface (SDO) and the S/P-DIF output interface are used for audio output.

In encoding mode, the recorded samples are enhanced with synchronization framing information and sent to the memory device via the parallel interface port (PIO). The incoming audio data is also monitored to the audio output interfaces.

In decoding mode, the frames of PCM audio samples are read from the memory device via the parallel interface port (PIO) or the SDIB interface and played back on the output interfaces.

A schematic overview of the recording signal flow is given on Figure 1.

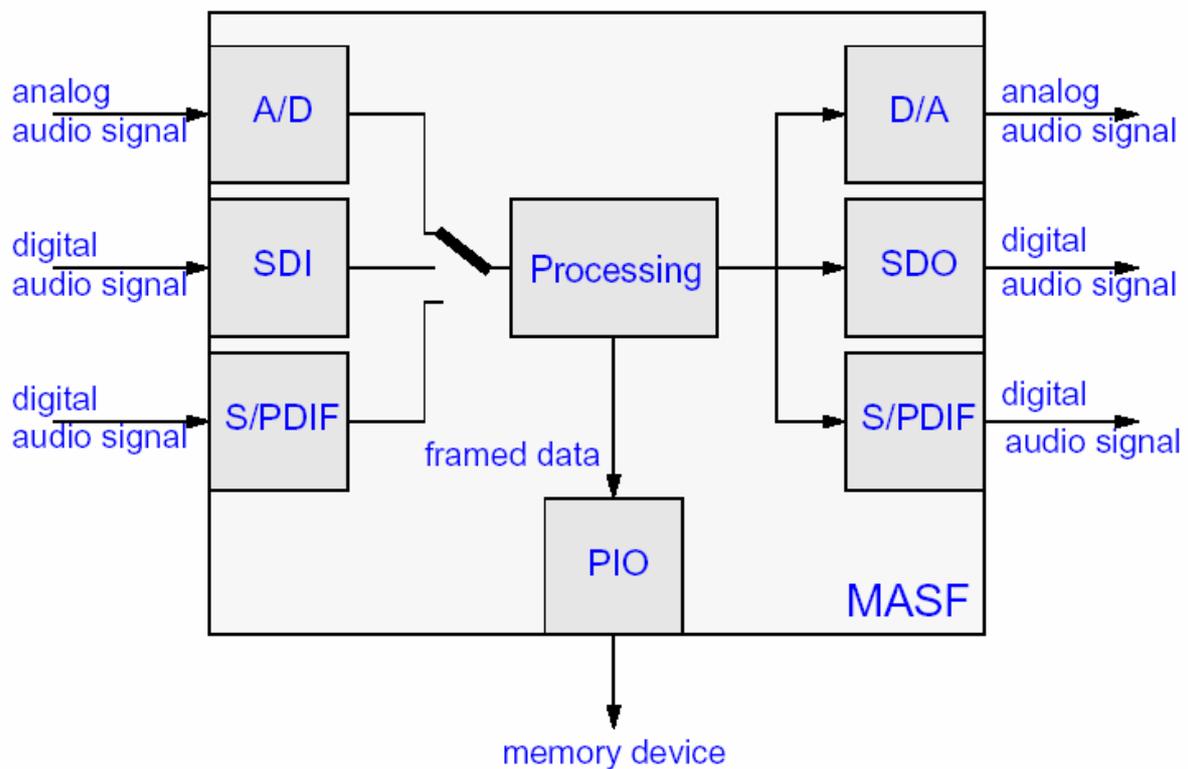


Figure 1: Recording signal flow

A schematic overview of the playback signal flow is given on Figure 2.

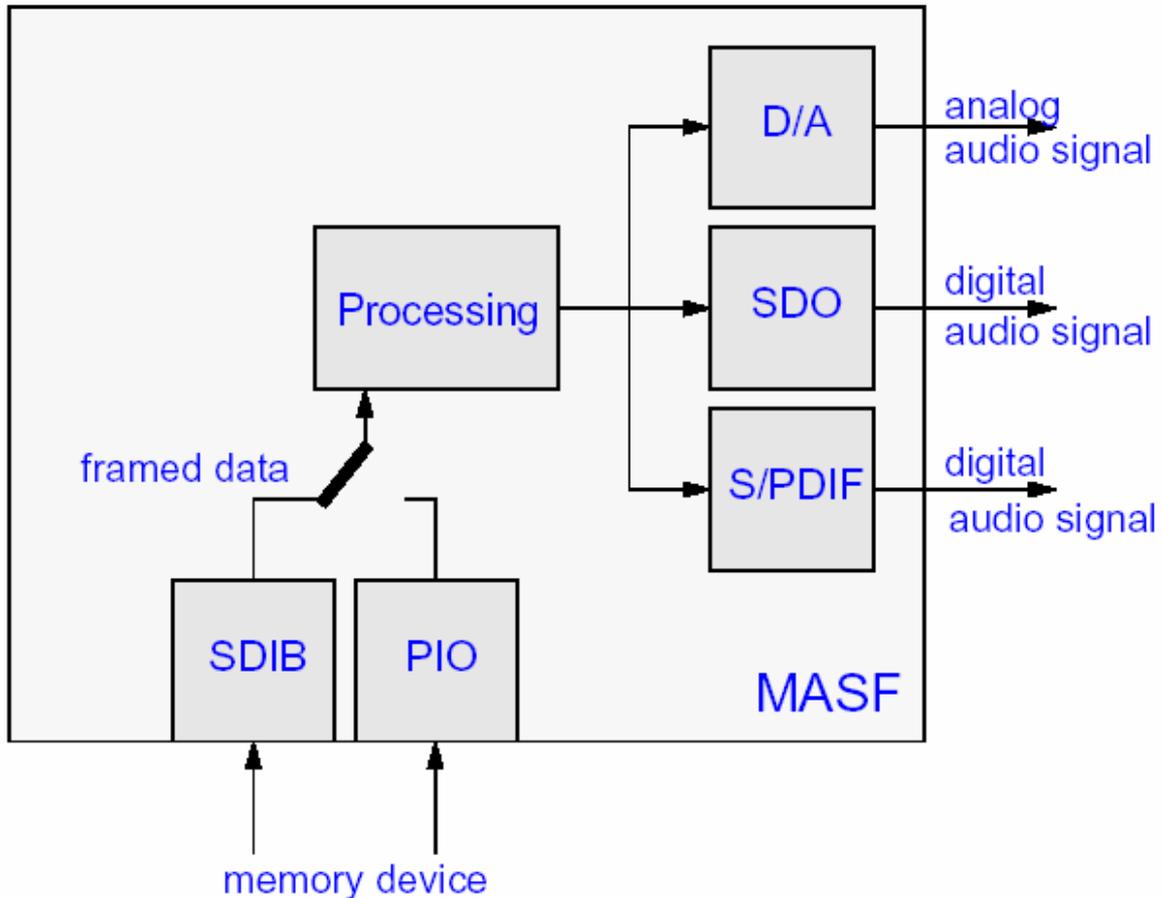


Figure 2: Playback signal flow

1.2. Features

- Download software for MASF ICs
- Recording of PCM audio samples to a memory device
- Input from integrated A/D converter, I2S or S/PDIF interface
- Playback of PCM audio data from a memory device
- Output to integrated D/A converter, synchronous I2S interface and S/P-DIF
- Parallel data transfer to/from memory device
- Serial data transfer for playback from memory device
- Optional self-synchronizing framing format
- Sample rates of 8, 11.025, 12, 16, 22.05, 24, 32, 44.1, 48 kHz are supported
- Recording and playback of stereo and mono signals

2. Software usage

2.1. Requirements

The PCM Audio Recorder software is implemented as downloadable software for the MAS 3587 F digital signal processors. The processor must have all the interfaces used by the software.

In order to use the software an appropriate hardware system should be set up. The system at least contains a MAS 3587 F, an appropriate power supply and a device (micro controller) that controls MAS using the I2C interface.

2.2. Framing format

For playback via serial data input interface, framing information for synchronization is mandatory. For parallel recording/playback, the insertion/checking of framing information can be disabled.

The output format of sample data is switchable between big endian and little endian format. This does not affect the synchronization information.

The frame format is independent of sample rate and channel mode. Those values have to be supplied correctly for recording and playback.

If framing is switched on, each frame starts with the WAVE header (4 bytes, ASCII “WAVE”). Followed by 1024 PCM encoded audio samples of 16-bit each. In stereo mode, a frame contains 512 sample pairs. In mono mode, a frame contains 1024 mono samples. The format of a data frame is shown in Figure 3.

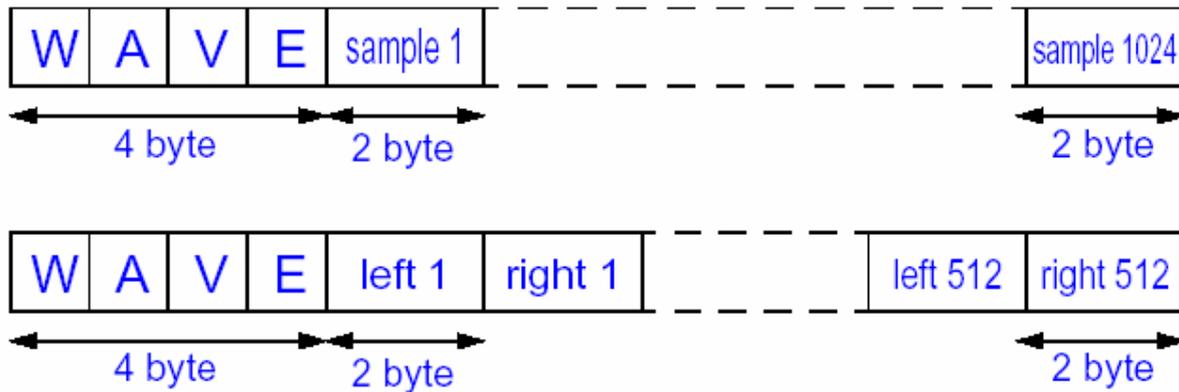


Figure 3: Frame format for mono and stereo channel mode

If framing is switched off, the data stream just consists of a continuous sequence of 16 bit samples.

2.3. User interface

The communication between the PCM Audio Recorder software and the device controlling it is implemented as a set of control and status memory locations accessible through I2C interface.

After start the software waits in the main (idle) loop (see Figure 1.) until the user sets the desired configuration and sets the validate bit (MainIOControl memory cell, bit 0) to one confirming that the configuration of the software is finished. When the software detects the validate bit, it interprets the control memory cell contents. If an operating mode is selected (PLAY or RECORD, MainIOControl bits [13:12]) the software moves to the appropriate operation mode.

Address	Control Memory Cell
D0:\$0661	MainIOControl
D0:\$0662	InterfaceStatusControl
D0:\$0663	OscillatorFrequency
D0:\$0664	OutputClockConfiguration
D0:\$0665	FrameCounter
D0:\$0666	ApplicationRunning
D0:\$0667	SampleRateStatus
D0:\$0668	BufferErrorCounter
D0:\$0669	SoftMute
D0:\$066a	SPDIFChannelStatusBits
D0:\$066b	SampleRate
D0:\$066c	OutputVolumeLL
D0:\$066d	OutputVolumeLR
D0:\$066e	OutputVolumeRL
D0:\$066f	OutputVolumeRR
D0:\$0670	InputVolumeLL
D0:\$0671	InputVolumeLR
D0:\$0672	InputVolumeRL
D0:\$0673	InputVolumeRR

Table 1: User interface memory cells**2.3.1. Control memory cells**

The configuration of the PCM-Audio-Recorder is done via the control memory cells described in the following sections.

2.3.1.1 Main I/O Control

IOControlMain is used for selecting/deselecting the appropriate data input interface and for setting up the serial data output interface.

Bit [10:9] switch between idle, recording and playback mode.

Bit [0] is the so called validate bit, which has to be set by the controller to validate all control memory cell changes.

2.3.1.2 Sample Rate

SampleRate is used to select the desired sample rate and mono or stereo channel mode. As no sample rate information is stored inside the packed frames, the sample rate and channel mode must be supplied on recording and playback. The only exception is recording from S/PDIF input. In this case, the sampling frequency is auto-detected and stored in a status memory cell.

Bit [5] selects the endian format of the PCM data. If it is set to one, the two bytes of each sample are swapped.

Bit [6] is used to switch between transmission with included synchronization header information and raw sample transfer.

2.3.1.3 Interface Status Control

InterfaceControl allows to enable/disable the data I/O interfaces. In addition, the clock of the output data interface interfaces, SPDIF and SDO, can be set to weak mode.

Bit [0] switches between internal A/D converter and external SDI pins.

2.3.1.4 Oscillator Frequency

OfreqControl contains the oscillator frequency in kHz. Its default value is 18432 kHz.

2.3.1.5 Output Clock Configuration

The CLKO output pin of the MASF can be enabled/disabled via bit [19] of OutClkConfig. The pin provides an audio over-sampling clock. It is calculated as sampling frequency x over-sampling factor (e.g. 48000 x 512 = 24576000).

The possible frequencies are given in the following table.

fs/kHz	Output Frequency at CLKO/MHz				
	Synth. Clock bit [8]=1	Scaler On bit [8]=0, bit [17]=0		Scaler Plus Extra Division bit [8]=0, bit [17]=1	
48	24.576	512·fs	24.576	256·fs	12.288
44.1	22.5792		22.5792		11.2896
32	24.576	768·fs	24.576	384·fs	12.288
24			12.288		6.144
22.05	22.5792		11.2896		5.6448
16	24.576	768·fs	12.288	384·fs	6.144
12			6.144		256·fs
11.025	22.5792		5.6448		2.8224
8	24.576	768·fs	6.144	384·fs	3.072

Table 2: Settings of bit[8] and bit[17] in OutClkConfig and resulting CLKO output frequencies

2.3.1.6 Soft Mute

By setting bit [0], the audio output is muted for both recording and playback mode. The mute is done in a linear manner.

In recording mode setting bit [1] allows pausing the recording without breaking the synchronization header sequence. If this bit is enabled, the incoming audio data is just not written to the output buffer.

In playback mode setting bit [1] pauses the playback. While in pause mode, the data is not received through input interfaces.

2.3.1.7 S/PDIF Channel Status Bits

These bits are used to signal the device type over the S/PDIF lines. It covers the first 16 bits of channel status information. The bits indicating the sampling rate (bits [24-27]) are inserted automatically according to the following table.

fs/kHz	SPDIF Channel Status Bits indicating sample rate bits [24-27]
48	0 1 0 0
44.1	0 0 0 0
32	1 1 0 0
24	0 1 0 0
22.05	0 0 0 0
16	1 1 0 0
12	0 1 0 0
11.025	0 0 0 0
8	1 1 0 0

Table 3: SPDIF Channel Status Bits - sample rate bits

Below 32kHz, the bits indicating the sample rate are actually the same as the bits of the double or quadruple sample rate.

2.3.1.8 Volume Control

The digital Baseband Volume Matrix is used for controlling the digital gain.

Different Volume Matrices are used for recording and playback.

The fixed-point gain values correspond to 20 bit 2's complement notation.

Memory address (hex)	Function	Name
D0:\$0661	<p>Main I/O Control (reset = 0x125hex)</p> <p>IOControlMain is used for selecting/deselecting the appropriate data input interface and for setting up the serial data output interface. In serial input mode the packed audio data is expected at the serial input interface SDIB. In the 8-bit-parallel input mode (default) the PIO pins PI[19:12] are used.</p> <p>bit [14] Invert serial output clock (SOC) 0 (reset) do not invert SOC 1 invert SOC</p> <p>bit [13:12] Recording/Playback selection 0 (reset) idle mode 1 audio playback 2 audio recording 3 reserved</p> <p>bit [11] Serial data output delay 0 (reset) no additional delay</p>	MainIOControl

Memory address (hex)	Function	Name
	1 additional delay of data related to word strobe reserved bit [10] reserved bit [9:8] Recording: Audio data input select 00 audio input at SDI with PLL 01 (reset) audio input at SDI/internal ADC without PLL 10 S/PDIF input 11 reserved Playback: Data input select 00 serial input at interface B 01 (reset) parallel input at PIO pins[19...12] 10 reserved 11 reserved bit [7] Recording: Invert serial input clock (SIC) 0 (reset) do not invert SIC 1 invert SIC bit [6] Recording: Serial data input delay 0 (reset) no additional delay 1 additional delay of data related to wordstrobe bit [5] SDO Word Strobe Invert 0 do not invert 1 (reset) invert outgoing word strobe signal bit [4] Bits per Sample at SDO 0 (reset) 32 bits/sample 1 16 bits/sample bit [3] reserved bit [2] Playback: Serial data input interface B clock invert (pin SIBC) 0 not inverted (data latch at rising clock edge) 1 (reset) incoming clock signal is inverted (data latch falling clock edge) Recording: SDI word strobe invert 0 do not invert 1 (reset) invert incoming word strobe signal bit [1] reserved bit [0] Validate 0 (reset) 1 changes in control memory will become effective	
D0:\$0662	Interface Status Control (reset = 05hex) This control cell allows to enable/disable the data I/O interfaces. In addition, the clock of the output data interfaces, S/PDIF and SDO, can be set to a low impedance mode. bit [6] reserved bit [5] enable/disable SPDIF output 0 (reset) enable S/PDIF output 1 S/PDIF output invalid bit [4] reserved bit [3] enable/disable serial data output SDO 0 SDO valid data 1 (reset) SDO invalid data bit [2] Output clock characteristics (SDO and S/PDIF output) 0 low impedance 1 (reset) high impedance bit [1] reserved bit [0] Enable/disable external serial data input SDI 0 use external audio source(SDI) 1 (reset) use internal A/D converter as audio	Interface Status Control

Memory address (hex)	Function	Name
	<p style="text-align: center;">source</p> <p>Note: The pins SIC, SII, SID are switched to output mode if bit [0]=1</p>	
D0:\$0663	<p>Oscillator Frequency (reset = 18432dec) bit [19..0] Oscillator frequency in kHz In order to achieve a correct internal operating frequency of the DSP, the nominal crystal frequency has to be written into this memory cell.</p>	Oscillator Frequency
D0:\$0664	<p>Output Clock Configuration (pin CLKO) (reset = 8000hex) bit [19] CLKO configuration 0 output clock signal at CLKO 1 (reset) CLKO is tri-state bit [18] reserved bit [17] Additional division by 2 if scaler is on (bit[8] cleared) 0 (reset) oversampling factor 512/768 1 oversampling factor 256/384 bit [16:9] reserved bit [8] Output clock scaler 0 (reset) set output clock according to audio sample rate (see Table 2) 1 output clock fixed at 24.576 or 22.5792 MHz bit [7..0] reserved</p>	Output Clock Configuration
D0:\$0669	<p>Soft Mute (reset = 0hex) bit [1] Recording: Pause mode 0 (reset) normal recording operation 1 recording processed is paused bit [0] Mute audio output 0 (reset) no mute of audio output 1 audio output is muted The mute is done in a linear manner within 60ms. If the software is paused, the mute is done immediately. The same procedure is performed for de-mute.</p>	Soft Mute
D0:\$066b	<p>Sample Rate (reset = 0hex) This control cell allows to set the sample rate that is used for recording and playback, switch between stereo or mono mode, select big or little endian format and specify if header information shall be included into the bit-stream. In case of S/PDIF input, the sample rate is auto-detected. bit [6] Header transmission 0 (reset) enable transmission/checking of framing information 1 disable transmission/checking of framing information bit [5] Endian format 0 (reset) big endian 1 little endian bit [4] Channel mode 0 (reset) stereo 1 mono bit [3:0] Sample Rate 0000 8 kHz (reset) 0001 11.025 kHz 0010 12 kHz 0011 reserved 0100 16 kHz 0101 12.05 kHz</p>	Sample Rate

Memory address (hex)	Function	Name
	0110 24 kHz 0111 reserved 1000 32 kHz 1001 44.1 kHz 1010 48 kHz 1011 reserved 1100 reserved 1101 reserved 1110 reserved 1111 reserved	
D0:\$066a	S/PDIF Channel Status Bits Category Code (reset = 8200hex) These bits are used to signal the device type over the S/PDIF lines. It covers the first 16 bits of channel status information. The bits indicating the sampling rate (bits [24-27]) are inserted automatically.	SPDIF Channel StatusBits
D0:\$066c	Volume Input Control Left -> Left Gain (reset = 80000hex)	OutputVolume LL
D0:\$066d	Volume Input Control Left -> Right Gain (reset = 0hex)	OutputVolume LR
D0:\$066e	Volume Input Control Right -> Left Gain (reset = 0hex)	OutputVolume RL
D0:\$066f	Volume Input Control Right -> Right Gain (reset = 80000hex)	OutputVolume RR
D0:\$0670	Volume Output Control Left -> Left Gain (reset = 80000hex)	InputVolume LL
D0:\$0671	Volume Output Control Left -> Right Gain (reset = 0hex)	InputVolume LR
D0:\$0672	Volume Output Control Right -> Left Gain (reset = 0hex)	InputVolume RL
D0:\$0673	Volume Output Control Right -> Right Gain (reset = 80000hex)	InputVolume RR

Table 4: Description of control memory cells

2.3.2. Status memory cells

The status memory cells reflect the actual state of the recording/playback process and its environment.

2.3.2.1 Frame Counter

The counter will be incremented with each new frame that is processed. In recording mode it counts the frames sent to memory device, while in playback mode it contains the number of frames played back.

With an invalid bit stream at its input (e.g. an invalid header is detected) or timeout of data, the frame counter is set to '0'.

2.3.2.2 Application Running

The value of this memory location indicates the operation mode of the software.

2.3.2.3 Sample Rate Status

The value of this memory location indicates the sample rate used for recording or playback. In recording mode, when S/PDIF input is used the memory location contains the auto-detected sample rate.

2.3.2.4 Buffer Error Counter

In recording mode, the counter will be incremented each time a stereo sample pair (or a mono sample) can't be written to the output buffer, as it is already full.

During playback, the counter contains the number of synchronization errors.

It is reset to '0' at the beginning of the processing.

Memory address (hex)	Function	Name
D0:\$0665	Frame Counter bit [19:0] number of processed frames	Frame Counter
D0:\$0666	Application Running bit [19:0] 80001 _{hex} : playback 80002 _{hex} : recording 80003 _{hex} : recording, S/PDIF auto-detection	Application Running
D0:\$0667	Sample Rate Status In case of S/PDIF input, the detected input sample frequency is written to this cell. For SDI input, the frequency is just copied from the Sample Rate control cell. bit [3:0] Detected sample rate 0000 8 kHz (reset) 0001 11.025 kHz 0010 12 kHz 0011 reserved 0100 16 kHz 0101 12.05 kHz 0110 24 kHz 0111 reserved 1000 32 kHz 1001 44.1 kHz 1010 48 kHz 1011 reserved 1100 reserved 1101 reserved 1110 reserved 1111 reserved	Sample Rate Status
D0:\$0668	Buffer Error Counter bit [19:0] number of lost sample pairs on encoding	Buffer Error Counter

Table 5: Description of status memory cells

2.4. Software setup

In order to use the software, it has to be downloaded to the MAS. After download the desired operation configuration should be chosen, and activated by setting the validate bit to one.

Since the software uses the A/D and D/A converters, there must be enabled before starting the processing.

2.4.1. Downloading the software

The code is distributed to 4 blocks of 1kW. The download procedure consists of following steps:

- Freezing MAS using command "run \$0".
- Switching the configurable memory blocks to data address space at address by writing \$00000 to the P_SELECT register.
- Downloading the code to the memory D0 at address \$800 (2kW) and to the memory D1 at address \$800 (2kW).
- Switching the configurable memory blocks to program address space at address \$1000 by writing \$f0000 to the P_SELECT register.

After these steps, the software is ready to run. The program execution starts from address \$1000.

2.4.2. Commands

As shown on Figure 4, the software has three states: IDLE, PLAYING and RECORDING.

The main commands are: PLAY, RECORD, STOP, MUTE and PAUSE. The commands are described in the following table:

Command	Description
PLAY	OperationMode = PLAY MainIOControl bits [13:12] = 1 Validate = 1 MainIOControl bit [0] = 1
RECORD	OperationMode = RECORD MainIOControl bits [13:12] = 2 Validate = 1 MainIOControl bit [0] = 1
STOP	OperationMode = IDLE MainIOControl bits [13:12] = 0 Validate = 1 MainIOControl bit [0] = 1
MUTE	SoftMute = MUTE SoftMute bit [0] = 1 Validate = 1 MainIOControl bit [0] = 1
PAUSE	SoftMute = PAUSE SoftMute bit [1] = 1 Validate = 1 MainIOControl bit [0] = 1

Table 6: User commands

The following table shows which commands are regarded in different states.

State Command	IDLE	PLAY	RECORD
PLAY	After the command immediately moves to the PLAY state.		
RECORD	After the command immediately moves to the RECORD state.		
STOP		Immediately stops receiving through input interfaces and stops playing. Pending DMA operations are interrupted (both PIO and SIOB). Returns to IDLE state.	Waits until a frame is completed, then returns to idle state. If the input is late while completing the frame it is filled with 0.
MUTE/ UNMUTE		Mutes within a given number of 32 word packets. The mute is performed on linear base. If (de-) mute occurs during pause, the volume is set to 0 (original value).	Mutes within a given number of 32 word packets. The mute is performed on linear base (soft mute).
PAUSE		Stops sending request for data; the output is cleared. Pending requests are finished.	Stops sending data to memory device. The current frame is not completed. The audio output is not affected.
UNPAUSE		The pause state is left when the difference between the input write pointer and the output read pointer indicates that the input buffer is full. This mechanism avoids buffer underflow immediately after pause.	Since the pause command affects only the transmission to the memory device, after the un-pause command the pause state is left immediately.

2.4.3. Setting up the software for playback mode

The first step in starting the software is to configure the mode that will be used. In case of PLAY mode, the following steps should be performed:

1. Enable the D/A converter using the codec configuration registers.
2. In the MainIOControl control memory cell choose the appropriate input interface (PIO or serial). Ensure that a data source (memory device) is connected to the interface chosen.
3. In the control memory cell SampleRate set the appropriate sample rate.
4. Set the number of channels in the SampleRate control cell. In mono mode, the both output channels are same.
5. Set the byte ordering of data received through input interfaces in the SampleRate control cell.
6. If the data being received contains framing information, in the SampleRate control cell turn framing information checking on. If the input is serial, the data stream should always contain framing information.
7. The output serial and S/PDIF interfaces may be turned on/off using the InterfaceStatusControl memory cell. The output interfaces can be also configured to operate in weak mode.
8. Using the OutputClockConfiguration control memory cell, the required OCLK configuration may be chosen.
9. By setting 1 in MainIOControl bits [13:12] choose the playing operation mode.
10. Set the Validate bit (MainIOControl bit[0]) to 1, thus indicating that the software configuration is finished and the content of the control memory cells is valid.
11. When the software detects that the validate bit is high, it moves to the PLAY mode, and updates the status memory cells.
12. During playing the output can be paused or muted using "Soft Mute Commands". In case of pause, the data reception is paused, but in case of mute, the data is received but not transmitted to the audio output. The playing can be stopped using the STOP button.

2.4.4. Setting up the software for recording mode

The first step in starting the software is to configure the mode that will be used. In case of RECORD mode, the following values should be set:

1. Enable the A/D and D/A converters using the codec configuration register.
2. In the MainIOControl control memory cell choose the appropriate input interface (serial with software PLL, integrated A/D without software PLL or S/PDIF with software PLL).
3. In the control memory cell SampleRate set the appropriate sample rate. If the input is S/PDIF, the sample rate is auto-detected.
4. Set the number of channels in the SampleRate control cell. In mono mode, only the left input channel is processed and both output channels are same.
5. Set the byte ordering of data received through input interfaces in the SampleRate control cell.
6. If the data being sent should contain framing information, in the SampleRate control cell turn framing information insertion on.
7. The output serial and S/PDIF interfaces may be turned on/off using the InterfaceStatusControl memory cell. The output interfaces can be also configured to operate in weak mode.
8. Using the OutputClockConfiguration control memory cell, the required OCLK configuration may be chosen.
9. By setting 2 in MainIOControl bits [13:12] choose the playing operation mode.
10. Set the Validate bit (MainIOControl bit[0]) to 1, thus indicating that the software configuration is finished and the content of the control memory cells is valid.
11. When the software detects that the validate bit is high, it moves to the PLAY mode, and updates the status memory cells.
12. During processing the output can be muted using the Mute command. In case of pause, the data transmission through PIO is paused. The recording may be stopped using the STOP button.

2.5. Error handling and error reporting

The software contains built-in error handling routines for the situations described in the table below. The cause of error is indicated by the content of the LastErrorCode register (register G1P31).

Event	State	PLAY	RECORD

Event	State	PLAY	RECORD
Synchronization error		<p>CAUSE: In case when framing is on but the input data stream does not contain valid synchronization information, synchronization error occurs.</p> <p>ACTION: The Frame Counter is reset; the Buffer Error Counter is incremented.</p> <p>LastErrorCode = \$01</p>	
Input Buffer Overflow			<p>CAUSE: The output PIO interface is late; the input buffer pointer overwrites the data not yet transmitted.</p> <p>ACTION: The buffer error counter is incremented by the value of the dropped samples. The frame counter is reset.</p> <p>LastErrorCode = \$11</p>
Output Buffer Underflow		<p>CAUSE: The input interface does not provide enough data.</p> <p>ACTION: The output buffer is filled with zeroes. The frame counter is reset.</p> <p>LastErrorCode = \$01</p>	<p>CAUSE: The input interface does not provide enough data.</p> <p>ACTION: The output buffer is filled with zeroes.</p> <p>The input buffer read and write pointers are set to the beginning of the input buffer. The PLL is disabled until the input starts again. If this happens while completing the frame after stop, the frame is completed with zeroes.</p> <p>The frame counter is reset.</p> <p>LastErrorCode = \$12</p>
S/PDIF Sync Error			<p>CAUSE: The input interface cannot synchronize to the input bit stream.</p> <p>ACTION: The output buffer is filled with zeroes.</p> <p>The input buffer read and write pointers are set to the beginning of the input buffer. The PLL is disabled until the input starts again and the synchronization bit is stable at 1 for at least 4ms.</p> <p>If this happens while completing the frame after stop, the frame is completed with zeroes.</p> <p>The frame counter is reset.</p> <p>LastErrorCode = \$12</p>

Table 7: Error handling procedures

Error Code	Value	Description
PLAY_SyncError	\$01	In case when framing is on but the input data stream does not contain valid synchronization information,

		synchronization error occurs.
PLAY_OutputBufferUnderflow	\$02	In case when the input interface (PIO or SDI(B)) does not receive data quick enough, there is nothing to play back.
RECORD_InputBufferOverflow	\$11	In case when the device attached to the output PIO interface does not receives data quick enough, some of the samples in the input buffer will be overwritten with new data before they are transmitted to the device.
RECORD_OutputBufferUnderflow	\$12	In case when the input interface (SDI or S/PDIF) does not receive data quick enough, there is nothing to play back.
RECORD_SyncLost	\$13	In case when the input is S/PDIF and the synchronization bit goes 0, the input is invalid.

Table 8: Error codes

3. Implementation details

3.1. Software structure

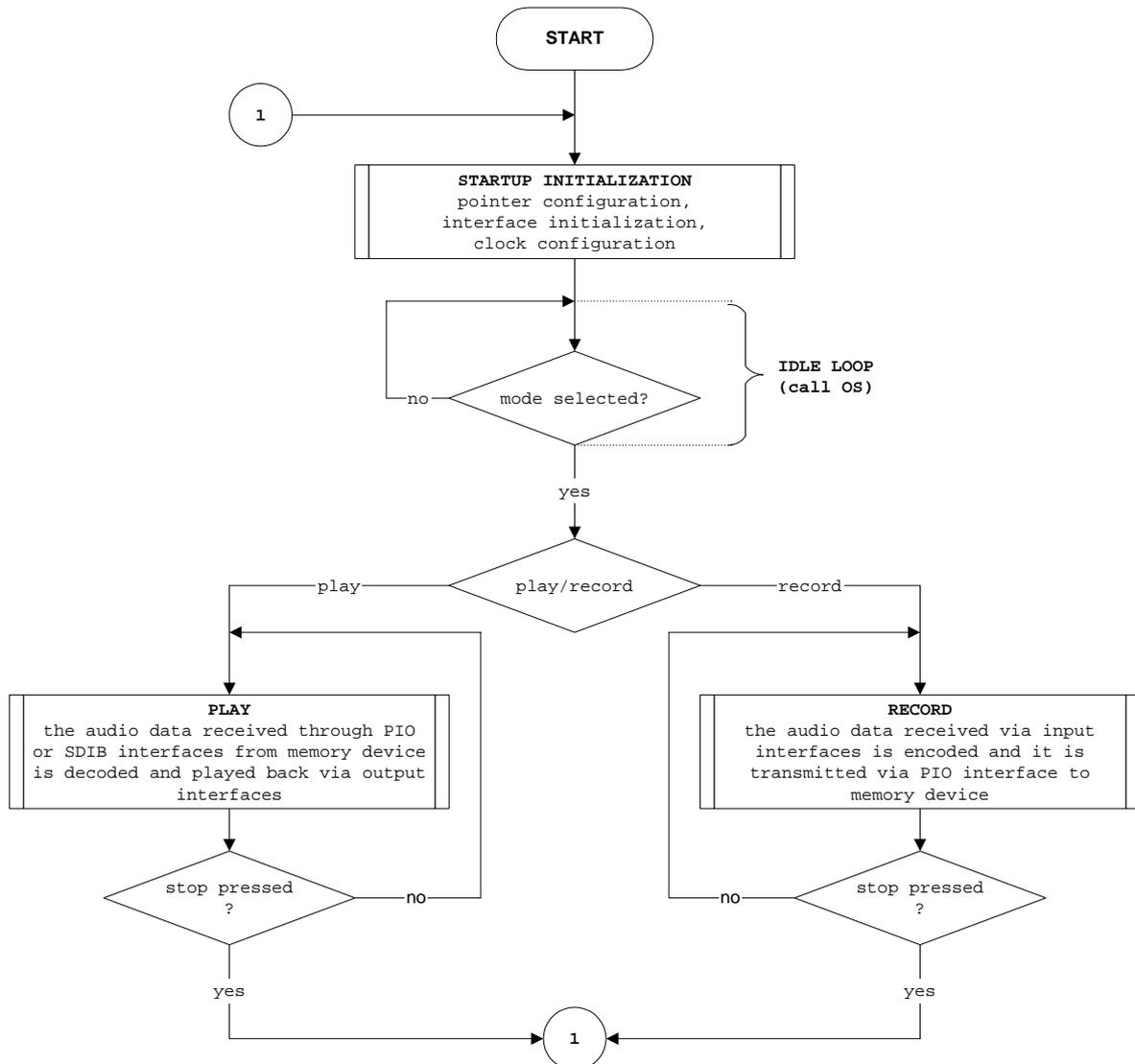


Figure 4: Software flow-chart

3.2. Interfaces

In both operation modes, the output interfaces are started after the output buffer is filled with audio samples. This mechanism introduces a delay of ~10ms between the input and output and at the same time ensures that there is enough time for the software to process data before the output interface run out of data.

Once the output interfaces started, they are never stopped until the operation mode is left. This behavior is required because the output S/PDIF bit stream conveys the channel status bits, which position is fixed within the bit-stream.

3.2.1. Interfaces in recording mode

When using the serial interface SDI as audio source, input and output audio interfaces are running synchronous to the MAS core. The data frames are bursted over the PIO interface. In this case, there is no software PLL necessary to synchronize to a different audio clock.

When using the S/PDIF input as audio source, the incoming sample frequency has to be auto detected. The MAS core and all audio output interfaces have to be synchronized with software PLL.

3.2.2. Audio input

The audio sample input can be switched between internal A/D converter, SDI pins and S/PDIF input. A volume matrix is applied to the incoming audio samples. The configuration of the SDI interface is specified in the user interface.

3.2.3. Audio output

The audio output in recording mode is available for monitoring the incoming audio samples. A volume matrix is applied before sending the audio samples to the internal D/A converter, the SDO pins and the S/P-DIF output interface. The interface configuration for the SDO interface is specified in the user interface.

3.2.4. Data output

The frames of packed audio data are sent via the PIO interface in PIO-DMA output mode in packets of 36 bytes each. If the output buffer is already full, no more samples are written to the buffer. In this case, the buffer error counter is incremented for each purged stereo sample.

3.3. Interfaces in playback mode

The output audio interfaces are running synchronous to the MAS core. The data frames are bursted over the PIO interface. There is no software PLL to synchronize to a different audio clock.

3.3.1. Audio output

A volume matrix is applied before sending the unpacked audio samples to the internal D/A converter, the SDO pins and the S/P-DIF output interface. The interface configuration for the SDO interface is specified in the user interface.

3.3.2. Data input

The input of audio data frames can be switched between parallel and serial mode. In parallel mode the frames are received via the PIO interface in PIO-DMA input mode in packets of 32 bytes each. In serial mode, the frames are received via the serial interface SDIB in continuous transmission mode with demand signaling. In case of serial input, the framing information must be present in the data received.

If the processor runs out of data, the output is muted until new data arrives. The frame counter is also reset in this case.

3.4. Processor resources

3.4.1. Processor clock

In order to handle data received/transmitted at different sample rates, the MAS processor operates at different clock. The clock values are given in the table below.

Sample rate	Processor clock
48 kHz	24.576 MHz
44.1 kHz	22.579 MHz
32 kHz	24.576 MHz
24 kHz	24.576 MHz
22.05 kHz	22.579 MHz
16 kHz	24.576 MHz
12 kHz	24.576 MHz
11.025 kHz	22.579 MHz
8 kHz	24.576 MHz

Table 9: MAS operating frequencies

3.4.2. Program memory

The memory of the MAS processor is divided to program and data memory. The following tables show the usage of both memories.

Module	Code	Start address	End address	Length
Common code		\$1000	\$1278	\$278 (632)
Play mode		\$1279	\$1669	\$3f0 (1008)
Record mode		\$1800	\$1bd9	\$3d9 (985)

Table 10: Code memory usage

Module	Memory	Memory D0	Memory D1
Input Buffer		\$0 - \$5ff	
Output Buffer			\$0 - \$5ff
Other Data		\$600 - \$708	\$600 - \$67b

Table 11: I/O buffer locations

3.5. Timings

3.5.1. Mute

After the user selects mute, the software start the muting procedure by linearly decrementing the gain of the output samples with each block being copied. After 60ms, the gain reaches 0. If the software is paused when mute is selected, the gain is set to 0 immediately.

The same applies for de-mute.

3.5.2. Micro-controller response time

In case of playing the input buffer is 1536 words long. This capacity is enough to store 16ms of data at highest sample rate.

$$t = \frac{1536}{48000 * 2} = 16ms$$

This size of the buffer ensures that the micro-controller response time may be up to 16ms.

This applies to recording mode also.

In both operation modes, the output is started when 10ms of data is received. This introduces a delay of 10ms between the input and output signal.

3.5.3. S/PDIF auto-detection time

In recording mode, if as input S/PDIF interface is chosen, the software tries to match the sample rate of the input. It starts from the lowest sample rate (8kHz). It sets the software configuration for it, waits 20ms, and check if the hardware PLL in MAS succeeded to match the input rate. If not, it tries with the next (higher) sample rate. If none of the sample rates is auto-detected, the auto-detection procedure is restarted.

Since there are 9 supported sample rates, and to try one sample rate takes 20ms, in worst case the auto-detection takes $9 * 20ms = 180ms$. In best case it takes 20ms.

3.5.4. Reaction time on input/output under/overflow

If the input interface stops receiving data, the output interfaces will run out of data. Since the input and output interfaces are continuously monitored, this situation is detected immediately, and the output buffer is filled with silence.

3.5.5. S/PDIF resynchronization period

In recording mode, when S/PDIF is used as input interface and the S/PDIF interface loses the synchronization, the software waits 4ms of stable synchronization bit before starts processing again.

3.6. Hardware dependency

The MAS chip must have all the interfaces the software uses. They are:

- integrated A/D converter
- serial input
- S/PDIF input
- serial input with demand signal
- PIO input/output interface
- serial output
- integrated D/A converter and
- S/PDIF output.

It must have 4kW of configurable memory and must support code downloading. The software requires 3kW of program memory and 4kW of data memory.

4. Known problems

- If MAS (on Caesar) is powered through the DC/DC converter (1.5V), for sampling frequencies 44.1kHz, 22.05kHz and 11.025kHz, during recording when the PIO output is recorded by MST a byte is occasionally skipped. This happens when the data transmitted contains a value with many zero in it followed by a value that contains many one. At that moment, the data is not stable enough time to be detected by MST. This behavior disappears when the MAS is powered by 3.3V and 5V.
- Since the MST card does not support S/PDIF data recording of sample rates lower than 16kHz, the S/PDIF recording is not verified for sample rates 8kHz, 11.025kHz and 12kHz. The next step is to provide an S/PDIF device with capability to accept lower data rates.
In case of sample frequencies 16kHz, 22.050kHz and 24kHz there were some problems when MST is used as data source. When two MAS F were connected, the MAS F cannot synchronize at sample rate 16kHz. The tests for higher sample rates passed.
- In case when the output serial interface is configured to send 32 bits per sample, at higher sample rates (44.1kHz and 48kHz, bit rate 2.8 and 3.0 Mbps respectively) the data saved by MST software differs from expected. Some samples are received shifted, and an additional sample (0) is inserted after the shifted one. The signal on the oscilloscope looks as expected and the interface configuration shadow registers contain correct values. During additional testing it is determined that the MST cannot handle those bit rates due to hardware limitations.

Contents:

1. OVERVIEW	1
1.1. INTRODUCTION	1
1.2. FEATURES	2
2. SOFTWARE USAGE	2
2.1. REQUIREMENTS	2
2.2. FRAMING FORMAT	3
2.3. USER INTERFACE	3
2.3.1. <i>Control memory cells</i>	4
2.3.2. <i>Status memory cells</i>	8
2.4. SOFTWARE SETUP	9
2.4.1. <i>Downloading the software</i>	9
2.4.2. <i>Commands</i>	9
2.4.3. <i>Setting up the software for playback mode</i>	11
2.4.4. <i>Setting up the software for recording mode</i>	11
2.5. ERROR HANDLING AND ERROR REPORTING	11
3. IMPLEMENTATION DETAILS	14
3.1. SOFTWARE STRUCTURE	14
3.2. INTERFACES	14
3.2.1. <i>Interfaces in recording mode</i>	14
3.2.2. <i>Audio input</i>	15
3.2.3. <i>Audio output</i>	15
3.2.4. <i>Data output</i>	15
3.3. INTERFACES IN PLAYBACK MODE	15
3.3.1. <i>Audio output</i>	15
3.3.2. <i>Data input</i>	15
3.4. PROCESSOR RESOURCES	15
3.4.1. <i>Processor clock</i>	15
3.4.2. <i>Program memory</i>	16
3.5. TIMINGS	16
3.5.1. <i>Mute</i>	16
3.5.2. <i>Micro-controller response time</i>	16
3.5.3. <i>S/PDIF auto-detection time</i>	16
3.5.4. <i>Reaction time on input/output under/overflow</i>	16
3.6. HARDWARE DEPENDENCY	16
4. KNOWN PROBLEMS	17