

MA-3 Sound Middleware Specification

Version 1.10.0

April 22, 2003

Yamaha Corporation

[Notes]

This document is the specification of MA-3 Sound Middleware as sample source code.
This explains the expected operation of Sound Middleware, but doesn't guarantee operation of sample Middleware.

Copyright to this document is the property of Yamaha Corporation.
Transfer or copying of this document in part or in whole requires the permission of Yamaha Corporation.
The contents of this document are subject to change without notice.



Copyright © 2001-2003 YAMAHA Corporation

All rights reserved

CONFIDENTIAL

Contents

1	Introduction	1
1.1	Module configuration	1
1.2	Function of each module	2
1.3	Functional specifications	3
1.3.1	Functional model.....	3
1.3.2	Timer and software interrupt.....	4
1.3.3	Built-in SRAM.....	4
1.3.4	Tone generation channel / FM Voice / WT Voice	4
1.3.5	Reproduction of Stream Audio	4
1.3.6	Synchronization of Motor / LED.....	4
1.3.7	Delayed (Sequencer) system	5
1.3.8	Real time system	5
1.3.9	Audio system.....	5
1.3.10	Simultaneous reproduction of systems	6
1.4	Relationship among principal functions	7
2	MA Sound Sequencer.....	8
2.1	Status transition	9
2.2	Function definition	10
2.2.1	MaSound_Initialize	10
2.2.2	MaSound_DeviceControl.....	10
2.2.3	MaSound_Create.....	11
2.2.4	MaSound_Delete.....	11
2.2.5	MaSound_Load	12
2.2.6	MaSound_Unload	12
2.2.7	MaSound_Open	14
2.2.8	MaSound_Close	14
2.2.9	MaSound_Control	15
2.2.10	MaSound_Standby.....	18
2.2.11	MaSound_Seek.....	18
2.2.12	MaSound_Start	19
2.2.13	MaSound_Stop	19
2.2.14	MaSound_Pause	20
2.2.15	MaSound_Restart	20
3	MA Stream Converter	21
3.1	Outline	21
3.2	Status transition	22
3.3	Stream converter processing procedure.....	23
3.3.1	MaSrmCnv_Initialize	23
3.3.2	MaSrmCnv_Load.....	23
3.3.3	MaSrmCnv_Open	23
3.3.4	MaSrmCnv_Control	23

3.3.5	MaSrmCnv_Standby	23
3.3.6	MaSrmCnv_Seek	23
3.3.7	MaSrmCnv_Start.....	23
3.3.8	MaSrmCnv_Stop.....	23
3.3.9	MaSrmCnv_Close	23
3.3.10	MaSrmCnv_Unload.....	24
3.3.11	MaSrmCnv_End	24
3.3.12	MaSrmCnv_Convert.....	24
3.4	Function definition	25
3.4.1	MaSrmCnv_Initialize	25
3.4.2	MaSrmCnv_End.....	25
3.4.3	MaSrmCnv_Load.....	26
3.4.4	MaSrmCnv_Unload	27
3.4.5	MaSrmCnv_Open	27
3.4.6	MaSrmCnv_Close	27
3.4.7	MaSrmCnv_Control.....	28
3.4.8	MaSrmCnv_GetLength	29
3.4.9	MaSrmCnv_GetPos.....	29
3.4.10	MaSrmCnv_SetVolume	29
3.4.11	MaSrmCnv_SetShortMessage	30
3.4.12	MaSrmCnv_SetLongMessage	30
3.4.13	MaSrmCnv_Standby.....	30
3.4.14	MaSrmCnv_Seek.....	31
3.4.15	MaSrmCnv_Start	31
3.4.16	MaSrmCnv_Stop	31
3.4.17	MaSrmCnv_Convert.....	32
4	MA Sound Driver	33
4.1	Tone generation model	35
4.2	Function definition	36
4.2.1	MaSndDrv_Initialize.....	36
4.2.2	MaSndDrv_End	36
4.2.3	MaSndDrv_Create.....	37
4.2.4	MaSndDrv_Free.....	37
4.2.5	MaSndDrv_DeviceControl.....	38
4.2.6	MaSndDrv_GetTimeError.....	39
4.2.7	MaSndDrv_GetPos	39
4.2.8	MaSndDrv_SetSpeed	39
4.2.9	MaSndDrv_SetKey	40
4.2.10	MaSndDrv_SetKeyControl.....	40
4.2.11	MaSndDrv_SetFmExtWave.....	40
4.2.12	MaSndDrv_SetVolume	41
4.2.13	MaSndDrv_SetVoice	41
4.2.14	MaSndDrv_GetVoice.....	41
4.2.15	MaSndDrv_SetStream	42

4.2.16	MaSndDrv_SetCommand.....	43
4.2.17	MaSndDrv_Nop	44
4.2.18	MaSndDrv_UserEvent.....	44
4.2.19	MaSndDrv_NoteOn.....	44
4.2.20	MaSndDrv_NoteOnMa2.....	45
4.2.21	MaSndDrv_NoteOnMa2Ex	45
4.2.22	MaSndDrv_NoteOff	46
4.2.23	MaSndDrv_NoteOffMa2	46
4.2.24	MaSndDrv_NoteOffMa2Ex.....	46
4.2.25	MaSndDrv_ProgramChange.....	47
4.2.26	MaSndDrv_ModulationDepth	47
4.2.27	MaSndDrv_ChannelVolume	48
4.2.28	MaSndDrv_Panpot	48
4.2.29	MaSndDrv_Expression.....	48
4.2.30	MaSndDrv_Hold1	49
4.2.31	MaSndDrv_AllSoundOff.....	49
4.2.32	MaSndDrv_ResetAllControllers.....	50
4.2.33	MaSndDrv_AllNoteOff	50
4.2.34	MaSndDrv_MonoModeOn.....	50
4.2.35	MaSndDrv_PolyModeOn	51
4.2.36	MaSndDrv_PitchBend.....	51
4.2.37	MaSndDrv_BendRange	52
4.2.38	MaSndDrv_StreamOn	52
4.2.39	MaSndDrv_StreamOff.....	53
4.2.40	MaSndDrv_StreamSlave	53
4.2.41	MaSndDrv_StreamPanpot	53
4.2.42	MaSndDrv_MasterVolume	54
4.2.43	MaSndDrv_SystemOn.....	54
4.2.44	MaSndDrv_LedOn	55
4.2.45	MaSndDrv_LedOff.....	55
4.2.46	MaSndDrv_MotorOn.....	56
4.2.47	MaSndDrv_MotorOff	56
4.2.48	MaSndDrv_StreamSeek	57
5	MA Resource Manager.....	58
5.1	Function definition	59
5.1.1	MaResMgr_Initialize	59
5.1.2	MaResMgr_GetResourceInfo	59
5.1.3	MaResMgr_GetDefWaveAddress	59
5.1.4	MaResMgr_GetDefVoiceAddress	59
5.1.5	MaResMgr_GetDefVoiceSynth.....	60
5.1.6	MaResMgr_GetDefVoiceKey	60
5.1.7	MaResMgr_RegStreamAudio	60
5.1.8	MaResMgr_DelStreamAudio.....	60
5.1.9	MaResMgr_GetStreamAudioInfo	61

5.1.10	MaResMgr_AllocStreamAudio	61
5.1.11	MaResMgr_FreeStreamAudio	61
5.1.12	MaResMgr_AllocRam	62
5.1.13	MaResMgr_FreeRam	62
5.1.14	MaResMgr_AllocCh	62
5.1.15	MaResMgr_FreeCh	63
5.1.16	MaResMgr_AllocFmVoice	63
5.1.17	MaResMgr_FreeFmVoice	63
5.1.18	MaResMgr_AllocWtVoice	64
5.1.19	MaResMgr_FreeWtVoice	64
5.1.20	MaResMgr_AllocSoftInt	64
5.1.21	MaResMgr_FreeSoftInt	65
5.1.22	MaResMgr_AllocLed	65
5.1.23	MaResMgr_FreeLed	65
5.1.24	MaResMgr_AllocMotor	66
5.1.25	MaResMgr_FreeMotor	66
5.1.26	MaResMgr_AllocSequencer	66
5.1.27	MaResMgr_FreeSequencer	67
5.1.28	MaResMgr_AllocTimer	67
5.1.29	MaResMgr_FreeTimer	67
5.1.30	MaResMgr_SetStreamSeekPos	68
6	MA Device Driver	69
6.1	Function definition	70
6.1.1	MaDevDrv_Initialize	70
6.1.2	MaDevDrv_DeviceControl	70
6.1.3	MaDevDrv_InitRegisters	72
6.1.4	MaDevDrv_VerifyRegisters	72
6.1.5	MaDevDrv_PowerManagement	72
6.1.6	MaDevDrv_ReceiveData	73
6.1.7	MaDevDrv_SendDirectPacket	73
6.1.8	MaDevDrv_SendDelayedPacket	73
6.1.9	MaDevDrv_SendDirectRamData	74
6.1.10	MaDevDrv_SendDirectRamVal	74
6.1.11	MaDevDrv_StartSequencer	74
6.1.12	MaDevDrv_StopSequencer	75
6.1.13	MaDevDrv_EndOfSequence	75
6.1.14	MaDevDrv_ClearFifo	75
6.1.15	MaDevDrv_StreamSetup	75
6.1.16	MaDevDrv_StreamUpdate	76
6.1.17	MaDevDrv_ControlInterrupt	76
6.1.18	MaDevDrv_SetAudioMode	76
6.1.19	MaDevDrv_GetStreamPos	76
6.1.20	MaDevDrv_GetSeekBuffer	77
6.1.21	MaDevDrv_SeekControl	77

6.1.22	MaDevDrv_AddIntFunc.....	77
6.1.23	MaDevDrv_RemoveIntFunc	78
6.1.24	MaDevDrv_IntHandler.....	78
6.1.25	MaDevDrv_StreamHandler.....	78
6.1.26	MaDevDrv_SoftInt0.....	78
6.1.27	MaDevDrv_SoftInt1	79
6.1.28	MaDevDrv_SoftInt2.....	79
6.1.29	MaDevDrv_SoftInt3.....	79
6.1.30	MaDevDrv_Timer0	79
6.1.31	MaDevDrv_Timer1	80
6.1.32	MaDevDrv_Fifo	80
7	Default tones	81
7.1	Melody sound set	81
7.2	Percussion sound set.....	83

Notes

The explanation of the definition of functions uses the type of definition as described below.

Type of definition	Meaning	Type of definition	Meaning
UINT8	8 bits without code	SINT8	8 bits with code
UINT16	16 bits without code	SINT16	16 bits with code
UINT32	32 bits without code	SINT32	32 bits with code

This specification describes on the assumption that MIDI Key#60 = C4 = 440Hz with an equal temperament.

Definition of error codes

Error code name	Value	Meaning
MASMW_SUCCESS	0	Successful
MASMW_ERROR	-1	General purpose error
MASMW_ERROR_ARGUMENT	-2	Argument is erroneous.
MASMW_ERROR_RESOURCE_OVER	-3	Resource cannot be secured.
MASMW_ERROR_FILE	-16	File contents are incorrect.
MASMW_ERROR_CONTENTS_CLASS	-17	The value of unreproducible SMAF Contents Class
MASMW_ERROR_CONTENTS_TYPE	-18	The value of unreproducible SMAF Contents Type
MASMW_ERROR_CHUNK_SIZE	-19	SMAF Chunk Size is incorrect.
MASMW_ERROR_CHUNK	-20	SMAF Track Chunk is incorrect.
MASMW_ERROR_UNMATCHED_TAG	-21	Designated tag information is not present.
MASMW_ERROR_SHORT_LENGTH	-22	Sequence data is short.

Revision

Version	Date	Description	
0.7	April 23, 2001	Initial edition	
0.8	June 1, 2001	1.1	Module configuration drawing was changed.
		1.3	Function model was added to function specification. Contents of other explanation were changed. Description of reproduction system was added.
		1.4	Drawings related to principal functions were changed.
		2	Name of each function name of MA-3 Sound Sequencer was changed from Ma3SndSeq... to Ma3Sound_....
		2.1	Status transition was changed.
		2.2.1	Description of Ma3Sound_Initialize () was changed.
		2.2.2	Description of Ma3Sound_DeviceControl () was changed.
		2.2.3	Argument of Ma3Sound_Create () was changed.
		2.2.5	Argument of Ma3Sound_Load () was changed.
		2.2.7	Meaning of argument of Ma3Sound_Open () was changed.
		2.2.9	Meaning of argument of Na3Sound_Control () was changed.
		3	Description of MA-3 Stream Converter was added.
		3.2	Status transition was changed.
		3.3	Description of stream converter processing procedure was added.
		3.4.3	Argument of Ma3SrmCnv_Load () was changed.
		3.4.5	Description of Ma3SrmCnv_Open () was added.
		3.4.7	Meaning of argument of Ma3SrmCnv_Control () was changed.
		3.4.13	Ma3SrmCnv_SetBind () was added.
		4.1	Tone generation model was added.
		4.2.3	Argument of Ma3SndDrv_Create () was changed.
		4.2.6	Ma3SndDrv_GetPos ()was added.
		4.2.9	Function name of Ma3SndDrv_SetKeyControl () was changed.
		4.2.10	Ma3SndDrv_SetFmExtWave ()was added.
		4.2.13	Meaning of argument of Ma3SndDrv_SetCommand () was changed.
		4.2.14	Argument of Ma3SndDrv_Nop () was changed.
		4.2.15	Argument of Ma3SndDrv_UserEvent () was changed.
		4.2.16	Argument of Ma3SndDrv_NoteOn ()was changed.
		4.2.17	Argument of Ma3SndDrv_NoteOff () was changed.
		4.2.18	Argument of Ma3SndDrv_ProgramChange () was changed.
		4.2.19	Argument of Ma3SndDrv_ModulationDepth () was changed.
		4.2.20	Argument of Ma3SndDrv_ChannelVolume () was changed.
		4.2.21	Argument of Ma3SndDrv_Panpot () was changed.
		4.2.22	Argument of Ma3SndDrv_Expressin () was changed.
		4.2.23	Argument of Ma3SndDrv_Hold1 () was changed.
		4.2.24	Argument of Ma3SndDrv_AllSoundOff () was changed.
		4.2.25	Argument of Ma3SndDrv_ResetAllControllers () was changed.
		4.2.26	Argument of Ma3SndDrv_AllNoteOff () was changed.
		4.2.27	Argument of Ma3SndDrv_MonoModeOn () was changed.
		4.2.28	Argument of Ma3SndDrv_PolyModeOn () was changed.
		4.2.29	Argument of Ma3SndDrv_PitchBend () was changed.
		4.2.30	Ma3SndDrv_BendRange ()was added.
		4.2.31	Argument of Ma3SndDrv_StreamOn () was changed.
		4.2.32	Argument of Ma3SndDrv_StreamOff () was changed.
		4.2.33	Argument of Ma3SndDrv_StreamPanpot () was changed.

Version	Date	Description	
0.8	June 1, 2001	4.2.34	Argument of Ma3SndDrv_MasterVolume () was changed.
		4.2.35	Argument of Ma3SndDrv_SystemOn () was changed.
		4.2.36	Argument of Ma3SndDrv_LedOn () was changed.
		4.2.37	Argument of Ma3SndDrv_LedOff () was changed.
		4.2.38	Argument of Ma3SndDrv_MotorOn () was changed.
		4.2.39	Argument of Ma3SndDrv_MotorOff () was changed.
		5.1.3	Ma3ResMgr_GetDefWaveAddress () was added.
		5.1.9	Ma3ResMgr_GetStreamAudioInfo () was added.
		6.1.2	Ma3DevDrv_PowerManagement () was added.
		6.1.3	Meaning of argument of Ma3DevDrv_DeviceControl () was changed.
		6.1.7	Contents of Ma3DevDrv_SendDirectRamData () were changed.
		6.1.8	Contents of Ma3DevDrv_SendDirectRamVal () were changed.
		7	Default tone was added.
0.9	June 18, 2001	2.1	Status transition was changed.
		2.2.9	Contents of processing of Ma3Sound_Control () were changed.
		3.4.7	Contents of processing of Ma3SrmCnv_Control () were changed.
		3.4.16	play_mode was deleted from argument of Ma3SrmCnv_Start ().
1.0	June 29, 2001		All module names and function names were changed.
		2.2.2	Contents of processing of MaSound_DeviceControl () were changed.
		2.2.9	Contents of processing of MaSound_Control () were changed.
1.1.0	July 11, 2001		Values of error code definition were changed.
		2.2.2	Function of argument mode of MaSound_Load () was added.
		2.2.9	Description of MaSound_Control () was changed.
		3.4.3	Function of argument mode of MaSrmCnv_Load () was added.
1.2.0	June 18, 2001	2.2.9	Specification of acquisition of MaSound_Control () PhraseList was changed.
		4.2.12	The number of registrations of MaSndDrv_SetStream () was corrected.
1.2.1	June 25, 2001	2.2.7	The value of argument open_mode was added to MaSound_Open ().
		4.2.5	MaSndDrv_DeviceControl () was added.
1.2.2	August 3, 2001	2.1	Status transition was corrected.
		2.2.9	Description of MaSound_Control () was changed.
1.2.3	September 19, 2001	1.1	Module configuration figure was corrected.
		1.3.7	The number of registration of Stream Audio was corrected.
		1.3.10	Mode was unified into open_mode.
		2.1	Clerical error of Status transition diagram was corrected.
		2.2.5	Argument value of callback function of MaSound_Load() was corrected.
		2.2.7	Clerical error of the table of MaSound_Open() was corrected.
		3.2	Clerical error of Status transition diagram was corrected.
1.3.0	October 5, 2001	2.2.2	The command to MaSound_DeviceControl() was added.
		4.2.5	The command to MaSndDrv_DeviceControl() was added.
		6.1.3	The command to MaDevDrv_DeviceControl() was added.
1.4.0	October 31, 2001	2.2.9	The function was added to cmd of MaSound_Control().

Version	Date	Description	
1.4.1	Nov. 15, 2001	2.2.9	Description of MaSound_Control() was corrected.
		4.2.14	cmd No. of MaSndDrv_SetCommand() was changed.
		4.2.17	Description of MaSndDrv_NoteOn() was changed.
		4.2.18	Description of MaSndDrv_NoteOnMa2() was added.
		4.2.19	Description of MaSndDrv_NoteOnMa2Ex() was added.
		4.2.20	Description of MaSndDrv_NoteOff() was added.
		4.2.21	Description of MaSndDrv_NoteOffMa2() was added.
		4.2.22	Description of MaSndDrv_NoteOffMa2Ex() was added.
		4.2.36	Description of MaSndDrv_StreamOn() was corrected.
		4.2.37	Description of MaSndDrv_StreamOff() was corrected.
		4.2.38	Description of MaSndDrv_StreamSlave() was added.
		4.2.39	Description of MaSndDrv_StreamPanpot() was corrected.
1.5.0	Nov. 26, 2001	1.1	Module configuration was changed.
		2.2.7	Description of MaSound_Oepn() was changed.
		3.4.5	Description of MaSrmCnv_Open() was changed.
		3.4.7	Description of MaSrmCnv_Control() was changed.
		6.1.9	Description of MaDevDrv_StartSequencer() was added.
		6.1.10	Description of MaDevDrv_StopSequencer() was added.
		6.1.11	Description of MaDevDrv_EndOfSequence() was added.
		6.1.12	Description of MaDevDrv_ClearFifo() was added.
		6.1.13	Description of MaDevDrv_StreamSetup() was added.
		6.1.14	Description of MaDevDrv_StreamUpdate() was added.
		6.1.15	Description of MaDevDrv_ControlInterrupt() was added.
		6.1.16	Description of MaDevDrv_SetAudioMode() was added.
		6.1.17	Description of MaDevDrv_GetStreamPos() was added.
		6.1.18	Description of MaDevDrv_GetSeekBuffer () was added.
		6.1.19	Description of MaDevDrv_SeekControl () was added.
		6.1.23	Description of MaDevDrv_StreamHandler() was added.
		6.1.24	Description of MaDevDrv_SoftInt0() was added.
		6.1.25	Description of MaDevDrv_SoftInt1() was added.
		6.1.26	Description of MaDevDrv_SoftInt2() was added.
		6.1.27	Description of MaDevDrv_Timer0() was added.
		6.1.28	Description of MaDevDrv_Timer1() was added.
		6.1.29	Description of MaDevDrv_Fifo() was added.
1.6.0	Dec. 07, 2001	2.2.9	Control command of MaSound_Control() was added.
		4.2.5	Description of MaSndDrv_DeviceControl() was added.
1.6.1	Dec. 20, 2001	1.3.5	Clerical error was corrected.
1.6.1	Dec. 20, 2001	1.3.9	Clerical error was corrected.
		2.2.3	Clerical error was corrected.
		2.2.12	Description of MaSound_Start() was added.
		3.2	Clerical error was corrected.
		5.1.3	Clerical error of MaResMgr_GetDefWaveAddress() was corrected.
		6.1.9	Clerical error of MaDevDrv_StartSequencer() was corrected.
		6.1.10	Clerical error of MaDevDrv_StopSequencer() was corrected.
		6.1.18	Clerical error of MaDevDrv_GetSeekBuffer() was corrected.
		6.1.19	Clerical error of MaDevDrv_SeekControl() was corrected.
1.6.2	January 17, 2002	1.3.7	Contents explanation was changed.
		1.3.8	Contents explanation was changed.
		1.3.9	Contents explanation was changed.
		2.1	Ccorrected that state transition diagrams differed.

Version	Date	Description	
1.6.2	January 17, 2002	6.1.1	The return value was changed into SINT32 from void.
		6.1.2	The return value was changed into SINT32 from void.
		6.1.4	MaDevDrv_InitRegisters() was added newly.
		6.1.5	MaDevDrv_VerifyRegisters() was added newly.
		6.1.27	MaDevDrv_SoftInt3() was added.
1.7.0	January 30, 2002	1.1	SMF0 of the module configuration figure was corrected to SMF. Clerical error was corrected.
		1.3.3	The notation was changed into 1024bytes from 1Kbytes.
		1.4	Drawing was changed.
		1.3.5	The byte count of Cyclic Buffer was added.
			About Fs of the value of a table, ADPCM 4kHz-48kHz was changed into 4kHz – 32kHz, and PCM 4kHz-24kHz was changed into 4kHz – 16kHz.
			About Buffer of the value of a table, 1Kbytes was changed into 992bytes.
			About the time of the value of a table, min 42.6ms was changed at min 60ms.
		1.3.7	The number of voice registration was changed into 254 from 256.
			7 (1) was added to WT (Stream) of SMAF/MA-3.
		1.3.8	The number of voice registration was changed into 254 from 256.
			Voice definition support of FM/WT in MIDI was deleted.
		1.3.9	SMAF/MA-2 was added to the correspondence format.
			Fs value was changed into 4K-16 / 32kHz from 4K-24 / 48kHz.
		1.4	Drawing was changed.
		2.2.3	The description of SMAF/Phrase L1 was deleted.
			SMF format 0 was changed into SMF format0 or 1.
		2.2.5	In the definition of id value of a callback function, “0-125: user events” were changed to “0-15: user event” and “16-125: Reserved”.
		2.2.7	7 (1) was added to WT (Stream) of a sequencer system and a real-time system.
			The value of RAM was changed into bytes from Kbytes, respectively.
		2.2.9	ctrl_num=9: synchronous setup was changed into Reserved.
			MASMW_SET_BIND was deleted among ctrl_num definitions.
		2.2.11	Assignment of a flag value was changed into the unused or set 0.
		3	The MaSrmCnv_SetBind function was deleted.
		3.1	SMF format 0 was changed into SMF format0 or 1.
			The description of SMAF/Phrase was deleted.
		3.3.3	Explanation of Built-in RAM was changed.
		3.3.6	Explanation was changed.
		3.4.3	The description of SMAF/Phrase was deleted.
		3.4.7	ctrl_num=9: synchronous setup was changed into Reserved.
		3.4.14	Assignment of a flag value was changed into the unused or set 0.
		4.1	Channel preference was deleted.
		4.2.12	The number of voice registration was changed into 254 voices from melody voice 128 voice and drum voice 127 voice.
		6	MaDevDrv_InitRegisters, MaDevDrv_VerifyRegisters, MaDevDrv_SoftInt3 were added.
		6.1.7	Argument UINT8 *ptr was changed into const UINT8 *ptr.
		6.1.8	Argument UINT8 *ptr was changed into const UINT8 *ptr.
		6.1.9	Argument UINT8 *data_ptr was changed into const UINT8 *data_ptr.
		6.1.19	Argument void was changed into UINT8 ctrl. Explanation of ctrl was added.
		6.1.22	Argument INTFUNC func was changed into void (UINT(*int_func)8 ctrl).
		6.1.25	Argument UINT8 ram_val was added.

Version	Date	Description	
1.7.1	February.20, 2002	1.3.2	The explanation of Software Interrupt #2/#3 was changed.
		1.3.5	About Fs of the value of a table, ADPCM 4kHz-32kHz was corrected to 4kHz – 24kHz, and PCM 4kHz-16kHz was corrected to 4kHz – 12kHz.
			About the time of the value of a table, min 62ms was corrected at min 82.6ms.
		1.3.9	Fs value was corrected from 4K-16 / 32kHz to 4K-12 / 24kHz.
		2.2.1	Explanation of a return value was corrected.
		2.2.9	The clerical error in the table of ctrl_num definition was corrected.
		3.4.7	Explanation was corrected. Explanation of ctrl_num=17/18/19/20 was added.
		4.1	A figure was corrected.
		5	In the contents of MaResMgr_AllocRam in the table (1Kbyte) was deleted.
		5.1.1	A explanation was corrected.
		5.1.3	Return value 0: No registration was added.
		5.1.4	The error code of Return value, Negative number was added.
		5.1.8	A clerical error was corrected
		5.1.10	The explanation of argument was added.
		5.1.11	The explanation of argument was added.
		5.1.20	The explanation of argument was added.
		5.1.21	The explanation of argument was added.
			The explanation of Argument was changed to bit3 – bit0 from bit0.
		5.1.22	The explanation of argument was added.
1.8.0	March.8, 2002	4.2.3	The type setting function of Percussion sound set was added to the argument cnv_mode of MaSndDrv_Create
		4.2.15	A clerical error of description of MaSndDrv_SetCommand was corrected.
1.8.1	March.27, 2002	4.2.12	MaSndDrv_SetVolume was added.
		4.2.13	The contents of description of MaSndDrv_SetVoice were changed. The setup specification of key was changed.
		4.2.37	The setting range of MaSndDrv_BendRange range was changed into 0-24 from 1-24.
1.8.2	May.20, 2002	2.2.6	The type of argument file_id of MaSound_Unload was corrected from UINT8 to SINT32.
1.9.0	January 16, 2003	1.1	WAV playback system path was added to the figure.
		1.3.9	WAV playback was added to audio system.
		1.3.10	“Simultaneous reproduction of systems” was changed to “Simultaneous reproduction of systems between format”
		2.2.9	The following items were added to MaSound_Contorl
		3.4.7	- Designation of playback speed - Designation of playback count - Acquisition of playback volume
1.10.0	April 22, 2003	1.3.5	The upper limit of PCM of Fs was changed from 12KHz to 24KHz. The time of PCM was changed from min 82.6ms to 41.3ms. The footnote was added.
		1.3.9	The explanation of Audio system was devided into this spec. and Specification according to format. The upper limit of SMAF/Audio playback Fs was changed from 12KHz/24KHz to 8KHz/16KHz. The upper limit of WAV playback Fs was changed from 16KHz to 24KHz.
		1.4	Figure was changed.
		2.2.3	The definition name was corrected from MA3SMW_ to MASMW_. WAV was added to the definition.

		3.1	WAV was added to the format to be supported.
		3.4.3	WAV was added to the table.
		4	MaSndDrv_StreamSeek was added to the table.
		4.2.16	StreamSeek was added to MaSndDrv_SetCommand.
		4.2.48	MaSndDrv_StreamSeek was added.
		5	MaResMgr_SetStreamSeekPos was added to the table.
		5.1.9	The argument *seek_pos was added to MaResMgr_GetStreamAudioInfo.
		5.1.30	MaResMgr_SetStreamSeekPos was added.

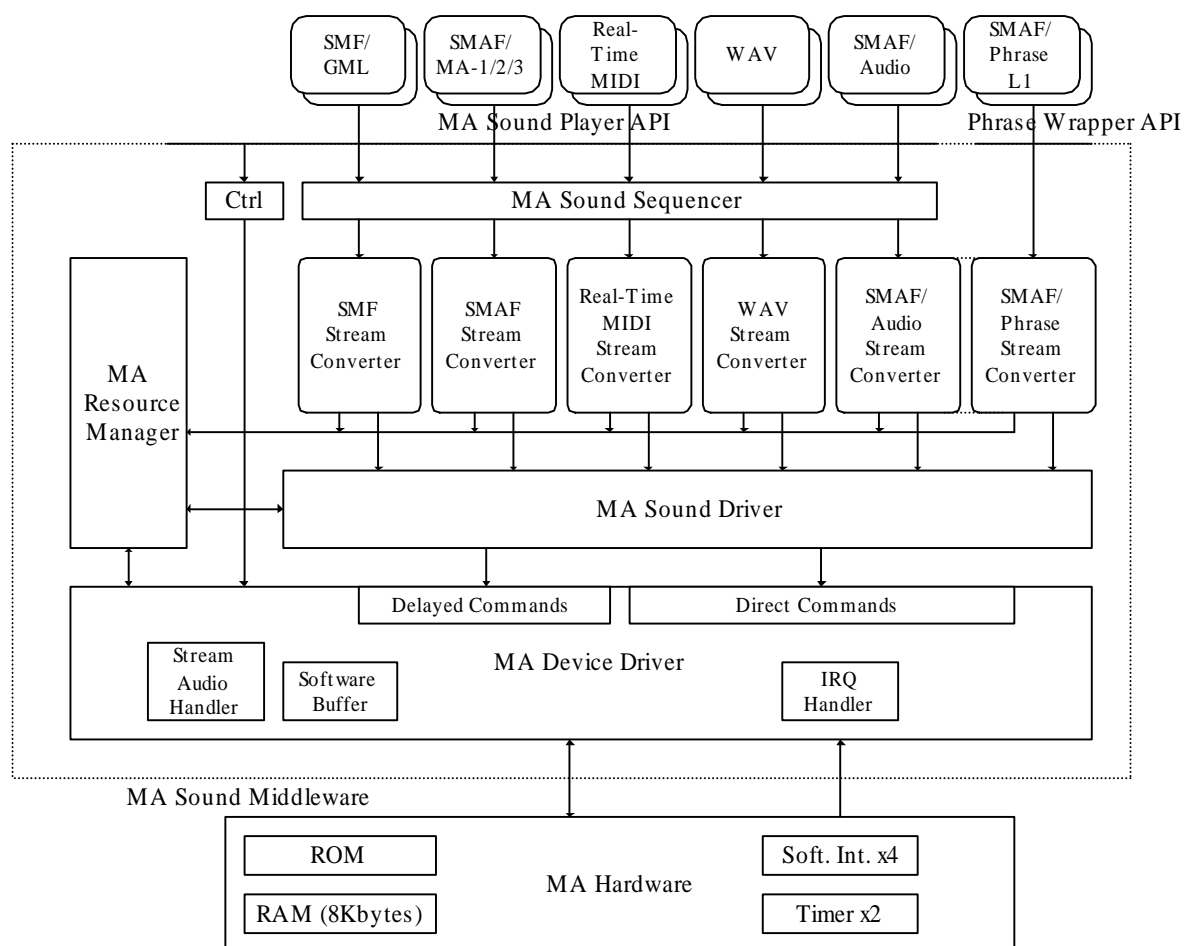
1 Introduction

This document defines API (Application Program Interface) so that the functions of Mobile Audio LSI “MA-3” (hereafter called MA-3) can be used from OS or applications, and shows the specifications for functions of Sound Middleware that performs the installation. It also describes the detailed contents of it.

Refer to MA-3 Sound Middleware Release Note about actual application.

1.1 Module configuration

The module configuration of MA Sound Middleware is as shown below. MA Sound Middleware consists of multiple modules. The Stream Converter modules are independent from each other, and thus, it is possible to delete any unnecessary module.



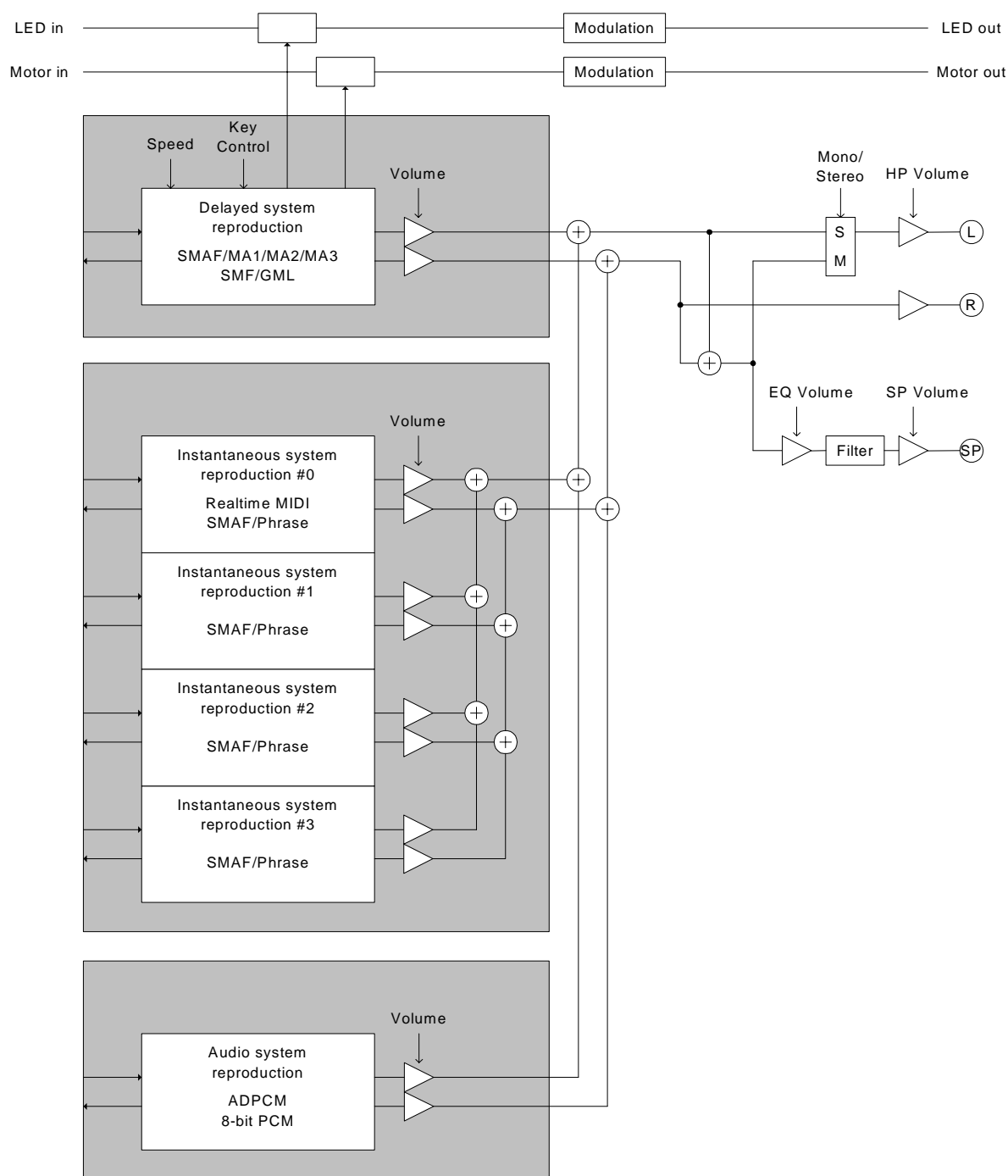
1.2 Function of each module

- MA Sound Sequencer
Provides a group of functions that performs control of stream converter that performs conversion processing dynamically so that musical pieces format data can be reproduced.
- MA Stream Converter
Provides a group of functions that performs stream conversion of each data format.
- MA Sound Driver
Provides a group of functions for performing tone generation.
- MA Resource Manager
Performs management of hardware resources. Provides a group of functions for the state of utilization of hardware resources, and securing / release of resources, etc.
- MA Device Driver
Performs control of devices. Provides a group of functions that performs initialization of devices, and read / write of each register.

1.3 Functional specifications

This Sound Middleware Specification uses hardware resources by defining specifications as described below.

1.3.1 Functional model



1.3.2 Timer and software interrupt

Timer	
Timer #0	Used for both supplement of stream audio reproduction data and real time system converter
Timer #1	Used for delay system converter
Software interrupt	
Software Interrupt #0	Used for control of Stream Audio #0 reproduction (WT #7)
Software Interrupt #1	Used for control of Stream Audio #1 reproduction (WT #6)
Software Interrupt #2	For notification of User event
Software Interrupt #3	For notification of End

1.3.3 Built-in SRAM

- Management of Built-in SRAM is performed by MA Resource Manager. The unit of management is block, and one block is equal to 1024 bytes.
- When the hardware sequencer is in operation, rewriting of the contents of SRAM is not performed except updating of data at Stream Audio reproduction. Therefore, tone parameters except Stream Audio data are supported to the extent that can be stored in the SRAM.

1.3.4 Tone generation channel / FM Voice / WT Voice

- This is managed by MA Resource Manager.

1.3.5 Reproduction of Stream Audio

- RAM of 1024 bytes per one Stream Audio is secured and used 992 bytes among those as Cyclic Buffer.
- Up to two Stream Audios are used, and the first one is assigned to WT #7 and second one to WT #6.
- When reproduction of Stream Audio is performed, Timer #0 is used exclusively. For 20 ms or less, continuous mode is set.
- Software interrupt #0 (WT#7)/#1 (WT#6) is used as the reproduction control signal.

	ADPCM	PCM
Fs	4KHz to 24KHz	4kHz to 24KHz
Bit	4-bit	8-bit
Buffer	992 bytes	992 bytes
Time	min 82.6ms	min 41.3ms

Note: Max of Fs of PCM playback is 24KHz, but the limitation of Max of Fs is different by the playback format.

1.3.6 Synchronization of Motor / LED

- Converter integrates ON/OFF of Motor/LED into the sequence.
- Exclusive channels for Motor/LED are needed. (Number of generated tones are not reduced.)
- For poli-tone generation, all notes are subject to OR operation.
- Fundamental control and mode setting are executed from upper side by API.

1.3.7 Delayed (Sequencer) system

Executes reproduction that uses hardware sequencer. It also deals with key control and tempo control. The system uses registered tones of up to 254 for FM / WT, and those of up to 32 for Stream Audio. It supports LED / Motor synchronization.

Format	FM mode	CH	FM	WT (Stream)
SMAF/MA-1	4OP	4	4	0 (0)
SMAF/MA-2	4OP	16	16	0 (0)/0 (1)
SMAF/MA-3	2OP	16	32	8 (0)/7 (1)/6 (2)
	4OP	16	16	8 (0)/7 (1)/6 (2)
SMF/GM Lite	2OP	16	32	8 (0)
	4OP	16	16	8 (0)

1.3.8 Real time system

The system uses registered tones of up to 254 for FM / WT. It does not support Stream Audio. Default for MIDI, 2OP or 4OP is designated by the definition file. For Phrase, the system treats four sequences.

Format	FM mode	CH	FM	WT (Stream)
SMAF/Phrase × 4	4OP	16	16	0 (0)
MIDI/GM Lite	2OP	16	32	8 (0)
	4OP	16	16	8 (0)

1.3.9 Audio system

Executes reproduction of audio data.

It complies with Yamaha spec. ADPCM, 8 bit offset bin PCM and 8 bit 2's comp bin PCM. Supports Fs ranging from 4 k ~ 12 kHz (8bit) / 24 (4bit)kHz.

In SMAF/Audio playback, both of ADPCM and PCM are made applicable of playback as waveform format, and Fs supports 4K ~ 8KHz(PCM) / 4K ~ 16KHz (ADPCM).

WAV playback considers only 8bit PCM as waveform format to playback and Fs supports 4K~24KHz.

It executes a call of callback function at the end of a reproduction to notify of the end.

Format	FM mode	CH	FM	WT(Stream)
SMAF/Audio/MA-2	-	0	0	0(1)
SMAF/Audio/MA-3	-	0	0	0(1)
WAV	-	0	0	0(1)

1.3.10 Simultaneous reproduction of systems

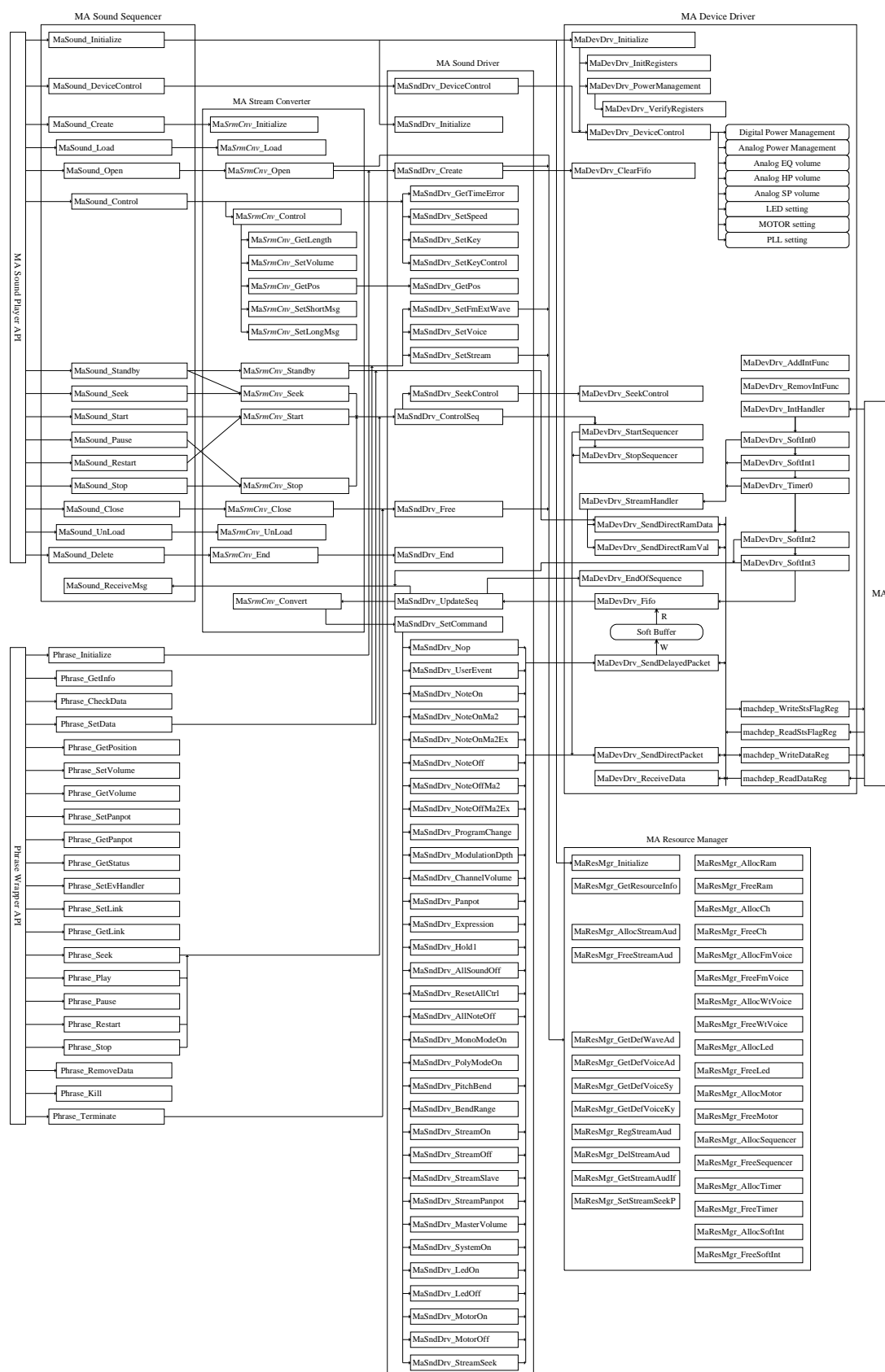
Usually, MA-3 sound middleware supports reproducing only one data simultaneously but reproducing one or more data simultaneously for SMAF/Phrase and SMAF/Audio.

Describes the maximum reproduction number for every format in the following.

Format	Maximum number of tones	Note
SMAF / MA-1 / 2 / 3	1	Cannot reproduce simultaneously with other data
SMF / GM Lite	1	Cannot reproduce simultaneously with other data
SMAF / Phrase	4	Can reproduce simultaneously with SMAF / Audio
MIDI / GM Lite	1	Cannot reproduce simultaneously with other data
SMAF / Audio / MA-2 / 3	1	Can reproduce simultaneously with SMAF / Phrase
WAV	1	Cannot reproduce simultaneously with other data

* Supports to reproduce of maximum 5 data simultaneously by using SMAF/Phrase and SMAF/Audio.

1.4 Relationship among principal functions

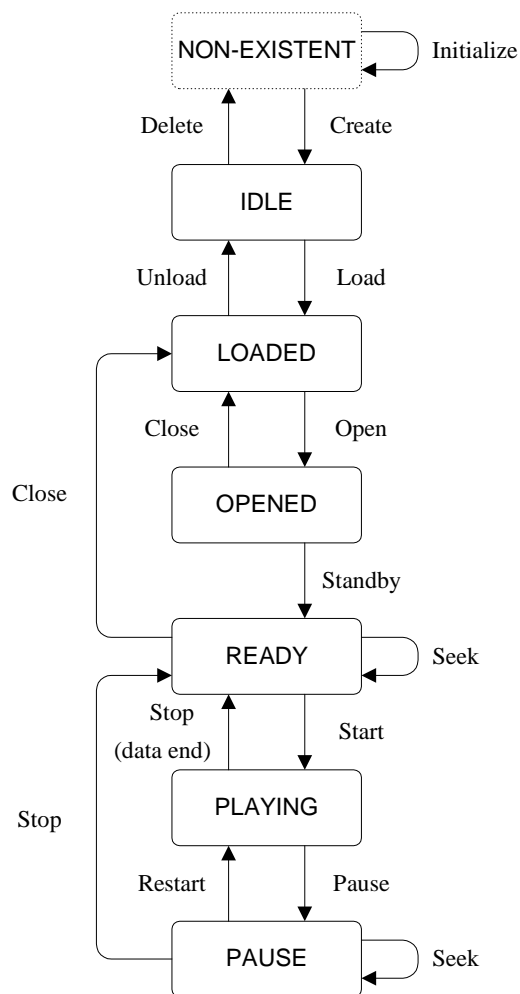


2 MA Sound Sequencer

MA Sound Sequencer is a module that defines API (Application Program Interface) for controlling MA Sound Middleware from an application.

Function name	Description
MaSound_Initialize	Executes initialization of MA Sound Sequencer.
MaSound_DeviceControl	Executes setting of devices.
MaSound_Create	Registration of MA Stream Converter
MaSound_Load	Sequencer loading processing
MaSound_Open	Sequencer opening processing
MaSound_Control	Sets control value.
MaSound_Standby	Sequencer standby processing
MaSound_Seek	Sequencer seeking processing
MaSound_Start	Sequencer starting processing
MaSound_Pause	Sequencer pause processing
MaSound_Restart	Sequencer restart from pause processing
MaSound_Stop	Sequencer stopping processing
MaSound_Close	Sequencer closing processing
MaSound_Unload	Sequencer unloading processing
MaSound_Delete	Deletion of registration of MA Stream Converter

2.1 Status transition



Status	Meaning
NON-EXISTENT	Unregistered
IDLE	Idling
LOADED	File loaded
OPENED	Resources are so secured that loaded file can be reproduced.
READY	Ready for reproduction
PLAYING	Reproduction in progress
PAUSE	Pausing

2.2 Function definition

2.2.1 MaSound_Initialize

SINT32 MaSound_Initialize (void);

Description

Executes initialization of MA Sound Sequencer.

Argument

None

Returned value

0 Successful.
Negative Error code.

- Initialization of MA Device Driver is performed, and power down mode cancellation and software reset for devices are performed. PLL setting is also performed.
- Initialization of MA Resource Manager is performed. All the resources become released state.
- Initialization of MA Sound Sequencer is performed.
- Initialization of MA Sound Driver is performed.
- Since hardware reset resets the volume setting of analog section, it is necessary to perform the volume setting of analog section by using MaSound_DrviceControl.

2.2.2 MaSound_DeviceControl

SINT32 MaSound_DeviceControl (UINT8 cmd, UINT8 param1, UINT8 param2, UINT8 param3);

Description

Performs various settings of devices.

Argument

cmd Command number
param1 Parameter 1
param2 Parameter 2
param3 Parameter 3

Setting	cmd	param1	param2	param3
Power management (digital)	0	val	0	0
Power management (analog)	1	val	0	0
EQ Volume	2	eq_vol	0	0
HP Volume	3	mono	vol_l	vol_r
SP Volume	4	hp_vol	0	0
LED control source setting	5	source	0	0
LED blinking setting	6	val	0	0
LED forced control	7	sw	0	0
Motor control source setting	8	source	0	0
Motor blinking setting	9	val	0	0
Motor forced control	10	sw	0	0
Acquisition of PLL value	11	0	0	0
Acquisition of sequencer processing flag	12	0	0	0

Returned value

Non-negative Successful. When a value is returned, the value.
Negative Error code

2.2.3 MaSound_Create

SINT32 MaSound_Create (UINT8 cnv_id);

Description

Registers the designated MA Stream Converter and initializes it.

Argument

cnv_id ID number of MA Stream Converter

Returned value

Non-negative Function ID of registered MA Stream Converter
 Negative Error code

Name	Value	Stream converter
MASMW_CNVID_MMF	1	SMAF/MA-1/MA-2/MA-3
MASMW_CNVID_RMD	3	Realtime MIDI
MASMW_CNVID_AUD	4	SMAF/Audio
MASMW_CNVID_MID	5	SMF format 0 or 1/GM Lite or Level 1
MASMW_CNVID_WAV	11	WAV

2.2.4 MaSound_Delete

SINT32 MaSound_Delete (SINT32 func_id);

Description

Deletes MA Stream Converter that has been registered with func_id.

Argument

func_id func_id

Returned value

0 Successful
 Negative Error code

2.2.5 MaSound_Load

SINT32 MaSound_Load (SINT32 func_id, UINT8 *file_ptr, UINT32 file_size, UINT8 mode, SINT32 (*func) (UINT8 id), void *ext_args);

Description

Performs loading processing of MA Stream Converter.

Contents of the Load processing includes preparation for registration of designated format data, interpretation of format data on the memory, and processing of MA Stream Converter. Error is returned when the format data is abnormal.

Argument

func_id	Function ID of MA Stream Converter to be designated.			
file_ptr	Pointer for file data storage domain			
file_size	Byte size of file data			
mode	Designates format check for file data.			
	0:	Check is not performed.		
	1:	Check is performed.		
	2:	Only check is performed. Since internal information is not acquired, it cannot be opened after this.		
	3:	Performs acquisition of contents information. Since internal information is not acquired, it cannot be opened after this.		
func	Callback function	id	0 to 15	User event
			16 to 125	Reserved
			126	Notification of repeat
			127	Notification of sequence end
ext_args	Extension argument specific to each MA Stream Converter. NULL is set when no extension argument is needed.			

Returned value

Non-Negative	File ID
Negative	Error code

2.2.6 MaSound_Unload

SINT32 MaSound_Unload(SINT32 func_id, SINT32 file_id, void *ext_args);

Description

Performs unloading processing of MA Stream Converter.

The unloading processing release internal work domain that has been secured by loading.

Be sure that MaSound_Load and MaSound_Unload with the same func_id are to be used as a pair.

Argument

func_id	Function ID of MA Stream Converter to be designated.		
file_id	File ID		
ext_args	Extension argument specific to each MA Stream Converter. NULL is set when no extension argument is needed.		

Returned value

0	Successful
Negative	Error code

2.2.7 MaSound_Open

SINT32 MaSound_Open (SINT32 func_id, SINT32 file_id, UINT16 open_mode, void * ext_args);

Description

Performs opening processing of MA Stream Converter.

Contents of the opening processing include securing of resources necessary for format data that has been registered by loading and judgement for possibility of reproduction.

Argument

func_id	Function ID of MA Stream Converter to be designated.
file_id	File ID
open_mode	Function mode (0/1/16)
ext_args	Extension argument specific to each MA Stream Converter. NULL is set when no extension argument is needed.

[Sequencer system]

open_mode	FM-OP	CH	FM	WT (Stream)	RAM
0	4/2-OP	16	16(32)	8(0)/7(1)/6(2)	8176bytes

[Real time system]

open_mode	FM-OP	CH	FM	WT (Stream)	RAM
0	4/2-OP	16	16(32)	8(0)/7(1)/6(2)	8176bytes
16	4-OP	16	16	7(0)	7152bytes

[Audio system]

open_mode	FM-OP	CH	FM	WT (Stream)	RAM
1	-	0	0	0(1)	1024bytes

Returned value

0	Successful
Negative	Error code

2.2.8 MaSound_Close

SINT32 MaSound_Close (SINT32 func_id, SINT32 file_id, void * ext_args);

Description

Performs closing processing of MA Stream Converter.

The closing processing releases MA hardware resource that has been secured by the opening processing.

Argument

func_id	Function ID of MA Stream Converter to be designated.
file_id	File ID
ext_args	Extension argument specific to each MA Stream Converter. NULL is set when no extension argument is needed.

Returned value

0	Successful
Negative	Error code

2.2.9 MaSound_Control

SINT32 Ma3Sound_Control (SINT32 func_id, SINT32 file_id, UINT8 ctrl_num, void *prm, void * ext_args);

Description

Performs controlling processing of MA Stream Converter

Acquisition of present internal status (6) is dealt with after MaSound_Create ().

Setting of reproduction start point (12) and setting of reproduction end point (13) are dealt with only after MaSound_Open () and before MaSound_Standby ()

Acquisition of contents information (10) and acquisition of Phrase List information (11) can be done after MaSound_Load (), but other controls are dealt with only after MaSound_Open ().

Acquisition (19) of the information for playback can be performed after MaSound_Load (mode=2/1).

Registration (20) of the information for playback can be performed after MaSound_Create() before MaSound_Load().

Designation of the playback count (26) can be performed when the state of Sound middleware is Ready and Pause.

* Since effective ctrl_num is different by each converter, refer to the API specification of each converter.

Argument

func_id	Function ID of MA Stream Converter to be designated.
file_id	File ID
ctrl_num	Designates contents of processing.
	0: Sets volume of reproduction (0 - 127). Default is 100. Volume designation[dB] = 40 * Log (val / 127)
	1: Designates speed of reproduction (70 - 100 - 130). Default is 100.
	2: Designates relative change of reproduction key (-12 - 0 - +12). Default is 0.
	3: Acquires difference between basic time setting and actual value of time.
	4: Acquires reproduction position (unit: ms)
	5: Acquires reproduction time (unit: ms)
	6: Acquires present internal status.
	7: Sends usual MIDI message. (Specific to real time MIDI)
	8: Sends SysEx MIDI message. (Specific to real time MIDI)
	9: Reserved
	10: Acquires designated data of contents information.
	11: Acquires Phrase List information.
	12: Sets start point of reproduction.
	13: Sets end point of reproduction.
	14: Sets value of panpot (0 - 127). Default is 64.
	15: Acquires LED synchronization status. "1" is returned when LED synchronization is set for opened data and LED synchronization is set for sequence synchronization.
	16: Acquires vibration motor synchronization status. "1" is returned when VIB synchronization is set for opened data and VIB synchronization is set for sequence synchronization.
	17: Designates the Note number which generates a user event. (0 - 127)
	18: Acquires the sequence time under conversion. (unit: ms)
	19: Acquires the information for playback.
	20: Registers the information for playback.
	26: Designate the playback speed (25 - 100 - 400). Default is 100.
	27: Designate the playback count (0 - 255, 0: Infinitely). Default is not set.
	29: Acquires the playback volume.
	30: Sets the period callback.
prm	Parameter needed for processings that are designated with ctrl_num.
ext_args	Extension argument specific to each MA Stream Converter. NULL is set when no extension argument is needed.

Returned value

Non-negative Successful. When a value is returned, the value.

Negative Error code

■ ctrl_num definition

Name	Value	Message
MASMW_SET_VOLUME	0	Sets volume for reproduction.
MASMW_SET_SPEED	1	Sets speed of reproduction.
MASMW_SET_KEYCONTROL	2	Designates relative change of reproduction key.
MASMW_GET_TIMERERROR	3	Acquires difference between basic time setting and actual value of time.
MASMW_GET_POSITION	4	Acquires position of reproduction.
MASMW_GET_LENGTH	5	Acquires time of reproduction.
MASMW_GET_STATE	6	Acquires internal status.
MASMW_SEND_MIDIMSG	7	Sets usual MIDI message.
MASMW_SEND_SYSEXMIDIMSG	8	Sets SysEx MIDI message.
MASMW_GET_CONTENTSDATA	10	Acquires contents information.
MASMW_GET_PHRASELIST	11	Acquires Phrase List information.
MASMW_SET_STARTPOINT	12	Sets start point of reproduction.
MASMW_SET_ENDPOINT	13	Sets end point of reproduction.
MASMW_SET_PANPOT	14	Sets panpot.
MASMW_GET_LEDSTATUS	15	Acquires LED synchronous status.
MASMW_GET_VIBSTATUS	16	Acquires the vibration motor synchronous status.
MASMW_SET_EVENTNOTE	17	Designates the Note number which generates a user event.
MASMW_GET_CONVERTERTIME	18	Acquires the sequence time under conversion.
MASMW_GET_LOADINFO	19	Acquires the information for playback.
MASMW_SET_LOADINFO	20	Registers the information for playback.
MASMW_SET_SPEEDWIDE	26	Designates the playback speed.
MASMW_SET_REPEAT	27	Designates the playback count.
MASMW_GET_CONTROL_VAL	29	Acquires the playback volume.
MASMW_SET_CB_INTERVAL	30	Designates the period callback.

■ Relationship between ctrl_num and I/O

ctrl#	Content	I/O		
0	Sets tone for reproduction.	prm	UINT8 *	Value of volume (0 - 127)
1	Sets speed of reproduction.	prm	UINT8 *	Value of speed (70 - 130)
2	Sets relative change of reproduction key.	prm	SINT8 *	Value of relative change (-12 - 12)
3	Acquires difference of basic time.	prm	NULL	
		return	Value of time difference	
4	Acquires position of reproduction.	prm	NULL	
		return	Reproduction position [ms]	
5	Acquires time of reproduction.	prm	NULL	
		return	Reproduction time [ms]	
6	Acquires internal status.	prm	NULL	

		return	Value of internal status	
7	Usual MIDI message	prm	UINT32 * msg	MIDI message (setting from LSB side)
8	SysEx MIDI message	prm	MASMW_MIDIMSG structural body	
			UINT8 * msg	Pointer for MIDI message
			UINT32 size	Size of MIDI message
10	Acquires data of contents information.	prm	MASMW_CONTENTSINFO structural body	
			UINT16 code	Code type
			UINT8 tag[2]	Tag name to be read
			UINT8 * buf	Buffer pointer of destination of data storage
			UINT32 size	Buffer size of destination of data storage
		return	No. of bytes of tag contents (When the buffer size is exceeded, data to the point is set and error value is returned.)	
11	Acquires Phrase List information.	prm	MASMW_PHRASELIST structural body	
			UINT8 tag[2]	Tag name to be read
			UINT32 start	Start point [ms]
			UINT32 stop	Stop point [ms]
12	Sets start point of reproduction.	prm	UINT32 *	Start point [ms]
13	Sets end point of reproduction.	prm	UINT32 *	End point [ms]
14	Sets value of panpot.	prm	UINT8 *	Value of panpot (0 - 127)
15	Acquires LED synch. status	return	0: No LED synch. setting, 1: LED synch. setting	
16	Acquires VIB synch. status	return	0: No VIB synch. setting, 1: VIB synch. setting	
17	User event Note number	prm	MASMW_EVENTNOTE structure	
			UINT8 ch	Channel number (0-15)
			UINT8 note	Note number (0-127)
18	Time acquisition under conversion	prm	NULL	
		return	Sequence data time under conversion [ms]	
19	Acquisition of the information for playback	prm	UINT8 *info	The pointer to the information write-in area for playback (128 bytes of size is required)
20	Registration of the information for playback	prm	UINT8 *info	The pointer to the information storing area for playback (128 bytes of size is required)
26	Designates the playback speed	prm	UINT32 *	Speed value (25 - 400)
27	Designates the playback count	prm	UINT8*	Playback count (0 - 255). However, 0 is infinitely
29	Sets the period callback	prm	PMASMW_GETCTL structure	
			bControl	Control number
			bCh	Channel number
		return	Volume number	
30	Sets the period callback	prm	UINT8*	Call time [msec] However, supports only 0 and 20. 0 is interpreted as callback-free.

2.2.10 MaSound_Standby

SINT32 MaSound_Standby (SINT32 func_id, SINT32 file_id, void * ext_args);

Description

Performs standby processing of MA Stream Converter.

Content of standby processing include preparation for immediately starting reproduction when Start is executed. Setting initial value is performed.

Argument

func_id	Function ID of MA Stream Converter to be designated.
file_id	File ID
ext_args	Extension argument specific to each MA Stream Converter. NULL is set when no extension argument is needed.

Returned value

0	Successful
Negative	Error code

2.2.11 MaSound_Seek

SINT32 MaSound_Seek (SINT32 func_id, SINT32 file_id, UINT32 pos, UINT8 flag, void * ext_args);

Description

Performs seeking processing of MA Stream Converter.

The seek cannot be performed during reproduction.

Argument

func_id	Function ID of MA Stream Converter to be designated.
file_id	File ID
pos	Reproduction start position (ms)
flag	0 fixed
ext_args	Extension argument specific to each MA Stream Converter. NULL is set when no extension argument is needed.

Returned value

0	Successful
Negative	Error code

2.2.12 MaSound_Start

SINT32 MaSound_Start (SINT32 func_id, SINT32 file_id, UINT16 play_mode, void * ext_args);

Description

Performs starting processing of MA Stream Converter.
 The starting processing performs starting of reproduction.
 play_mode which is published after MaSound_Standby() is effective until next MaSound_Standby().
 Forbids change of the numeric value in the middle. However, after an automatic end can be reconfigured.

Argument

func_id	Function ID of MA Stream Converter to be designated.
file_id	File ID
play_mode	Reproduction mode. May take the following values.
	0 Loop reproduction
	1 - 255 Reproduction count
	Other than the above Reserved
ext_args	Extension argument specific to each MA Stream Converter. NULL is set when no extension argument is needed.

Returned value

0	Successful
Negative	Error code

2.2.13 MaSound_Stop

SINT32 MaSound_Stop (SINT32 func_id, SINT32 file_id, void * ext_args);

Description

Performs stopping processing of MA Stream Converter.
 The stopping processing performs stopping of reproduction.

Argument

func_id	Function ID of MA Stream Converter to be designated.
file_id	File ID
ext_args	Extension argument specific to each MA Stream Converter. NULL is set when no extension argument is needed.

Returned value

0	Successful
Negative	Error code

2.2.14 MaSound_Pause

SINT32 MaSound_Pause (SINT32 func_id, SINT32 file_id, void * ext_args);

Description

Stops sequence data reproduction temporarily.
For audio systems, this operation is the same as Stop.

Argument

func_id	Function ID of MA Stream Converter to be designated.
file_id	File ID
ext_args	Extension argument specific to each MA Stream Converter. NULL is set when no extension argument is needed.

Returned value

0	Successful
Negative	Error code

2.2.15 MaSound_Restart

SINT32 MaSound_Restart (SINT32 func_id, SINT32 file_id, void * ext_args);

Description

Cancels pause of sequence data reproduction.
For audio systems, this operation is the same as Start.

Argument

func_id	Function ID of MA Stream Converter to be designated.
file_id	File ID
ext_args	Extension argument specific to each MA Stream Converter. NULL is set when no extension argument is needed.

Returned value

0	Successful
Negative	Error code

3 MA Stream Converter

3.1 Outline

It converts each format so that it can be reproduced.

The converter interprets applicable data format to create packet data by using MA Sound Driver so that the can be reproduced.

The creation processing is performed with stream processing.

The converter provides the following functions so that it can be called from MA Sound Sequencer.

Function name	Description
<i>MaSrmCnv_Initialize</i>	Performs initialization of MA Stream Converter
<i>MaSrmCnv_Load</i>	Stream converter loading processing
<i>MaSrmCnv_Control</i>	Setting control value into stream converter
<i>MaSrmCnv_Open</i>	Stream converter opening processing
<i>MaSrmCnv_Standby</i>	Stream converter standby processing
<i>MaSrmCnv_Seek</i>	Stream converter seeking processing
<i>MaSrmCnv_Start</i>	Stream converter starting processing
<i>MaSrmCnv_Stop</i>	Stream converter stopping processing
<i>MaSrmCnv_Close</i>	Stream converter closing processing
<i>MaSrmCnv_Unload</i>	Stream converter unloading processing
<i>MaSrmCnv_Convert</i>	Performs Sequence data conversion processing
<i>MaSrmCnv_End</i>	Performs MA Stream Converter ending processing

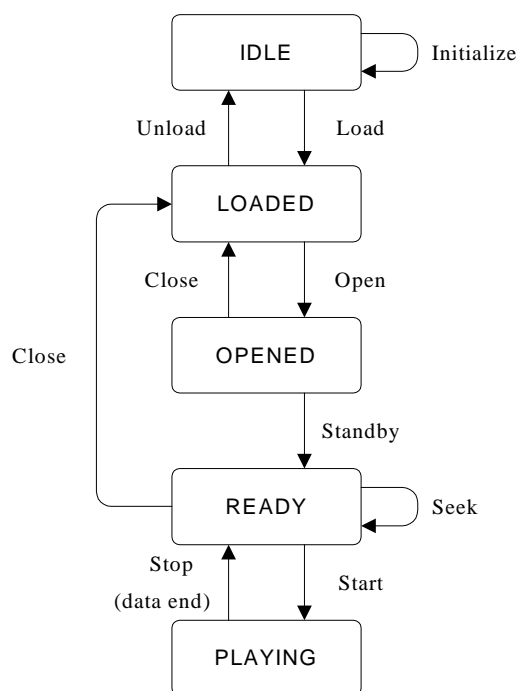
Note: The function names are presented for explanation purpose. Actually, the function names vary among the stream converters.

These functions in the group are to be synchronization type which performs return processing after ending the processing. Each function is to be given a configuration that allows parallel operation with the same function for call with different ID. (It is to be guaranteed that no problem occurs even if the worst comes to the worst. Especially, it is necessary to take care because for Convert function and others, the call is generally performed asynchronously.)

Formats that are supported and name of their Converter functions are as follows:

Format	Function name
SMAF/MA-1/MA-2/MA-3	MaMmfCnv_
SMF format 0 or 1/GM Lite or Level1	MaMidCnv_
Realtime MIDI	MaRmdCnv_
SMAF/Audio	MaAudCnv_
WAV	MaWavCnv_

3.2 Status transition



Status	Meaning
IDLE	Idling
LOADED	File loaded
OPENED	Resources have been secured so that loaded files can be reproduced.
READY	Reproduction can be made immediately.
PLAYING	Reproduction is in progress.

3.3 Stream converter processing procedure

3.3.1 MaSrmCnv_Initialize

This is called from MaSound_Create (). Performs initialization of interior of Stream Converter. When it is necessary to guarantee the resources, secure the necessary resource by using MaSndDrv_Create () at this stage.

3.3.2 MaSrmCnv_Load

Checks the data format that was handed over, and returns an error if the format cannot be handled. When it is specified to perform an error check, check if irregular data is included. Understand necessary hardware resources (channels, FM Voice, WT Voice and Stream Audio).

3.3.3 MaSrmCnv_Open

Secure hardware resources that needed by MaSndDrv_Create (). The processing if hardware resources cannot be secure is to be made in accordance with the specifications of converter. The converter performs address management under secured built-in RAM.

3.3.4 MaSrmCnv_Control

Receives direction needed for conversion. Nonrelevant commands are ignored.

3.3.5 MaSrmCnv_Standby

Prepares data for writing tone parameter or others, and write the data into RAM built in the device through MaDevDrv_SendDirectRamData.

3.3.6 MaSrmCnv_Seek

Performs start position change processing for the designated seek position, and prepares control data or others that require change so that they can be outputted when MaSrmCnv_Convert () is called.

Uses MaSndDrv_ControlSequencer() and controls a sequencer.

3.3.7 MaSrmCnv_Start

Changes internal state so that reproduction of designated sequence data is started. Uses MaSndDrv_ControlSequencer () to perform control of the sequencer.

3.3.8 MaSrmCnv_Stop

Changes internal state so that reproduction of designated sequence data is stopped. Uses MaSndDrv_ControlSequencer () to perform control of the sequencer.

3.3.9 MaSrmCnv_Close

Releases resources that have been secured with MaSndDrv_Free ().

3.3.10 MaSrmCnv_Unload

Releases resources that have been secured by the stream converter itself if they exist.

3.3.11 MaSrmCnv_End

Called from MaSound_Delete (). Releases resources that have been secured until this moment if they exist.

3.3.12 MaSrmCnv_Convert

Decodes sequence data in applicable format, and calls MaSndDrv_SetCommand () of MA Sound Driver. Performs loop processing by using a reproduction mode designated with argument of MaSrmCnv_Start (). Issues applicable number in case of loop or at sequence end by using User Event of MaSndDrv_SetCommand ().

3.4 Function definition

3.4.1 MaSrmCnv_Initialize

SINT32 *MaSrmCnv_Initialize* (void);

Description

Performs initialization of stream converter to place it in the usable state.

Called from MaSndSeq_Create.

For real time system converter and audio system converter that requires guarantee of resources that use real time path, the resources have to be secured at this stage, not at the time of Open.

Argument

None

Returned value

0 Successful.

Negative Error code.

3.4.2 MaSrmCnv_End

SINT32 *MaSrmCnv_End* (void);

Description

Performs ending processing of stream converter.

Called from MaSndSeq_Delete ().

It is necessary to design so that multiple ending processing does not cause problem.

For real time system converter and audio system converter that requires guarantee of resources that use real time path, the resources have to be secured at this stage, not at the time of Close.

Argument

None

Returned value

0 Successful.

Negative Error code.

3.4.3 MaSrmCnv_Load

SINT32 *MaSrmCnv_Load* (UINT8 *file_ptr, UINT32 file_size, UINT8 mode, SINT32 (* func) (UINT8 id), void * ext_args);

Description

Checks format data on the memory to see if any error exists.

At the same time, it makes preparation so that the stream converter can perform processing.

Returns an error if the format data is abnormal. At this time, Unload is not required.

After this, it performs operation by handing over the registration number of format data returned with the Returned value by using Argument.

Format check can be made by using this function if data with other registration number that are in reproduction is within corresponding data count.

Each converter is to have a configuration that allows simultaneous Load of designated number. This is used as a mechanism for allowing format check during reproduction.

Format	Number that can be Loaded simultaneously
SMAF/MA-1	2
SMAF/MA-2	2
SMAF/MA-3	2
SMF/GM Lite	2
Real-Time MIDI/GM Lite	1
SMAF/Audio	2
WAV	2

Argument

file_ptr	Pointer for format data storage domain
file_size	Byte size of format data
mode	Designates whether error check is to be made or not. 0: Check is not performed. 1: Check is performed. 2: Only check is performed. Since no internal information is acquired, Open cannot be performed after this. 3: Performs acquisition of contents information. Since no internal information is acquired, Open cannot be performed after this.
func	Call back function
ext_args	Extension argument specific to each stream converter. NULL is set when no extension argument is needed.

Returned value

Non-negative	Successful. The value shows registration number of format data.
Negative	Error code.

3.4.4 MaSrmCnv_Unload

SINT32 *MaSrmCnv_Unload* (SINT32 file_id, void * ext_args);

Description

Releases format data.

Guarantese that call of Free () during Convert () processing does not cause failure.

Argument

file_id Format data registration number

ext_args Extension argument specific to each stream converter. NULL is set when no extension argument is needed.

Returned value

0 Successful.

Negative Error code.

3.4.5 MaSrmCnv_Open

SINT32 *MaSrmCnv_Open* (SINT32 file_id, UINT16 mode, void * ext_args);

Description

Although a mode is designated, it is identified with a fixed value in each converter, and is not reflected.

The number that can be Opened is up to 4 for Phrase system, and only one for other system.

Argument

file_id Format data registration number

mode Function mode

ext_args Extension argument specific to each stream converter. NULL is set when no extension argument is needed.

Returned value

0 Successful.

Negative Error code.

3.4.6 MaSrmCnv_Close

SINT32 *MaSrmCnv_Close* (SINT32 file_id, void * ext_args);

Description

Releases resources that have been secured.

Call MaSndDrv_Free ().

Argument

file_id Format data registration number

ext_args Extension argument specific to each stream converter. NULL is set when no extension argument is needed.

Returned value

0 Successful.

Negative Error code.

3.4.7 MaSrmCnv_Control

SINT32 *MaSrmCnv_Control* (SINT32 file_id, UINT8 ctrl_num, void * prm, void * ext_args);

Description

Various setup, informational acquisition, etc. are performed to each Stream Converter.

*Note) Since it changes with converters, effective ctrl_num is the thing of reference of the API specification of each converter.

Argument

file_id	Format data registration number
ctrl_num	Designates a content of processing. <ul style="list-style-type: none"> 0: Setting of reproduction volume (0 to 127) Default is 100. 1: Designation of reproduction speed (70 to 100 to 130) Default is 100 2: Designation of relative change of reproduction key (-12 to 0 to +12) Default is 0. 3: Acquisition of difference between basic time setting and actual time 4: Acquisition of reproduction position (unit: ms) 5: Acquisition of reproduction time (unit: ms) 6: Acquisition of present internal status 7: Sends regular MIDI message. (dedicated to Realtime MIDI) 8: Sends SysEx MIDI message. (dedicated to Realtime MIDI) 9: Reserved 10: Acquires designated data of contents information. 11: Acquires Phrase List information. 12: Sets reproduction start point. 13: Sets reproduction end point. 14: Sets panpot value (0 to 127) Default is 64. 15: Acquires LED synch. status. 16: Acquires vibration motor synch. status. 17: Specification of a Note number that a user event is generated. (0 – 127) 18: Acquisition of the sequence time under conversion. (ms Unit) 19: Acquisition of the information for reproduction. 20: Registration of the information for reproduction. 26: Designates the playback speed (25 - 400). Default is 100. 27: Designates the playback count. Default is not set. 29: Acquires the playback volume. 30: Sets the period callback.
prm	Parameter needed for processing designated by ctrl_num
ext_args	Extension argument specific to each stream converter. NULL is set when no extension argument is needed.

Returned value

Non-negative	Successful. When a value is returned, the value.
Negative	Error

3.4.8 MaSrmCnv_GetLength

SINT32 *MaSrmCnv_GetLength* (SINT32 file_id);

Description

Returns reproduction time of registered data.
 However, the time does not include the one of the loop.
 Real-Time system does not support this.

Argument

file_id	Format data registration number
---------	---------------------------------

Returned value

Non-negative	Reproduction time (ms)
Negative	Error

3.4.9 MaSrmCnv_GetPos

SINT32 *MaSrmCnv_GetPos* (SINT32 file_id);

Description

Returns present reproduction position
 Returns stop position when stopped.
 Returns last position when reproduction ends.

Argument

file_id	Format data registration number
---------	---------------------------------

Returned value

Non-negative	Reproduction time (ms)
Negative	Error

3.4.10 MaSrmCnv_SetVolume

SINT32 *MaSrmCnv_SetVolume* (SINT32 file_id, UINT8 vol);

Description

Sets reproduction volume of each sequence. Default is 127.

Argument

file_id	Format data registration number
vol	Volume (0 to 127) Volume designation [dB] = 40 × Log (vol / 127)

Returned value

Non-negative	Successful.
Negative	Error

3.4.11 MaSrmCnv_SetShortMessage

SINT32 *MaSrmCnv_SetShortMessage* (SINT32 file_id, UINT32 message);

Description

Normally, sends MIDI message. Status byte must be attached.
Dedicated to Real-Time MIDI

Argument

file_id	Format data registration number
message	MIDI message. Message is packed from LSB side.

Returned value

0	Successful.
Negative	Error

3.4.12 MaSrmCnv_SetLongMessage

SINT32 *MaSrmCnv_SetLongMessage* (SINT32 file_id, UINT8 * msg_ptr, UINT32 msg_size);

Description

Sends System Exclusive MIDI message.
Dedicated to Real-Time MIDI

Argument

file_id	Format data registration number
msg_ptr	Pointer for System Exclusive MIDI message
msg_size	Byte size of System Exclusive MIDI message

Returned value

0	Successful.
Negative	Error

3.4.13 MaSrmCnv_Standby

SINT32 *MaSrmCnv_Standby* (SINT32 file_id, void * ext_args);

Description

Extracts tone data that are included in the format data of registration number to create data that are placed in the built-in RAM and information table of the RAM to register them.
Volume of all channels is to be set for silence, and panpot is to be set to center.
Since it takes some time (approximately 20 msec) to settle in a volume at a designated position, the converter can perform a processing that performs Wait until the volume settles after the volume is set.

Argument

file_id	Format data registration number
ext_args	Extension argument specific to each stream converter. NULL is set when no extension argument is needed.

Returned value

0	Successful.
Negative	Error code.

3.4.14 MaSrmCnv_Seek

SINT32 *MaSrmCnv_Seek* (SINT32 file_id, UINT32 pos, UINT8 flag, void * ext_args);

Description

Moves sequence reproduction position to immediately before designated time.

Data before start point of data is interpreted that it is located before the time “0”.

Be sure to output message that reflects default set value that is specified for each data format (Be careful also of synchronization of LED/Motor).

Argument

file_id	Format data registration number
pos	Reproduction start position (ms)
flag	0 fixed
ext_args	Extension argument specific to each stream converter. NULL is set when no extension argument is needed.

Returned value

0	Successful.
Negative	Error code.

3.4.15 MaSrmCnv_Start

SINT32 *MaSrmCnv_Start* (SINT32 file_id, void * ext_args);

Description

Performs starting of reproduction from present position.

It is necessary that multiple issue does not cause problem.

Argument

file_id	Format data registration number
ext_args	Extension argument specific to each stream converter. NULL is set when no extension argument is needed.

Returned value

0	Successful.
Negative	Error code.

3.4.16 MaSrmCnv_Stop

SINT32 *MaSrmCnv_Stop* (SINT32 file_id, void * ext_args);

Description

Stops reproduction. The reproduction position is the same as the stop position.

For sound deadening, stop reproduction after issuing AllSoundOff.

It is necessary that multiple issues do not cause problem.

Argument

file_id	Format data registration number
ext_args	Extension argument specific to each stream converter. NULL is set when no extension argument is needed.

Returned value

0	Successful.
Negative	Error code.

3.4.17 MaSrmCnv_Convert

SINT32 MaSrmCnv_Convert (void);

Description

Generates messages by converting data located at present position.

For H/WSequencer system, it issues one intermediate message by converting data located at present position. For S/WSequencer system, it issues an intermediate message if a message that reaches the issue time by converting data located at present position is present. Advance the time by 1 after issuing a message or messages.

Basically, change of status by the control message is not performed during data conversion.

Actually, since one original command does not always become one intermediate code, the converter that performs Convert with QUEUE where the results of conversion is stored first outputs intermediate code located in the QUEUE, and then performs data conversion after the QUEUE becomes empty.

Argument

None

Returned value

Positive	Number of messages issued
0	Successful.
Negative	Error code.

4 MA Sound Driver

MA Sound Driver prepares an interface for performing tone generation processing. The value set for Argument can be used to select to send packet data to the real time write path or to send packet data to delayed write path. However, when sending packet data to delayed write path, it is necessary that the processing of MA Sound Sequencer is performed in accordance with the procedure.

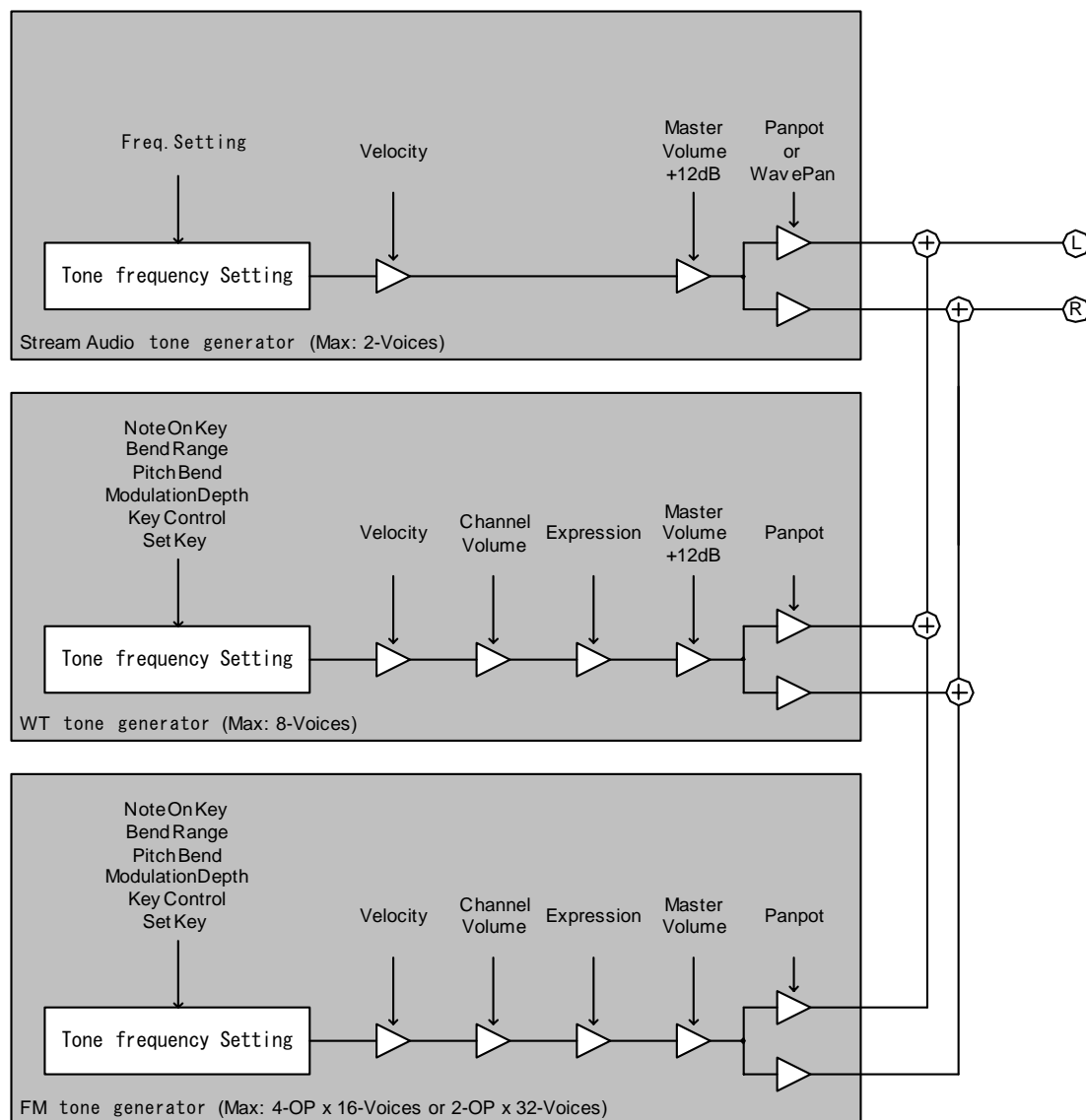
Function name	Description
MaSndDrv_Initialize	Performs initialization of MA Sound Driver.
MaSndDrv_End	Performs ending processing of MA Sound Driver.
MaSndDrv_Create	Secures necessary resources.
MaSndDrv_Free	Releases resources that have been secured.
MaSndDrv_DeviceControl	Performs setting for MA-3 devices.
MaSndDrv_UpdatePos	Performs update of reproduction position information.
MaSndDrv_GetTimeError	Acquires difference between basic time setting and actual value of time.
MaSndDrv_GetPos	Returns present reproduction position.
MaSndDrv_SetSpeed	Designates reproduction speed.
MaSndDrv_SetKey	Designates relative change of reproduction key.
MaSndDrv_SetKeyControl	Sets whether key control is accepted or not.
MaSndDrv_SetFmExtWave	Registers FM extension tone.
MaSndDrv_GetVoice	Returns whether tone is registered.
MaSndDrv_SetVoice	Registers tones.
MaSndDrv_SetStream	Registers stream audio waveform data.
MaSndDrv_ControlSequencer	Performs control of sequencer.
MaSndDrv_SeekControl	Performs seek processing.
MaSndDrv_SetCommand	Issues sequences or events.
MaSndDrv_UpdateSequence	Performs update of sequence data.

Function name	Description
MaSndDrv_Nop	Issues NOP.
MaSndDrv_UserEvent	Issues user events.
MaSndDrv_NoteOn	Issues Note-on of designated channels.
MaSndDrv_NoteOnMa2	Issues compatible NoteOn of designated channel.
MaSndDrv_NoteOnMa2Ex	Issues compatible direct pitch of designated channel.
MaSndDrv_NoteOff	Issues Note-off of designated channels.
MaSndDrv_NoteOffMa2	Issues compatible NoteOff of designated channel.
MaSndDrv_NoteOffMa2Ex	Issues compatible direct pitch of designated channel.
MaSndDrv_ProgramChange	Sets program number of designated channels.
MaSndDrv_ModulationDepth	Sets depth of vibrato of designated channels.
MaSndDrv_ChannelVolume	Sets volume of designated channels.
MaSndDrv_Panpot	Sets panpot value of designated channels.
MaSndDrv_Expression	Sets expression value of designated channels.
MaSndDrv_Hold1	Performs damper setting for designated channels.
MaSndDrv_AllSoundOff	Performs all sound deadening of designated channels.
MaSndDrv_ResetAllControllers	Resets control value of designated channels.
MaSndDrv_AllNoteOff	Performs all Note-off of designated channels.
MaSndDrv_MonoModeOn	Sets designated channels for mono-phonic tone generation.
MaSndDrv_PolyModeOn	Sets designated channels for poly-phonic tone generation.
MaSndDrv_PitchBend	Sets pitch bend value of designated channels.
MaSndDrv_BendRange	Sets maximum change width of pitch bend.
MaSndDrv_StreamOn	Performs start of tone generation of Stream Audio.
MaSndDrv_StreamOff	Performs stop of tone generation of Stream Audio.
MaSndDrv_StreamSlave	Performs slave setting of stream audio.
MaSndDrv_StreamPanpot	Sets panpot value of Stream Audio.
MaSndDrv_MasterVolume	Sets master volume.
MaSndDrv_SystemOn	Performs all initialization.
MaSndDrv_LedOn	Turn LED ON.
MaSndDrv_LedOff	Turn LED OFF.
MaSndDrv_MotorOn	Turn MOTOR ON.
MaSndDrv_MotorOff	Turn MOTOR OFF.
MaSndDrv_StreamSeek	Sets Seek position of Stream Audio.

4.1 Tone generation model

For MA Sound Driver, the two systems, real time system and delayed system, that operate the synthesizer shown below through real time system interface or (simultaneously) through delayed system interface are independent from each other, the amount their resources are limited because they treat one hardware actually.

- Since Stream Audio and WT uses the same tone generator hardware, the tone generations cannot exceed 8 in total.
- Each tone generator shares 8176 bytes hardware built-in RAM as expanded tone area with other one.
- Stream Audio tone generator uses 1024 byte hardware built-in RAM for tone generation.
- Assignment of tone generator is performed dynamically, and is determined by later-arrival- preferentially.
- The rule for tone generation covers three modes; poly-phonic, mono-phonic, and CH – tone generation fixed assignment.
- The destination of output of this tone generation model has Mix (volume adjustment mechanism).
- A tone generator that is used is determined depending on the tone.



4.2 Function definition

4.2.1 MaSndDrv_Initialize

SINT32 MaSndDrv_Initialize (void);

Description

Performs initialization of MA Sound Driver.

Argument

None

Returned value

0 Successful.

Negative Error code.

4.2.2 MaSndDrv_End

SINT32 MaSndDrv_End (void);

Description

Performs ending processing of MA Sound Driver.

Argument

None

Returned value

0 Successful.

Negative Error code.

4.2.3 MaSndDrv_Create

SINT32 MaSndDrv_Create (UINT8 seq_type, UINT8 time, UINT8 cnv_mode, UINT8 res_mode, UINT8 stream, UINT32 (*func) (void), UINT32 * ram_adrs, UINT32 * ram_size);

Description

Secures resources needed by Stream Converter.

Argument

seq_type	Designates a reproduction system (0: Delayed system, 1: Real time system, 2: Audio system)																																
time	Basic time of sequence that is outputted (0, 4 to 10, 20[ms]) When "0" is designatd, the timer for sequence is not secured.																																
cnv_mode	Conversion mode <table border="0"> <tr> <td>bit 0</td><td>Setting of drum tone (0 or 1)</td></tr> <tr> <td>0:</td><td>Standard mode</td></tr> <tr> <td>1:</td><td>MA-1/2 compatible mode</td></tr> <tr> <td>bit 2-1</td><td>Setting of DVA(0 to 2)</td></tr> <tr> <td>0:</td><td>Standard mode</td></tr> <tr> <td>1:</td><td>Simplification mode</td></tr> <tr> <td>2:</td><td>Forced mono mode</td></tr> <tr> <td>bit 3</td><td>Setting of melody tone generation (0 - 1)</td></tr> <tr> <td>0:</td><td>Standard mode</td></tr> <tr> <td>1:</td><td>MA-1/2 compatible mode</td></tr> <tr> <td>bit 4</td><td>FM mode when res_mode = 0 (0 - 1)</td></tr> <tr> <td>0:</td><td>2-OP</td></tr> <tr> <td>1:</td><td>4-OP</td></tr> <tr> <td>bit5</td><td>Type setting of Percussion Sound Set.</td></tr> <tr> <td>0:</td><td>GM-x</td></tr> <tr> <td>1:</td><td>GM Level 1</td></tr> </table>	bit 0	Setting of drum tone (0 or 1)	0:	Standard mode	1:	MA-1/2 compatible mode	bit 2-1	Setting of DVA(0 to 2)	0:	Standard mode	1:	Simplification mode	2:	Forced mono mode	bit 3	Setting of melody tone generation (0 - 1)	0:	Standard mode	1:	MA-1/2 compatible mode	bit 4	FM mode when res_mode = 0 (0 - 1)	0:	2-OP	1:	4-OP	bit5	Type setting of Percussion Sound Set.	0:	GM-x	1:	GM Level 1
bit 0	Setting of drum tone (0 or 1)																																
0:	Standard mode																																
1:	MA-1/2 compatible mode																																
bit 2-1	Setting of DVA(0 to 2)																																
0:	Standard mode																																
1:	Simplification mode																																
2:	Forced mono mode																																
bit 3	Setting of melody tone generation (0 - 1)																																
0:	Standard mode																																
1:	MA-1/2 compatible mode																																
bit 4	FM mode when res_mode = 0 (0 - 1)																																
0:	2-OP																																
1:	4-OP																																
bit5	Type setting of Percussion Sound Set.																																
0:	GM-x																																
1:	GM Level 1																																
res_mode	Resource mode (0 or 1 to 16)																																
stream	No. of streams (0 to 2)																																
func	Pointer for data supply function																																
ram_adrs	Head address of built-in RAM that is secured.																																
ram_size	Byte size of built-in RAM that is secured.																																

Returned value

Non-negative	Successful. Shows registration number.
Negative	Error code.

4.2.4 MaSndDrv_Free

SINT32 MaSndDrv_Free (SINT32 seq_id);

Description

Releases resources that have been secured.

Argument

seq_id	Sequence registration number
--------	------------------------------

Returned value

0	Successful.
Negative	Error code.

4.2.5 MaSndDrv_DeviceControl

SINT32 MaSndDrv_DeviceControl (UINT8 cmd, UINT8 param1, UINT8 param2, UINT8 param3);

Description

Performs various settings of device.

- When cmd is in the range from 0 to 4, the Argument is handed over to MaDevDrv_DeviceControl () to be set as it is.
- When cmd is in the range from 5 to 7, LED of MA-3 intermediate register that sets the source of control of LED and LED bit of the control register are set as follows.

cmd	param 1											
5	0				1				2			
6	0		1 - 5		0		1 - 5		0		1 - 5	
7	0	1	0	1	0	1	0	1	0	1	0	1
MODE	0				4	5	4	5	4	5	4	5
FREQ	-		0 - 4		-		0 - 4		-		0 - 4	
LED	0				0				1			
LED bit	-				1				0			

- When cmd is in the range from 8 to 10, MTR of MA-3 intermediate register that sets the source of control of Motor and MTR bit of the control register are set as follows.

cmd	param 1											
8	0				1				2			
9	0		1 - 5		0		1 - 5		0		1 - 5	
10	0	1	0	1	0	1	0	1	0	1	0	1
MODE	0				4	5	4	5	4	5	4	5
FREQ	-		0 - 4		-		0 - 4		-		0 - 4	
LED	0				0				1			
LED bit	-				1				0			

Argument

cmd Command number
param1 Parameter 1
param2 Parameter 2
param3 Parameter 3

Setting	cmd	param1	param2	param3
Power management (digital)	0	val	0	0
Power management (analog)	1	val	0	0
EQ Volume	2	eq_vol	0	0
HP Volume	3	mono	vol_l	vol_r
SP Volume	4	hp_vol	0	0
Source of control of LED	5	source	0	0
On/off of LED	6	val	0	0
Forced control of LED	7	sw	0	0
Source of control of motor	8	source	0	0
On/off of motor	9	val	0	0
Forced control of motor	10	sw	0	0
Acquisition of PLL value	11	0	0	0
Acquisition of sequencer processing flag	12	0	0	0
Acquisition of LED control source info	15	0	0	0
Acquisition of VIB control source info	16	0	0	0

Returned value

Non-negative Successful. When a value is returned, the value.
Negative Error code.

4.2.6 MaSndDrv_GetTimeError

SINT32 MaSndDrv_GetTimeError (SINT32 seq_id);

Description

Acquires difference between basic time setting and actual value of time.
In real time setting, "0" is returned all the time.

Argument

seq_id	Sequence registration number
--------	------------------------------

Returned value

Positive	Shows difference between time setting and actual value of time. (1.0 = 10000h) The time difference includes settings for original sender and PLL. To be > 1.0 when actual value is large.
0	Successful.
Negative	Error code.

4.2.7 MaSndDrv_GetPos

SINT32 MaSndDrv_GetPos (SINT32 seq_id);

Description

Returns present reproduction position
Returns stop position when stopped.
Last position when reproduction is ended.

Argument

seq_id	Sequence registration number
--------	------------------------------

Returned value

Non-negative	Reproduction time (ms)
Negative	Error

4.2.8 MaSndDrv_SetSpeed

SINT32 MaSndDrv_SetSpeed (SINT32 seq_id, UINT8 val);

Description

Sets reproduction speed.
Real time setting is invalid under the existing circumstances, and thus, Error occurs at all times.

Argument

seq_id	Sequence registration number
val	Reproduction speed [%] (70 to 130). 100 gives original speed (x1). Default is 100[%]

Returned value

0	Successful.
Negative	Error code.

4.2.9 MaSndDrv_SetKey

SINT32 MaSndDrv_SetKey (SINT32 seq_id, SINT8 val);

Description

Sets relative conversion of reproduction key in 100 [cents] increment.
This is applicable only to melody tones.
This can be changed even in reproduction (Next tone generation and after become valid.)
Default is 0

Argument

seq_id	Sequence registration number
val	Relative change of reproduction key (-12 to +12)

Returned value

0	Successful.
Negative	Error code.

4.2.10 MaSndDrv_SetKeyControl

SINT32 MaSndDrv_SetKeyControl (SINT32 seq_id, UINT8 ch, UINT8 val);

Description

Sets whether key control is to be accepted or not.
“0” means normal rule (that is, key control is valid for melody tone, and is invalid for drum tone).
“1” means that key control is invalid on designated channels.
“2” means that key control is valid.

Argument

seq_id	Sequence registration number
ch	Channel number (0 to 15)
val	Setting of key control (0 to 2)

0:	None
1:	OFF
2:	ON

Returned value

0	Successful.
Negative	Error code.

4.2.11 MaSndDrv_SetFmExtWave

SINT32 MaSndDrv_SetFmExtWave (SINT32 seq_id, UINT8 wave_no, UINT32 ram_adrs);

Description

Performs registration of FM extension waveform.

Argument

seq_id	Sequence registration number
wave_no	FM extension waveform number (15, 23, 31)
ram_adrs	Address of built-in RAM in which FM extension waveform is placed.

Returned value

0	Successful.
Negative	Error code.

4.2.12 MaSndDrv_SetVolume

SINT32 MaSndDrv_SetVolume(SINT32 seq_id, UINT8 volume);

Description

A volume setup for every system is performed.
A default is 100.

Argument

seq_id	Sequence registration number.
volume	Volume (0 to 127). Volume specification [dB] = 40 x Log (volume / 127). A default is 100.

Returned value

0	Successful.
Negative	Error code.

4.2.13 MaSndDrv_SetVoice

SINT32 MaSndDrv_SetVoice (SINT32 seq_id, UINT8 bank_no, UINT8 prog_no, UINT8 synth, UINT32 key, UINT32 ram_adrs);

Description

Performs registration of FM and WT extension tones
Up to 254 tones can be registered for each seq_id independently.
The registration table is initialized at MaSndDrv_Create (). (All registrations are erased.)
When FFFFFFFFh is designated for ram_adrs of Argument, it means deletion of registration. In this case, designation of only seq_id, bank_no and prog_no is acceptable.

Argument

seq_id	Sequence registration number
bank_no	Bank number (0, 1 to 127: Melody tones, 128, 129 to 255: Drum tones)
prog_no	Program (key) number (0 to 127) For melody tones, program number is set. For drum tones, key number is set.
synth	Synthesizer mode (1: FM, 2: WT)
key	For FM synthesizer, this designates tone generation interval at drum by using key number at MIDI (0 to 127). WT is Sampling frequency value. (0 to 48000 [Hz])
ram_adrs	Address of built-in RAM in which tones are placed.

Returned value

0	Successful.
Negative	Error code.

4.2.14 MaSndDrv_GetVoice

SINT32 MaSndDrv_GetVoice(SINT32 seq_id, UINT8 bank_no, UINT8 program_no);

Description

Returns whether tones are registered or not.

Argument

seq_id	Sequence registration number
bank_no	Bank no. (0, 1 to 127: Melody tones, 128, 129 to 255: Drum tones)
program_no	Program (key) number (0 to 127)

Returned value

1	Registered
0	Not registered
Negative	Error code

4.2.15 MaSndDrv_SetStream

SINT32 MaSndDrv_SetStream (SINT32 seq_id, UINT8 wave_id, UINT8 format, UINT32 frequency, UINT8 * wave_ptr, UINT32 wave_size);

Description

Performs registration of stream audio waveform data.

Up to 32 waveforms can be registered per each sequence registration.

When “0” is specified for the frequency of Argument, this means deletion of registration. In this case, it is acceptable when valid values are set only for seq_id and wave_id.

For MA-2 ADPCM, Pan is fixed to Center 0dB.

Argument

seq_id	Sequence registration number
wave_id	Waveform number (0 to 63)
format	Waveform format
	0: MA-2 ADPCM
	1: ADPCM
	2: 8-bit offset binary
	3: 8-bit 2's comp binary
frequency	Sampling frequency (0 or 4000 to 48000[Hz])
wave_ptr	Pointer for the place where waveform data are stored.
wave_size	Byte size of waveform data

Returned value

0	Successful.
Negative	Error code.

4.2.16 MaSndDrv_SetCommand

SINT32 MaSndDrv_SetCommand (SINT32 seq_id, UINT32 delta_time, UINT32 cmd, UINT32 param1, UINT32 param2, UINT32 param3);

Description

Issues sequences or events.

When ch that is not secured with MaSndDrv_Create is designated, the command is ignored.

Argument

seq_id Registration number

delta_time Time of issue [tick]

cmd Command number

Message	cmd	param1	param2	param3
Note On	0	ch	key	velocity
Note OnMa2	1	ch	key	velocity
Note OnMa2Ex	2	ch	key	velocity
Note Off	3	ch	key	0
Note OffMa2	4	ch	key	0
Note OffMa2Ex	5	ch	key	0
Program Change	6	ch	bank	prog
Modulation Depth	7	ch	val	0
Channel Volume	8	ch	val	0
Panpot	9	ch	val	0
Expression	10	ch	val	0
Hold 1	11	ch	val	0
All Sound Off	12	ch	0	0
Reset All Controllers	13	ch	0	0
All Note Off	14	ch	0	0
Mono Mode On	15	ch	0	0
Poly Mode On	16	ch	0	0
Pitch Bend	17	ch	val	0
Bend Range	18	ch	range	0
Stream On	19	ch	wav	velocity
Stream Off	20	ch	0	0
Stream Slave	21	wav	val	0
Stream Panpot	22	wav	val	0
Master Volume	23	val	0	0
System On	24	0	0	0
Led On	25	0	0	0
Led Off	26	0	0	0
Motor On	27	0	0	0
Motor Off	28	0	0	0
User Event	29	val	0	0
NOP	30	0	0	0
StreamSeek	31	0	wav	pos

param1 Parameter 1

param2 Parameter 2

param3 Parameter 3

Returned value

0 Successful.

Negative Error

4.2.17 MaSndDrv_Nop

```
static SINT32 MaSndDrv_Nop ( SINT32 seq_id, SINT32 delta_time, UINT32 type, UINT32 p2, UINT32 p3 );
```

Description

Issues NOP.

Argument

seq_id	Sequence registration number	
delta_time	Time of issue [tick]	Designates "0" or over for Delayed system. Designates Negative for Real time system.
type	Type of NOP (0: NOP_1, 1: NOP_2)	
p2	Unused: Set "0".	
p3	Unused: Set "0".	

Returned value

0	Successful.
Negative	Error code.

4.2.18 MaSndDrv_UserEvent

```
static SINT32 MaSndDrv_UserEvent ( SINT32 seq_id, SINT32 delta_time, UINT32 val, UINT32 p2, UINT32 p3 );
```

Description

Issues user events.

Argument

seq_id	Sequence registration number	
delta_time	Time of issue [tick]	Designates "0" or over for Delayed system. Designates Negative for Real time system.
val	Event number (0 to 127)	
p2	Unused: Set "0".	
p3	Unused: Set "0".	

Returned value

0	Successful.
Negative	Error code.

4.2.19 MaSndDrv_NoteOn

```
static SINT32 MaSndDrv_NoteOn ( SINT32 seq_id, SINT32 delta_time, UINT32 ch, UINT32 key, UINT32 velocity );
```

Description

Issues Note-on of designated channels.

Argument

seq_id	Sequence registration number	
delta_time	Time of issue [tick]	Designates "0" or over for Delayed system. Designates Negative for Real time system.
ch	Channel number (0 to 15)	
key	Tone generation key (0 to 127)	
velocity	Velocity (0 to 127)	
	Designation of volume [dB] = 40 × Log (Velocity / 127)	

Returned value

0	Successful.
Negative	Error code.

4.2.20 MaSndDrv_NoteOnMa2

```
static SINT32 MaSndDrv_NoteOnMa2( SINT32 seq_id, SINT32 delta_time, UINT32 ch, UINT32 key, UINT32 velocity );
```

Description

Issues Note-on of designated channels.

For processing MA-1/2 compatible

Argument

seq_id	Sequence registration number	
delta_time	Time of issue [tick]	Designates "0" or over for Delayed system. Designates Negative for Real time system.
ch	Channel number (0 to 15)	
key	Tone generation key (0 to 127)	
velocity	Velocity (0 to 127)	
	Designation of volume [dB] = $40 \times \text{Log} (\text{Velocity} / 127)$	

Returned value

0	Successful.
Negative	Error code.

4.2.21 MaSndDrv_NoteOnMa2Ex

```
static SINT32 MaSndDrv_NoteOnMa2Ex( SINT32 seq_id, SINT32 delta_time, UINT32 ch, UINT32 key, UINT32 velocity );
```

Description

Issues Note-on of designated channels.

For dealing with simple pitch bend message of MA-2

Argument

seq_id	Sequence registration number	
delta_time	Time of issue [tick]	Designates "0" or over for Delayed system. Designates Negative for Real time system.
ch	Channel number (0 to 15)	
key	Tone generation key (0x80000000 +BLOCK:FNUM set value of MA-2) [BLOCK (0 to 7) << 10] + FNUM (0~1023)	
velocity	Velocity (0 to 127)	
	Designation of volume [dB] = $40 \times \text{Log} (\text{Velocity} / 127)$	

Returned value

0	Successful.
Negative	Error code.

4.2.22 MaSndDrv_NoteOff

```
static SINT32 MaSndDrv_NoteOff ( SINT32 seq_id, SINT32 delta_time, UINT32 ch, UINT32 key, UINT32 p3 );
```

Description

Issues Note-off of designated channels.

Argument

seq_id	Sequence registration number	
delta_time	Time of issue [tick]	Designates "0" or over for Delayed system. Designates Negative for Real time system.
ch	Channel number (0 to 15)	
key	Tone generation key (0 to 12)	
p3	Unused: Set "0".	

Returned value

0	Successful.
Negative	Error code.

4.2.23 MaSndDrv_NoteOffMa2

```
static SINT32 MaSndDrv_NoteOffMa2( SINT32 seq_id, SINT32 delta_time, UINT32 ch, UINT32 key, UINT32 p3 );
```

Description

Issues Note-off of designated channels.

For MA-1/2 compatible

Argument

seq_id	Sequence registration number	
delta_time	Time of issue [tick]	Designates "0" or over for Delayed system. Designates Negative for Real time system.
Ch	Channel number (0 to 15)	
key	Tone generation key (0 to 127)。	
p3	Unused: Set "0".	

Returned value

0	Successful.
Negative	Error code.

4.2.24 MaSndDrv_NoteOffMa2Ex

```
static SINT32 MaSndDrv_NoteOffMa2Ex( SINT32 seq_id, SINT32 delta_time, UINT32 ch, UINT32 key, UINT32 p3 );
```

Description

Issues Note-off of designated channels.

For dealing with simple pitch bend message of MA-2

Argument

seq_id	Sequence registration number	
delta_time	Time of issue [tick]	Designates "0" or over for Delayed system. Designates Negative for Real time system.
Ch	Channel number (0 to 15)	
key	Tone generation key (0x80000000 + BLOCK:FNUM set value of MA-2)	
p3	Unused: Set "0".	

Returned value

0	Successful.
Negative	Error code.

4.2.25 MaSndDrv_ProgramChange

```
static SINT32 MaSndDrv_ProgramChange ( SINT32 seq_id, SINT32 delta_time, UINT32 ch, UINT32 bank_no, UINT32
program_no );
```

Description

Sets bank number and program number of designated channels.

Argument

seq_id	Sequence registration number	
delta_time	Time of issue [tick]	Designates "0" or over for Delayed system. Designates Negative for Real time system.
ch	Channel number	0 to 15
bank_no	bank number	0 Default melody tones 1 to 127 Registered melody tones 128 Default drum tones 129 to 255 Registered drum tones
program_no	program number	0 to 127

Returned value

0	Successful.
Negative	Error code.

4.2.26 MaSndDrv_ModulationDepth

```
static SINT32 MaSndDrv_ModulationDepth ( SINT32 seq_id, SINT32 delta_time, UINT32 ch, UINT32 depth, UINT32 p3 );
```

Description

Changes depth of vibrato of designated channels.

Depth of vibrato depends on the tones. Default is OFF.

Argument

seq_id	Sequence registration number
delta_time	Time of issue [tick] Designates "0" or over for Delayed system. Designates Negative for Real time system.
ch	Channel number (0 to 15)
depth	Depth of vibrato (0: OFF, 1: x 1, 2: x 2, 3: x 4, 4: x 8)
p3	Unused: Set "0".

Returned value

0	Successful.
Negative	Error code.

4.2.27 MaSndDrv_ChannelVolume

```
static SINT32 MaSndDrv_ChannelVolume ( SINT32 seq_id, SINT32 delta_time, UINT32 ch, UINT32 vol, UINT32 p3 );
```

Description

Sets volume of designated channels.

Argument

seq_id	Sequence registration number
delta_time	Time of issue [tick] Designates "0" or over for Delayed system. Designates Negative for Real time system.
ch	Channel number (0 to 15)
vol	Value of volume (0 to 127)
	Designation of volume [dB] = $40 \times \text{Log} (\text{vol} / 127)$ Default is vol = 100
p3	Unused: Set "0".

Returned value

0	Successful.
Negative	Error code.

4.2.28 MaSndDrv_Panpot

```
static SINT32 MaSndDrv_Panpot ( SINT32 seq_id, SINT32 delta_time, UINT32 ch, UINT32 pan, UINT32 p3 );
```

Description

Sets panpot of designated channels.
Default is pan = 64

Argument

seq_id	Sequence registration number
delta_time	Time of issue [tick] Designates "0" or over for Delayed system. Designates Negative for Real time system.
ch	Channel number (0 to 15)
pan	Value of panpot (0 to 127)
p3	Unused: Set "0".

Returned value

0	Successful.
Negative	Error code.

4.2.29 MaSndDrv_Expression

```
static UINT32 MaSndDrv_Expression ( SINT32 seq_id, SINT32 delta_time, UINT32 ch, UINT32 vol, UINT32 p3 );
```

Description

Sets the value of expression of designated channels.

Argument

seq_id	Sequence registration number
delta_time	Time of issue [tick] Designates "0" or over for Delayed system. Designates Negative for Real time system.
ch	Channel number (0 to 15)
vol	Value of expression (0 to 127)
	Designation of volume [dB] = $40 \times \text{Log} (\text{volume} / 127)$ Default is volume = 127
p3	Unused: Set "0".

Returned value

0	Successful.
Negative	Error code.

4.2.30 MaSndDrv_Hold1

```
static SINT32 MaSndDrv_Hold1 ( SINT32 seq_id, SINT32 delta_time, UINT32 ch, UINT32 val, UINT32 p3 );
```

Description

Performs damper setting of designated channels.

Whether damper setting is valid or not depends on the tones.

Argument

seq_id	Sequence registration number
delta_time	Time of issue [tick] Designates "0" or over for Delayed system. Designates Negative for Real time system.
ch	Channel number (0 to 15)
val	Damper setting (0: OFF, 1: ON)
p3	Unused: Set "0".

Returned value

0	Successful.
Negative	Error code.

4.2.31 MaSndDrv_AllSoundOff

```
static SINT32 MaSndDrv_AllSoundOff ( SINT32 seq_id, SINT32 delta_time, UINT32 ch, UINT32 p2, UINT32 p3 );
```

Description

Deadens the sound of all voices of designated channels.

Argument

seq_id	Sequence registration number
delta_time	Time of issue [tick] Designates "0" or over for Delayed system. Designates Negative for Real time system.
ch	Channel number (0 to 15)
p2	Unused: Set "0".
p3	Unused: Set "0".

Returned value

0	Successful.
Negative	Error code.

4.2.32 MaSndDrv_ResetAllControllers

```
static SINT32 MaSndDrv_ResetAllControllers ( SINT32 seq_id, SINT32 delta_time, UINT32 ch, UINT32 p2, UINT32 p3 );
```

Description

Resets control value of designated channels.

Modulation: OFF

Expression: 127

Damper: OFF

Pitch bend: 1.0

Argument

seq_id	Sequence registration number
delta_time	Time of issue [tick] Designates "0" or over for Delayed system. Designates Negative for Real time system.
ch	Channel number (0 to 15)
p2	Unused: Set "0".
p3	Unused: Set "0".

Returned value

0	Successful.
Negative	Error code.

4.2.33 MaSndDrv_AllNoteOff

```
static SINT32 MaSndDrv_AllNoteOff ( SINT32 seq_id, SINT32 delta_time, UINT32 ch, UINT32 p2, UINT32 p3 );
```

Description

Sets all voices of designated channels to Note-off.

Argument

seq_id	Sequence registration number
delta_time	Time of issue [tick] Designates "0" or over for Delayed system. Designates Negative for Real time system.
ch	Channel number (0 to 15)
p2	Unused: Set "0".
p3	Unused: Set "0".

Returned value

0	Successful.
Negative	Error code.

4.2.34 MaSndDrv_MonoModeOn

```
static SINT32 MaSndDrv_MonoModeOn ( SINT32 seq_id, SINT32 delta_time, UINT32 ch, UINT32 p2, UINT32 p3 );
```

Description

Sets designated channels to mono-phonetic tone generation.

Argument

seq_id	Sequence registration number
delta_time	Time of issue [tick] Designates "0" or over for Delayed system. Designates Negative for Real time system.
ch	Channel number (0 to 15)
p2	Unused: Set "0".
p3	Unused: Set "0".

Returned value

0	Successful.
Negative	Error code.

4.2.35 MaSndDrv_PolyModeOn

```
static SINT32 MaSndDrv_PolyModeOn ( SINT32 seq_id, SINT32 delta_time, UINT32 ch, UINT32 p2, UINT32 p3 );
```

Description

Sets designated channels to poly-phonictone generation.

Argument

seq_id	Sequence registration number
delta_time	Time of issue [tick] Designates "0" or over for Delayed system. Designates Negative for Real time system.
ch	Channel number (0 to 15)
p2	Unused: Set "0".
p3	Unused: Set "0".

Returned value

0	Successful.
Negative	Error code.

4.2.36 MaSndDrv_PitchBend

```
static SINT32 MaSndDrv_PitchBend ( SINT32 seq_id, SINT32 delta_time, UINT32 ch, UINT32 bend, UINT32 p3 );
```

Description

Sets pitch bend value of designated channels.
This is a cent-linear curve, and maximum width of change "max" is set by using BendRange message.
Default is 0 [cents]

Argument

seq_id	Sequence registration number
delta_time	Time of issue [tick] Designates "0" or over for Delayed system. Designates Negative for Real time system.
ch	Channel number (0 to 15)
bend	Amount of pitch bend (0 to 2000h to 3FFFh) 0 -max [cents] 2000h 0 (Default) 3FFFh + max [cents]
p3	Unused: Set "0".

Returned value

0	Successful.
Negative	Error code.

4.2.37 MaSndDrv_BendRange

SINT32 MaSndDrv_BendRange (SINT32 seq_id, SINT32 delta_time, UINT32 ch, UINT32 range, UINT32 p3);

Description

Set maximum width of change of pitch bend by 100 [cents].
 The pitch bend changes in the range from -max [cents] to +max [cents].
 This becomes valid from the next pitch bend message.
 Default is 200 [cents]

Argument

seq_id	Sequence registration number	
delta_time	Time of issue [tick]	Designates "0" or over for Delayed system. Designates Negative for Real time system.
ch	Channel number (0 to 15)	
range	Bend range (0 to 24)	
p3	Unused: Set "0".	

Returned value

0	Successful.
Negative	Error code.

4.2.38 MaSndDrv_StreamOn

SINT32 MaSndDrv_StreamOn (SINT32 seq_id, SINT32 delta_time, UINT32 ch, UINT32 wave_id, UINT32 velocity);

Description

Performs starting of tone generation of Stream Audio.
 Performs setting of other than KeyOn of applicable WT slot, writing of waveform number into RAM of SoftInt, and issue of SoftInt.

Argument

seq_id	Sequence registration number	
delta_time	Time of issue [tick]	Designates "0" or over for Delayed system. Designates Negative for Real time system.
ch	Channel number (0 to 15)	
wave_id	Waveform registration number (0 to 31)	
velocity	Velocity (0 to 127)	

Returned value

0	Successful.
Negative	Error code.

4.2.39 MaSndDrv_StreamOff

SINT32 MaSndDrv_StreamOff (SINT32 seq_id, SINT32 delta_time, UINT32 ch, UINT32 wave_id, UINT32 p3);

Description

Performs stopping of tone generation of Stream Audio.

Performs writing of Waveform registration number into RAM of SoftInt, and issue of SoftInt.

Argument

seq_id	Sequence registration number	
delta_time	Time of issue [tick]	Designates "0" or over for Delayed system. Designates Negative for Real time system.
ch	Channel number (0 to 15)	
wave_id	Waveform registration number (0 to 31)	
p3	Unused: Set "0".	

Returned value

0	Successful.
Negative	Error code.

4.2.40 MaSndDrv_StreamSlave

SINT32 MaSndDrv_StreamSlave(SINT32 seq_id, SINT32 delta_time, UINT32 ch, UINT32 wave_id, UINT32 p3);

Description

Sets slave of stream audio.

Performs setting of other than KeyOn of applicable WT slot.

Argument

seq_id	Sequence registration number	
delta_time	Time of issue [tick]	Designates "0" or over for Delayed system. Designates Negative for Real time system.
ch	Channel number (0 to 15)	
wave_id	Waveform registration number (0 to 31)	
velocity	Velocity (0 to 127)	

Returned value

0	Successful.
Negative	Error code.

4.2.41 MaSndDrv_StreamPanpot

SINT32 MaSndDrv_StreamPanpot (SINT32 seq_id, SINT32 delta_time, UINT32 wav_id, UINT32 pan, UINT32 p3);

Description

Sets panpot value specific to Stream Audio waveform data.

Argument

seq_id	Sequence registration number	
delta_time	Time of issue [tick]	Designates "0" or over for Delayed system. Designates Negative for Real time system.
wav_id	Waveform registration number (0 to 31)	
pan	Value of panpot (0 to 127, 128: Center 0 dB, 255: Channel panpot setting)	
p3	Unused: Set "0".	

Returned value

0	Successful.
Negative	Error code.

4.2.42 MaSndDrv_MasterVolume

```
static SINT32 MaSndDrv_MasterVolume ( SINT32 seq_id, SINT32 delta_time, UINT32 val, UINT32 p2, UINT32 p3 );
```

Description

Sets master volume.

Default is 45 (-18dB)

Argument

seq_id	Sequence registration number
delta_time	Time of issue [tick] Designates "0" or over for Delayed system. Designates Negative for Real time system.
val	Volume (0 to 127) Designation of volume [dB] = $40 \times \text{Log} (\text{val} / 127)$ Default is val = 45
p2	Unused: Set "0".
p3	Unused: Set "0".

Returned value

0	Successful.
Negative	Error code.

4.2.43 MaSndDrv_SystemOn

```
static SINT32 MaSndDrv_SystemOn ( SINT32 seq_id, SINT32 delta_time, UINT32 p1, UINT32 p2, UINT32 p3 );
```

Description

Performs initialization of control value.

Only the assigned channels are covered.

Program Number	0	
Bank Number	0 (CH=9: 128)	
Poly	1	poly mode
Channel Volume	100	-18dB
Panpot	64	Center
Modulation Depth	0	OFF
Expression	127	max
Hold1	0	OFF
Pitch Bend	1024	1.0
Bend Range	2	
Stream Panpot		
LED		OFF
Motor		OFF

Argument

seq_id	Sequence registration number
delta_time	Time of issue [tick] Designates "0" or over for Delayed system. Designates Negative for Real time system.
p1	Unused: Set "0".
p2	Unused: Set "0".
p3	Unused: Set "0".

Returned value

0	Successful.
Negative	Error code.

4.2.44 MaSndDrv_LedOn

```
static SINT32 MaSndDrv_LedOn ( SINT32 seq_id, SINT32 delta_time, UINT32 p1, UINT32 p2, UINT32 p3 );
```

Description

Turns ON LED.

However, whether LED is actually turned on or not, and operation at ON depends on the setting by MaSound_DeviceControl function.

Argument

seq_id	Sequence registration number	
delta_time	Time of issue [tick]	Designates "0" or over for Delayed system. Designates Negative for Real time system.
p1	Unused: Set "0".	
p2	Unused: Set "0".	
p3	Unused: Set "0".	

Returned value

0	Successful.
Negative	Error code.

4.2.45 MaSndDrv_LedOff

```
static SINT32 MaSndDrv_LedOff ( SINT32 seq_id, SINT32 delta_time, UINT32 p1, UINT32 p2, UINT32 p3 );
```

Description

Turns OFF LED.

However, whether LED is actually turned off or not depends on the setting by MaSound_DeviceControl function.

Argument

seq_id	Sequence registration number	
delta_time	Time of issue [tick]	Designates "0" or over for Delayed system. Designates Negative for Real time system.
p1	Unused: Set "0".	
p2	Unused: Set "0".	
p3	Unused: Set "0".	

Returned value

0	Successful.
Negative	Error code.

4.2.46 MaSndDrv_MotorOn

```
static SINT32 MaSndDrv_MotorOn ( SINT32 seq_id, SINT32 delta_time, UINT32 p1, UINT32 p2, UINT32 p3 );
```

Description

Turns ON motor for vibration.

However, whether the motor is actually turned on or not, and operation at ON depends on the setting by MaSound_DeviceControl function.

Argument

seq_id	Sequence registration number	
delta_time	Time of issue [tick]	Designates "0" or over for Delayed system. Designates Negative for Real time system.
p1	Unused: Set "0".	
p2	Unused: Set "0".	
p3	Unused: Set "0".	

Returned value

0	Successful.
Negative	Error code.

4.2.47 MaSndDrv_MotorOff

```
static SINT32 MaSndDrv_MotorOff ( SINT32 seq_id, SINT32 delta_time, UINT32 p1, UINT32 p2, UINT32 p3 );
```

Description

Turns OFF motor for vibration.

However, whether the motor is actually turned off or not depends on the setting by MaSound_DeviceControl function.

Argument

seq_id	Sequence registration number	
delta_time	Time of issue [tick]	Designates "0" or over for Delayed system. Designates Negative for Real time system.
p1	Unused: Set "0".	
p2	Unused: Set "0".	
p3	Unused: Set "0".	

Returned value

0	Successful.
Negative	Error code.

4.2.48 MaSndDrv_StreamSeek

static SINT32 MaSndDrv_StreamSeek(SINT32 seq_id, SINT32 delta_time, UINT32 ch, UINT32 wave_id, UINT32 seek_pos);

Description

Designates the Seek position of Stream Audio.

Argument

seq_id	Sequence registration number	
delta_time	Time of issue [tick]	Designates 0 or over for Delayed system. Designates negative for Real time system.
ch	Unused (Designates 0)	
wave_id	Waveform registration number (0 to 31)	
seek_pos	Seek position (Byte Offset)	

Returned value

0	Successful.
Negative	Error code.

5 MA Resource Manager

MA Resource Manager is a module that manages hardware resources.

Function name	Description
MaResMgr_Initialize	Initializes MA Resource Manager.
MaResMgr_GetResourceInfo	Returns information of hardware resources.
MaResMgr_GetDefWaveAddress	Returns address of waveform data.
MaResMgr_GetDefVoiceAddress	Returns address of tones.
MaResMgr_GetDefVoiceSynth	Returns type of synthesizer of tones.
MaResMgr_GetDefVoiceKey	Returns tone generation key of drum tones.
MaResMgr_RegStreamAudio	Registers data of Stream Audio.
MaResMgr_DelStreamAudio	Cancels registration of data of Stream Audio.
MaResMgr_GetStreamAudioInfo	Acquires information of waveform data of Stream Audio.
MaResMgr_AllocStreamAudio	Secures Stream Audio.
MaResMgr_FreeStreamAudio	Releases Stream Audio.
MaResMgr_AllocRam	Secures built-in RAM block.
MaResMgr_FreeRam	Releases built-in RAM block.
MaResMgr_AllocCh	Secures tone generation channel.
MaResMgr_FreeCh	Releases tone generation channel.
MaResMgr_AllocFmVoice	Secures FM Voice.
MaResMgr_FreeFmVoice	Releases FM Voice.
MaResMgr_AllocWtVoice	Secures WT Voice.
MaResMgr_FreeWtVoice	Releases WT Voice.
MaResMgr_AllocLed	Secures LED.
MaResMgr_FreeLed	Releases LED.
MaResMgr_AllocMotor	Secures Motor.
MaResMgr_FreeMotor	Releases Motor.
MaResMgr_AllocSequencer	Secures sequencer and performs the setting.
MaResMgr_FreeSequencer	Releases sequencer.
MaResMgr_AllocTimer	Secures Timer, and performs the setting.
MaResMgr_FreeTimer	Releases Timer.
MaResMgr_AllocSoftInt	Secures Software Interrupt.
MaResMgr_FreeSoftInt	Releases Software Interrupt.
MaResMgr_SetStreamSeekPos	Sets Seek position of Stream Audio.

5.1 Function definition

5.1.1 MaResMgr_Initialize

SINT32 MaResMgr_Initialize (void);

Description

Initializes MA Resource Manager. Initializes internal management information.

Argument

None

Returned value

0	Successful.
Negative	Error code.

5.1.2 MaResMgr_GetResourceInfo

PMA_RESOURCE_INFO MaResMgr_GetResourceInfo (void);

Description

Acquires the situation of use of hardware resources.

Writing into hardware resources information structural body that was acquired is prohibited.

Argument

None

Returned value

Pointer for resource information structural body.

5.1.3 MaResMgr_GetDefWaveAddress

SINT32 MaResMgr_GetDefWaveAddress (UINT8 wave_id);

Description

Returns address of built-in waveform data of designated registration number.

Argument

wave_id	Waveform number (0 to 6)
---------	--------------------------

Returned value

Positive	Successful. ROM address
0	No registration
Negative	Error code.

5.1.4 MaResMgr_GetDefVoiceAddress

SINT32 MaResMgr_GetDefVoiceAddress (UINT8 prog);

Description

Returns address of tones with designated tone registration number.

Argument

prog	Program number (0 to 127: melody tones, 128 to 255: drum tones)
------	---

Returned value

Positive	Successful.	Built-in ROM address
0	Tones are not registered.	
Negative	Error code	

5.1.5 MaResMgr_GetDefVoiceSynth

SINT32 MaResMgr_GetDefVoiceSynth (UINT8 prog);

Description

Returns the type of synthesizer that used with designated tone registration number.

Argument

prog	program number (0 to 127: melody tones、 128 to 255: drum tones)
------	---

Returned value

Positive	Type of synthesizer (1: FM, 2: WT, 3: Stream Audio)
0	Tones are not registered.
Negative	Error code.

5.1.6 MaResMgr_GetDefVoiceKey

SINT32 MaResMgr_GetDefVoiceKey (UINT8 prog);

Description

For FM tones, this returns tone generation key number of tones with designated tone registration number.
 For WT tones, this returns scale value assuming that C4 of tones of designated tone registration number is 1.0.

Argument

prog	Program number (128 to 255: drum tones)
------	---

Returned value

Positive	Tone generation key of tones
0	Tones are not registered.
Negative	Error code.

5.1.7 MaResMgr_RegStreamAudio

SINT32 MaResMgr_RegStreamAudio (UINT8 wave_id, UINT8 format, UINT8 * wave_ptr, UINT32 wave_size);

Description

Registers waveform data of Stream Audio into designated number.

Argument

wave_id	Waveform data registration number of Stream Audio
format	Format of waveform data of Stream Audio
wave_ptr	Pointer for waveform data of Stream Audio
wave_size	Byte size of waveform data of Stream Audio

Returned value

0	Successful.
Negative	Error code.

5.1.8 MaResMgr_DelStreamAudio

SINT32 MaResMgr_DelStreamAudio (UINT8 wave_id);

Description

Deletes waveform data of Stream Audio with designated number.

Argument

wave_id	Management number of waveform data of Stream Audio
---------	--

Returned value

0	Successful.
Negative	Error code.

5.1.9 MaResMgr_GetStreamAudioInfo

SINT32 MaResMgr_GetStreamAudioInfo (UINT8 wave_id, UINT8 * format, UINT8 ** wave_ptr, UINT32 * wave_size, UINT32 * seek_pos);

Description

Acquires information of waveform data of Stream Audio.

Argument

wave_id	Management number of waveform data of Stream Audio
format	Returns format of of waveform data of Stream Audio
wave_ptr	Returns pointer for which waveform data of Stream Audio is stored.
wave_size	Returns size of waveform data of Stream Audio.
seek_pos	Seek position of Stream Audio (Byte).

Returned value

0	Successful.
Negative	Error code.

5.1.10 MaResMgr_AllocStreamAudio

SINT32 MaResMgr_AllocStreamAudio (UINT32 sa_map);

Description

Secures designated Stream Audio.

Secures WT slot, built-in RAM, Software interrupt and timer.

Argument

sa_map	Map of Stream Audio: Bits 1 and 0 are valid. '1' is specified to be the bit to which the stream audio to secure corresponds. bit 0 Stream audio #0 bit 1 Stream audio #1 Note: All other bits are to be set to "0"
--------	--

Returned value

0	Successful.
Negative	Error code.

5.1.11 MaResMgr_FreeStreamAudio

SINT32 MaResMgr_FreeStreamAudio (UINT32 sa_map);

Description

Releases designated Stream Audio.

Releases WT slot, built-in RAM, Software interrupt and timer that have been secured.

Argument

sa_map	Map of Stream Audio: Bits 1 and 0 are valid. bit 0 Stream audio #0 bit 1 Stream audio #1 Note: All other bits are to be set to "0"
--------	---

Returned value

0	Successful.
Negative	Error code.

5.1.12 MaResMgr_AllocRam

SINT32 MaResMgr_AllocRam (UINT32 rb_map);

Description

Secures designated built-in RAM block. Two or more RAM blocks can be designated at a time.

Argument

rb_map	Map of RAM block to be secured: Bits 7 to 0 are valid. Designates “1” to an applicable bit of RAM block to be secured.
bit 0	RAM block #0
:	:
bit 7	RAM block #7

Note: All other bits are to be set to “0”.

Returned value

0	Successful.
Negative	Error code.

5.1.13 MaResMgr_FreeRam

SINT32 MaResMgr_FreeRam (UINT32 rb_map);

Description

Releases designated built-in RAM block. Two or more RAM blocks can be designated at a time.

Argument

rb_map	Map of RAM block to be released: Bits 7 to 0 are valid. Sets “1” to an applicable bit of RAM block to be released.
bit 0	RAM block #0
:	:
bit 7	RAM block #7

Note: All other bits are to be set to “0”.

Returned value

0	Successful.
Negative	Error code.

5.1.14 MaResMgr_AllocCh

SINT32 MaResMgr_AllocCh (UINT32 ch_map);

Description

Secures designated tone generation channel. Two or more tone generation channels can be designated at a time.

Argument

ch_map	Map of tone generation channel to be secured: Bits 15 to 0 are valid. Designates “1” for applicable bit of tone generation channel to be secured.
bit 0	Tone generation channel #0
:	:
bit 15	Tone generation channel #15

Note: All other bits are to be set to “0”.

Returned value

0	Successful.
Negative	Error code.

5.1.15 MaResMgr_FreeCh

```
SINT32 MaResMgr_FreeCh ( UINT32 ch_map );
```

Description	Releases designated tone generation channel. Two or more tone generation channels can be designated at a time.	
Argument	ch_map	Map of tone generation channel to be released: Bits 15 to 0 are valid. Sets “1” to applicable bit of tone generation channel to be secured.
	bit 0	Tone generation channel #0
	:	:
	bit 15	Tone generation channel #15
	Note: Note: All other bits are to be set to “0”.	
Returned value	0	Successful.
	Negative	Error code.

5.1.16 MaResMgr AllocFmVoice

```
SINT32 MaResMgr_AllocFmVoice ( UINT32 fm_map );
```

Description	Secures designated FM Voice. Two or more FM Voices can be designated at a time.	
Argument	fm_map	<p>Map of FM voices to be secured: Bits 31 to 16/bits 15 to 0 are valid.</p> <p>Designates “1” for applicable bit of FM voices to be secured.</p> <p>bit 0 Tone generation channel #0</p> <p> :</p> <p>bit 15/31 Tone generation channel #15/31</p> <p>Note: All other bits are to be set to “0”.</p> <p>Note: Whether up to bit 15 can be designated or up to bit 31 can be designated depends on the mode.</p>
Returned value	0	Successful.
	Negative	Error code.

5.1.17 MaResMgr_FreeFmVoice

SINT32 MaResMgr_FreeFmVoice (UINT32 fm_map);

Description	Releases designated FM Voice. Two or more FM Voices can be designated at a time.							
Argument	fm_map	<p>Map of FM voices to be released: Bits 31 to 16/bits 15 to 0 are valid. Designate “1” for applicable bit of FM voices to be released.</p> <table><tr><td>bit 0</td><td>Tone generation channel #0</td></tr><tr><td>:</td><td>:</td></tr><tr><td>bit 15/31</td><td>Tone generation channel #15/31</td></tr></table> <p>Note: All other bits are to be set to “0”. Note: Whether up to bit 15 can be designated or up to bit 31 can be designated depends on the mode.</p>	bit 0	Tone generation channel #0	:	:	bit 15/31	Tone generation channel #15/31
bit 0	Tone generation channel #0							
:	:							
bit 15/31	Tone generation channel #15/31							
Returned value	0	Successful.						
	Negative	Error code.						

5.1.18 MaResMgr_AllocWtVoice

SINT32 MaResMgr_AllocWtVoice (UINT32 wt_map);

Description

Secures designated WT Voice. Two or more WT Voices can be designated at a time.

Argument

wt_map	Map of WT voices to be secured: Bits 7 to 0 are valid. Designate “1” for applicable bit of WT voices to be secured.
bit 0	Tone generation channel #0
:	:
bit 7	Tone generation channel #7

Note: All other bits are to be set to “0”.

Returned value

0	Successful.
Negative	Error code.

5.1.19 MaResMgr_FreeWtVoice

SINT32 MaResMgr_FreeWtVoice (UINT32 wt_map);

Description

Releases designated WT Voice. Two or more WT Voices can be designated at a time.

Argument

wt_map	Map of WT voices to be released: Bits 7 to 0 are valid. Designate “1” for applicable bit of WT voices to be released.
bit 0	Tone generation channel #0
:	:
bit 7	Tone generation channel #7

Note: All other bits are to be set to “0”.

Returned value

0	Successful.
Negative	Error code.

5.1.20 MaResMgr_AllocSoftInt

SINT32 MaResMgr_AllocSoftInt (UINT32 si_map);

Description

Secures designated Software interrupt, and registers processing function.

Argument

si_map	Map of Software interrupt to be secured: Bits 3 to 0 are valid. Sets “1” to an applicable bit of Software interrupt to be secured.
bit 0	Software interrupt #0
:	:
bit 3	Software interrupt #3

Note: All other bits are to be set to “0”.

Returned value

0	Successful.
Negative	Error code.

5.1.21 MaResMgr_FreeSoftInt

SINT32 MaResMgr_FreeSoftInt (UINT32 si_map);

Description

Releases designated Software interrupt.

Argument

si_map	Map of Software interrupt to be released: Bit 3 -bit 0 is valid. Sets "1" to an applicable bit of Software interrupt to be released.
bit 0	Software interrupt #0
:	:
bit 3	Software interrupt #3

Note: All other bits are to be set to "0".

Returned value

0	Successful.
Negative	Error code.

5.1.22 MaResMgr_AllocLed

SINT32 MaResMgr_AllocLed (UINT32 ld_map);

Description

Secures designated LED.

Argument

ld_map	Map of LED to be secured: Bit 0 is valid. Sets "1" to applicable bit of LED to be secured.
bit 0	LED

Note: All other bits are to be set to "0".

Returned value

0	Successful.
Negative	Error code.

5.1.23 MaResMgr_FreeLed

SINT32 MaResMgr_FreeLed (UINT32 ld_map);

Description

Releases designated LED.

Argument

ld_map	Map of LED to be released: Bit 0 is valid. Sets "1" to applicable bit of LED to be released.
bit 0	LED

Note: All other bits are to be set to "0".

Returned value

0	Successful.
Negative	Error code.

5.1.24 MaResMgr_AllocMotor

SINT32 MaResMgr_AllocMotor (UINT32 mt_map);

Description

Secures designated motor.

Argument

mt_map	Map of Motor to be secured: Bit 0 is valid. Sets "1" to applicable bit of Motor to be secured. bit 0 Motor Note: All other bits are to be set to "0".
--------	---

Returned value

0	Successful.
Negative	Error code.

5.1.25 MaResMgr_FreeMotor

SINT32 MaResMgr_FreeMotor (UINT32 mt_map);

Description

Releases designated motor.

Argument

mt_map	Map of Motor to be released: Bit 0 is valid. Sets "1" to applicable bit of Motor to be released. bit 0 Motor Note: All other bits are to be set to "0".
--------	---

Returned value

0	Successful.
Negative	Error code.

5.1.26 MaResMgr_AllocSequencer

SINT32 MaResMgr_AllocSequencer (UINT8 seq_id, UINT16 base_time);

Description

Secures designated sequencer and performs the setting.
Secures Timer #1 for reproduction position count, and sets for sequencer synchronization.

Argument

seq_id	Sequencer number (0)
base_time	Minimum time unit of sequencer

Returned value

0	Successful.
Negative	Error code.

5.1.27 MaResMgr_FreeSequencer

SINT32 MaResMgr_FreeSequencer (UINT8 seq_id);

Description

Releases designated sequencer.

Argument

seq_id Sequencer number (0)

Returned value

0 Successful.
Negative Error code.

5.1.28 MaResMgr_AllocTimer

SINT32 MaResMgr_AllocTimer (UINT8 timer_id, UINT8 base_time, UINT8 time, UINT8 mode, UINT8 one_shot);

Description

Secures designated built-in timer, and performs the setting.

base_time that is set lastly is shared by all timers share (excluding sequencer synchronization)

Argument

timer_id Timer number (0 to 2)
base_time Minimum time unit of timer (0 to 127)
time Setting of time (0 to 127)
mode 0: Basic time of timer, 1: Basic time of sequencer
one_shot 0: Continuous, 1: One shot

Returned value

0 Successful.
Negative Error code.

5.1.29 MaResMgr_FreeTimer

SINT32 MaResMgr_FreeTimer (UINT8 timer_id);

Description

Releases designated built-in timer.

Argument

timer_id Timer number (0 to 2)

Returned value

0 Successful.
Negative Error code.

5.1.30 MaResMgr_SetStreamSeekPos

SINT32 MaResMgr_SetStreamSeekPos(UINT8 wave_id, UINT32 seek_pos);

Description

Sets Seek position of Stream Audio.

Argument

wave_id	Management number of waveform data of Stream Audio. (0 to 31)
seek_pos	Seek position of Stream Audio (Byte)

Returned value

0	Successful.
Negative	Error code.

6 MA Device Driver

MA Device Driver autucally performs writing into / reading from hardware registers, and hardware interrupt processing.

Function name	Description
MaDevDrv_Initialize	Initializes MA Device Driver.
MaDevDrv_PowerManagement	Performs setting for Power management.
MaDevDrv_InitRegisters	Initializes the register which does not require a software reset.
MaDevDrv_VerifyRegisters	Checks whether the software reset has been performed normally.
MaDevDrv_DeviceControl	Performs setting for device.
MaDevDrv_ReceiveData	Receives data from real time read path.
MaDevDrv_SendDirectPacket	Sets packets to real time write path.
MaDevDrv_SendDelayedPacket	Sets packets to delayed write path.
MaDevDrv_SendDirectRamData	Sends data to built-in RAM.
MaDevDrv_SendDirectRamVal	Sends fixed value of designated number to built-in RAM.
MaDevDrv_StartSequencer	Performs starting of sequencer.
MaDevDrv_StopSequencer	Performs stopping of sequencer.
MaDevDrv_EndOfSequence	Performs ending processing of sequence data.
MaDevDrv_ClearFifo	Performs clearing of FIFO.
MaDevDrv_StreamSetup	Performs Stream Audio waveform setting.
MaDevDrv_StreamUpdate	Performs replenishment of Stream Audio waveform data.
MaDevDrv_ControlInterrupt	Performs interrupt control.
MaDevDrv_SetAudioMode	Sets identification flag for Audio system.
MaDevDrv_GetStreamPos	Returns present reproduction position of Stream Audio.
MaDevDrv_GetSeekBuffer	Returns pointer for buffer at seek.
MaDevDrv_SeekControl	Performs control at seek.
MaDevDrv_AddIntFunc	Registers interrupt processing function.
MaDevDrv_RemoveIntFunc	Deletes interrupt processing function.
MaDevDrv_IntHandler	Interrupt processing function
MaDevDrv_StreamHandler	Performs control of Stream Audio.
MaDevDrv_SoftInt0	Performs interrupt processing of software interrupt #0.
MaDevDrv_SoftInt1	Performs interrupt processing of software interrupt #1.
MaDevDrv_SoftInt2	Performs interrupt processing of software interrupt #2.
MaDevDrv_SoftInt3	Performs interrupt processing of software interrupt #3.
MaDevDrv_Timer0	Performs interrupt processing of timer #0.
MaDevDrv_Timer1	Performs interrupt processing of timer #1.
MaDevDrv_Fifo	Performs FIFO interrupt processing.

6.1 Function definition

6.1.1 MaDevDrv_Initialize

SINT32 MaDevDrv_Initialize (void);

Description

Initializes device, and performs initialization of MA Device Driver.

Argument

None

Returned value

0 Successful.
Negative Error code.

6.1.2 MaDevDrv_DeviceControl

SINT32 MaDevDrv_DeviceControl (UINT8 cmd, UINT8 param1, UINT8 param2, UINT8 param3);

Description

Performs various setting of device.

Argument

cmd Command number
param1 Parameter 1
param2 Parameter 2
param3 Parameter 3

Message	cmd	param1	param2	param3
DIGITAL_PMG	0	val	0	0
ANALOG_PMG	1	val	0	0
EQVOL	2	vol	0	0
HPVOL	3	mono	vol_l	vol_r
SPVOL	4	vol	0	0
LED	5	led	freq	mode

MOTOR	6	mtr	freq	mode
PLL	7	adjust1	adjust2	0
VOL_MODE	8	mute	chvol	panpot
EFFECT	9	prb	0	0
FM_MODE	10	mode	0	0
GET_SEQSTATUS	11	0	0	0

Returned value

0	Successful.
Negative	Error code.

6.1.3 MaDevDrv_InitRegisters

```
void MaDevDrv_InitRegisters ( void );
```

Description

Initializes the register which does not require a software reset.

Argument

None

Returned value

None

6.1.4 MaDevDrv_VerifyRegisters

```
SINT32 MaDevDrv_VerifyRegisters ( void );
```

Description

Checks whether the software reset has been performed normally.

Argument

None

Returned value

0	Successful.
Negative	Error code.

6.1.5 MaDevDrv_PowerManagement

```
SINT32 MaDevDrv_PowerManagement ( UINT8 mode );
```

Description

Sets or releases device power down.

Argument

mode	0	Hardware initialization sequence (power down state)
	1	Hardware initialization sequence (regular operation)
	2	Power down enabling sequence
	3	Power down release sequence

Returned value

0	Successful.
Negative	Error code.

6.1.6 MaDevDrv_ReceiveData

 UINT8 MaDevDrv_ReceiveData(UINT32 address, UINT8 buffer_address);

Description

Reads register data.
 Writes real time write packet into REG_ID #3 real time read-in register, and reads byte data.
 Sets ID #3 into REG_ID of ADR #0 status flag.
 Generates real time read packet from address and buffer_address, writes it into real time read-in register.
 After verifying that data are ready, clear the flag and write buffer number.
 Read-in byte data that have been read.

Argument

address	Address of register to be read
buffer_address	Address of return buffer

Returned value

Byte data that have been read

6.1.7 MaDevDrv_SendDirectPacket

 SINT32 MaDevDrv_SendDirectPacket(const UINT8 *ptr, UINT16 size);

Description

Writes real time write packet into REG_ID #2 real time write register.

Argument

ptr	Pointer for domain where real time write packet is stored.
size	Byte size of real time write packet

Returned value

0	Successful
Negative	Error code

6.1.8 MaDevDrv_SendDelayedPacket

 SINT32 MaDevDrv_SendDelayedPacket(const UINT8 *ptr, UINT16 size);

Description

Writes delay write packet into REG_ID #1 delay write register.

Argument

ptr	Pointer for domain where delay write packet is stored.
size	Byte size of delay write packet

Returned value

0	Successful
Negative	Error code

6.1.9 MaDevDrv_SendDirectRamData

SINT32 MaDevDrv_SendDirectRamData(UINT32 address, UINT8 data_type, const UINT8 *data_ptr, UINT32 data_size);

Description

Converts data to be written into built-in RAM into packet form, and writes the data into REG_ID #2 real time write register.

Argument

address	Address of built-in RAM
data_type	Specifies the data type. (0: 8bit, 1: 7bit encode)
data_ptr	Pointer for domain where the data written in RAM is stored.
data_size	Byte size written in RAM.

Returned value

0	Successful
Negative	Error code

6.1.10 MaDevDrv_SendDirectRamVal

SINT32 MaDevDrv_SendDirectRamVal(UINT32 address, UINT32 data_size, UINT8 val);

Description

Converts data to be written into built-in RAM into packet form, and writes the data into REG_ID #2 real time write register.

Argument

address	Address of built-in RAM
data_size	Byte size of delay write packet
val	Byte data to be written

Returned value

0	Successful
Negative	Error code

6.1.11 MaDevDrv_StartSequencer

SINT32 MaDevDrv_StartSequencer(SINT32 seq_id, UINT8 ctrl);

Description

Starts sequencer.

Argument

seq_id	Sequence registration number
ctrl	Control flag

Returned value

0	Successful
Negative	Error code

6.1.12 MaDevDrv_StopSequencer

SINT32 MaDevDrv_StartSequencer(SINT32 seq_id, UINT8 ctrl);

Description

Stops sequencer.

Argument

seq_id	Sequence registration number
ctrl	Control flag

Returned value

0	Successful
Negative	Error code

6.1.13 MaDevDrv_EndOfSequence

SINT32 MaDevDrv_EndOfSequence(void);

Description

Receives that sequencer data have ended.

Argument

None

Returned value

0	Successful
Negative	Error code

6.1.14 MaDevDrv_ClearFifo

SINT32 MaDevDrv_ClearFifo(void);

Description

Performs clearing of FIFO.

Argument

None

Returned value

0	Successful
Negative	Error code

6.1.15 MaDevDrv_StreamSetup

SINT32 MaDevDrv_StreamSetup(UINT8 sa_id);

Description

Performs Stream Audio waveform setting.

Argument

sa_id	Stream Audio number
-------	---------------------

Returned value

0	Successful
Negative	Error code

6.1.16 MaDevDrv_StreamUpdate

SINT32 MaDevDrv_StreamSetup(UINT8 sa_id);

Description

Performs supplement of Stream Audio waveform data.

Argument

sa_id Stream Audio number

Returned value

0 Successful

Negative Error code

6.1.17 MaDevDrv_ControlInterrupt

SINT32 MaDevDrv_ControlInterrupt(UINT8 ctrl, UINT8 int_ctrl);

Description

Performs interrupt control.

Argument

ctrl Control flag (0: enable, 1: disable)

int_ctrl Interrupt number

Returned value

None

6.1.18 MaDevDrv_SetAudioMode

SINT32 MaDevDrv_SetAudioMode(UINT8 mode);

Description

Sets identification flag for audio system.

Argument

mode Mode (0: Non-audio system, 1: Audio system)

Returned value

None

6.1.19 MaDevDrv_GetStreamPos

UINT32 MaDevDrv_GetStreamPos(UINT8 ctrl);

Description

Returns Stream Audio reproduction position.

Argument

ctrl 0: Acquires the current position.

1: Resets a current position count.

Returned value

Audio reproduction position [ms]

6.1.20 MaDevDrv_GetSeekBuffer

 UINT8 * MaDevDrv_GetSeekBuffer(UINT16 * size);

Description

Returns pointer and size of buffer that set control data at seek.

Argument

*size	Pointer that returns buffer size.
-------	-----------------------------------

Returned value

Pointer for buffer

6.1.21 MaDevDrv_SeekControl

 SINT32 MaDevDrv_SeekControl(SINT32 seq_id, UINT32 buf_size);

Description

Sends control value at seek.

Argument

seq_id	Sequence registration number
buf_size	Byte size of buffer

Returned value

0	Successful
Negative	Error code

6.1.22 MaDevDrv_AddIntFunc

 SINT32 MaDevDrv_AddIntFunc(UINT8 number, void (* int_func)(UINT8 ctrl));

Description

Registers interrupt processing function.

Argument

number	Processing number
--------	-------------------

0	SIRQ #0	Function that is called when software interrupt 0 has occurred.
1	SIRQ #1	Function that is called when software interrupt 1 has occurred.
2	SIRQ #2	Function that is called when software interrupt 2 has occurred.
3	SIRQ #3	Function that is called when software interrupt 3 has occurred.
4		
5	TM #0	Function that is called when timer 0 interrupt has occurred.
6	TM #1	Function that is called when timer 1 interrupt has occurred.
7	FIFO	Function that is called when sequence FIFO interrupt has occurred.

int_func	Interrupt processing function to be registered.
----------	---

Returned value

0	Successful
Negative	Error code

6.1.23 MaDevDrv RemoveIntFunc

```
SINT32 MaDevDrv_RemoveIntFunc ( UINT8 number );
```

Description	Deletes interrupt processing function that is registered.	
Argument	number	Processing number
Returned value	0	Successful
	Negative	Error code

6.1.24 MaDevDrv_IntHandler

SINT32 MaDevDrv_IntHandler (void);

Description	Receives hardware interrupt. Identifies cause of interrupt, and calls interrupt processing function that has already been registered.	
Argument	None	
Returned value	0	Successful
	Negative	Failed

6.1.25 MaDevDrv_StreamHandler

```
SINT32 MaDevDrv_StreamHandler( UINT8 sa_id, UINT8 ctrl, UINT8 ram_val );
```

Description	Performs control of Stream Audio.	
Argument	sa_id	Stream Audio number (0 or 1)
	ctrl	Direction of control 0: Preparation of waveform data 1: Supplementation of waveform data
	ram_val	The value of SINT RAM (0 ~ 7Fh)
Returned value	0	Successful.
	Negative	Error code.

6.1.26 MaDevDrv_SoftInt0

```
void MaDevDrv_SoftInt0( UINT8 ctrl );
```

Description	Performs interrupt processing of software interrupt #0.		
Argument	ctrl	Unused	Sets "0".
Returned value			

None

6.1.27 MaDevDrv_SoftInt1

void MaDevDrv_SoftInt0(UINT8 ctrl);

Description

Performs interrupt processing of software interrupt #1.

Argument

ctrl Unused Sets "0".

Returned value

None

6.1.28 MaDevDrv_SoftInt2

void MaDevDrv_SoftInt2(UINT8 ctrl);

Description

Performs interrupt processing of software interrupt #2.

Argument

ctrl Unused Sets "0".

Returned value

None

6.1.29 MaDevDrv_SoftInt3

void MaDevDrv_SoftInt3(UINT8 ctrl);

Description

Performs interrupt processing of software interrupt #3.

Argument

ctrl Unused Sets "0".

Returned value

None

6.1.30 MaDevDrv_Timer0

void MaDevDrv_Timer0(UINT8 ctrl);

Description

Performs interrupt processing of time 0.

Argument

ctrl Unused Sets "0".

Returned value

None

6.1.31 MaDevDrv_Timer1

```
void MaDevDrv_Timer1( UINT8 ctrl );
```

Description

Performs interrupt processing of timer #1.

Argument

ctrl	Unused	Sets "0".
------	--------	-----------

Returned value

None

6.1.32 MaDevDrv_Fifo

```
void MaDevDrv_Timer0( UINT8 ctrl );
```

Description

Performs FIFO interrupt processing.

Argument

ctrl

Returned value

None

7 Default tones

7.1 Melody sound set

PC#	Instrument	Key Range	PC#	Instrument	Key Range
0	GrandPno	21 to 108	32	AcoBase	28 to 55
1	BritePno	21 to 108	33	FngrBass	28 to 55
2	E.GrandP	21 to 108	34	PickBass	28 to 55
3	HnkyTonk	21 to 108	35	Fretless	28 to 55
4	E.Piano1	28 to 103	36	SlapBas1	28 to 55
5	E.Piano2	28 to 103	37	SlapBas2	28 to 55
6	Harpsi	41 to 89	38	SynBass1	28 to 55
7	Clavi	36 to 96	39	SynBass2	28 to 55
8	Celesta	60 to 108	40	Violin	55 to 96
9	Glocken	72 to 108	41	Viola	48 to 84
10	MusicBox	60 to 84	42	Cello	36 to 72
11	Vibes	53 to 89	43	Contrabs	28 to 55
12	Marimba	48 to 84	44	TremStr	28 to 96
13	Xylophon	65 to 96	45	PizzStr	28 to 96
14	TubulBel	60 to 77	46	Harp	23 to 103
15	Dulcimar	60 to 84	47	Timpani	36 to 57
16	DrawOrgn	36 to 96	48	Strings1	28 to 96
17	PercOrgn	36 to 96	49	Strings2	28 to 96
18	RockOrgn	36 to 96	50	Syn.Str1	36 to 96
19	ChrchOrgn	21 to 108	51	Syn.Str2	36 to 96
20	ReedOrgn	36 to 96	52	ChoirAah	48 to 79
21	Acordion	53 to 89	53	VoiceOoh	48 to 79
22	Harmnica	60 to 84	54	SynVoice	48 to 84
23	TangoAcd	53 to 89	55	Orch.Hit	48 to 72
24	NylonGtr	40 to 84	56	Trumpte	58 to 94
25	SteelGtr	40 to 84	57	Trombone	34 to 75
26	JazzGtr	40 to 86	58	Tuba	29 to 55
27	CleanGtr	40 to 86	59	Mute.Trp	58 to 82
28	Mute.Gtr	40 to 86	60	Er.Horn	41 to 77
29	Ovrdrive	40 to 86	61	BrasSect	36 to 96
30	Dist.Gtr	40 to 86	62	SynBras1	36 to 96
31	GtrHarmo	40 to 86	63	SynBras2	36 to 96

PC#	Instrument	Key Range	PC#	Instrument	Key Range
64	SprnoSax	54-87	96	Rain	36-96
65	AltoSax	49-80	97	SoundTrk	36-96
66	TenorSax	42-75	98	Crystal	36-96
67	Bari.Sax	37-68	99	Atomosphr	36-96
68	Oboe	58-91	100	Bright	36-96
69	Eng.Horn	52-81	101	Goblins	36-96
70	Bassoon	34-72	102	Echoes	36-96
71	Clarinet	50-91	103	Sci-Fi	36-96
72	Piccolo	74-108	104	Sitar	48-77
73	Flute	60-96	105	Banjo	48-84
74	Recorder	60-96	106	Shamisen	50-79
75	PanFlute	60-96	107	Koto	55-84
76	Bottle	60-96	108	Kalimba	48-79
77	Shakhchi	55-54	109	Bagpipe	36-77
78	Whistel	60-96	110	Fiddle	55-96
79	Ocarina	60-84	111	Shanai	48-72
80	SeuqareLd	21-108	112	TnklBell	72-84
81	SawLead	21-108	113	Agogo	60-72
82	CaliopLd	36-96	114	SteelDrm	52-76
83	ChiffLd	36-96	115	WoodBlk	60-72 (*1)
84	CharanLd	36-96	116	TaikoDrm	60-72 (*2)
85	VoiceLd	36-96	117	MelodTom	60-72 (*3)
86	FifthLd	36-96	118	Syn.Drum	60-72 (*4)
87	Bass&Ld	21-108	119	RevCymbL	60-72 (*4)
88	NewAgePd	36-96	120	FretNoiz	60-72
89	WarmPad	36-96	121	BrthNoiz	60-72
90	PolySyPd	36-96	122	SeaShore	60-72 (*5)
91	ChoriPad	36-96	123	Tweet	60-72 (*6)
92	BowedPad	36-96	124	Telephone	60-72 (*7)
93	MetalPad	36-96	125	Helocptr	60-72 (*7)
94	HaloPad	36-96	126	Applause	60-72 (*6)
95	SweepPad	36-96	127	Gunshot	60-72 (*5)

Since tone generation of keys higher than 108 cannot be made, lower the octave to deal with this matter.

The following is applied even to other than Bank# = 0. (However, it is invalid in MA-1/2 compatible mode.)

- Exclusive assignment of three high hat sounds (Key#42 / Key#44 / Key#46)
- *1: 50 cents / half tone, Key#69 = F#4
- *2: 50 cents / half tone, Key#69 = A2
- *3: 50 cents / half tone, Key#69 = C#4
- *4: 50 cents / half tone
- *5: 20 cents / half tone
- *6: 5 cents / half tone
- *7: 10 cents / half tone

7.2 Percussion sound set

Bank# = 128

Key#	Instrument	Key#	Instrument
24	SeqClock H	56	Cowbell
25	Brush Tap	57	Crash Cymbal 2
26	Brush Swirl L	58	Vibraslap
27	Brush Slap	59	Ride Cymbal 2
28	Brush Swirl H	60	Bongo H
29	Snare Roll	61	Bongo L
30	Castanet	62	Conga H Mute
31	Snare L	63	Conga H Open
32	Sticks	64	Conga L
33	Bass Drum L	65	Timbale H
34	Open Rim Shot	66	Timbale L
35	Bass Drum M	67	Agogo H
36	Bass Drum H	68	Agogo L
37	Closed Rim Shot	69	Cabasa
38	Snare M	70	Maracas
39	Hand Clap	71	Samba Whistle H
40	Snare H	72	Samba Whistle L
41	Floor Tom L	73	Guiro Short
42	Hi-Hat Closed	74	Guiro Long
43	Floor Tom L	75	Claves
44	Hi-Hat Pedal	76	Wood Block H
45	Low Tom	77	Wood Block L
46	Hi-Hat Open	78	Cuica Mute
47	Mid Tom L	79	Cuica Open
48	Mid Tom H	80	Triangle Mute
49	Crash Cymbal 1	81	Triangle Open
50	High Tom	82	Shaker
51	Ride Cymbal 1	83	Jingle Bell
52	Chinese Cymbal	84	Belltree
53	RideCymbal Cup		
54	Tamboulin		
55	Splash Cymbal		

The following is applied even to other than Bank# = 128. (However, it is invalid in MA-1/2 compatible mode.)

- Exclusive assignment of three high hat tones (Key#42 / Key#44 / Key#46)
- Key#71 / Key#72 Exclusive assignment
- Key#73 / Key#74 Exclusive assignment
- Key#78 / Key#79 Exclusive assignment
- Key#80 / Key#81 Exclusive assignment