

Speech LSI

S1V3F351/S1V3F352

Sample Software

Manual

NOTICE: PLEASE READ THE FOLLOWING NOTICE CAREFULLY BEFORE USING THIS DOCUMENT

The contents of this document are subject to change without notice.

1. This document may not be copied, reproduced, or used for any other purpose, in whole or in part, without the consent of the Seiko Epson Corporation (“Epson”).
2. Before purchasing or using Epson products, please contact our sales representative for the latest information and always be sure to check the latest information published on Epson’s official web sites and other sources.
3. Information provided in this document such as application circuits, programs, usage, etc., are for reference purposes only. Using the application circuits, programs, usage, etc. in the design of your equipment or systems is your own responsibility. Epson makes no guarantees against any infringements or damages to any third parties’ intellectual property rights or any other rights resulting from the information. This document does not grant you any licenses, intellectual property rights or any other rights with respect to Epson products owned by Epson or any third parties.
4. Epson is committed to constantly improving quality and reliability, but semiconductor products in general are subject to malfunction and failure. By using Epson products, you shall be responsible for your hardware. Software and systems must be designed well enough to prevent death or injury as well as any property damage even if any of the malfunctions or failures might be caused by Epson products. When designing your products using Epson products, please be sure to check and comply with the latest information regarding Epson products (this document, specifications, data sheets, manuals, Epson’s web site, etc.). When using the information included above materials such as product data, charts, technical contents, programs, algorithms and application circuit examples, you shall evaluate your products both on a stand-alone basis as well as within your overall systems. You shall be solely responsible for deciding whether or not to adopt and use Epson products.
5. Epson has prepared this document and programs provided in this document carefully to be accurate and dependable, but Epson does not guarantee that the information and the programs are always accurate and complete. Epson assumes no responsibility for any damages which you incur due to misinformation in this document and the programs.
6. No dismantling, analysis, reverse engineering, modification, alteration, adaptation, reproduction, etc., of Epson products is allowed.
7. Epson products have been designed, developed and manufactured to be used in general electronic applications (office equipment, communications equipment, measuring instruments, home electronics, etc.) (“General Purpose”) and applications which is individually listed in this document or designated by Epson (“Designated Purpose”). Epson products are NOT intended for any use beyond the General Purpose and Designated Purpose uses that requires particular/higher quality or reliability in order to refrain from causing any malfunction or failure leading to death, injury, serious property damage or severe impact on society, including, but not limited to those listed below (“Particular Purpose”). Therefore, you are advised to use Epson products only for General Purpose and Designated Purpose uses. Should you desire to buy and use Epson products for a Particular Purpose, Epson makes no warranty and disclaims with respect to Epson products, whether express or implied, including without limitation any implied warranty of merchantability or fitness for any Particular Purpose. Please be sure to contact our sales representative and obtain approval in advance.
[Examples of Particular Purpose]
Space equipment (artificial satellites, rockets, etc.) /
Transportation vehicles and their control equipment (automobiles, aircraft, trains, ships, etc.) /
Medical equipment / Relay equipment to be placed on ocean floor /
Power station control equipment / Disaster or crime prevention equipment / Traffic control equipment / Financial equipment
Other applications requiring similar levels of reliability as those listed above. Please be sure to contact our sales representative for details of the other applications.
8. Epson products listed in this document and our associated technologies shall not be used in any equipment or systems that laws and regulations in Japan or any other countries prohibit to manufacture, use or sell. Furthermore, Epson products and our associated technologies shall not be used for developing weapons of mass destruction, or any other military purposes or applications. If exporting Epson products or our associated technologies, you shall comply with the Foreign Exchange and Foreign Trade Control Act in Japan, Export Administration Regulations in the U.S.A. (EAR) and other export-related laws and regulations in Japan and any other countries and follow the required procedures as provided by the relevant laws and regulations.
9. Epson assumes no responsibility for any damages (whether direct or indirect) caused by or in relation with your non-compliance with the terms and conditions in this document.
10. Epson assumes no responsibility for any damages (whether direct or indirect) incurred by any third party that you assign, transfer, loan, etc., Epson products to.
11. For more details or other concerns about this document, please contact our sales representative.
12. Company names and product names listed in this document are trademarks or registered trademarks of their respective companies.

Evaluation board/kit and Development tool important notice

1. Epson evaluation board/kit or development tool is designed for use for engineering evaluation, demonstration, or development purposes only. Do not use it for other purposes. It is not intended to meet the requirements of design for finished products.
2. Epson evaluation board/kit or development tool is intended for use by an electronic engineer and is not a consumer product. The user should use it properly and in a safe manner. Epson does not assume any responsibility or liability of any kind of damage and/or fire caused by the use of it. The user should cease to use it when any abnormal issue occurs even during proper and safe use.
3. The part used for Epson evaluation board/kit or development tool may be changed without any notice.

Rev. e1.4, 2023. 4

Table of Contents

1. Overview	1
1.1 Precautions for use	1
2. File Structure	2
3. S1V3F351/S1V3F352 Control Sequence	3
3.1 Sound Playback Control Sequence (main_play())	3
3.2 Sound Recording/Playback Control Sequence (main_rec_play())	4
3.3 Standby Mode Control Sequence (main_sleep())	5
3.4 Flash Programming Control Sequence (main_flash())	6
3.5 Tone Playback Control Sequence (main_tone_play())	7
4. Function Specification	8
4.1. Speech LSI Control Function	8
4.1.1 seSpeechInitialize	8
4.1.2 seSpeechReset	8
4.1.3 seSpeechSetSerialConfig	9
4.1.4 seSpeechSetOutputConfig	9
4.1.5 seSpeechSetSoundRom	10
4.1.6 seSpeechSetExternalFlashConfig	10
4.1.7 seSpeechSetExtCircuitControl	10
4.1.8 seSpeechSetVolume	11
4.1.9 seSpeechSetSpeed	11
4.1.10 seSpeechSetPitch (S1V3F351 only available)	11
4.1.11 seSpeechControlPlayback	12
4.1.12 seSpeechControlTone	13
4.1.13 seSpeechSetRecordedData	13
4.1.14 seSpeechControlRecording	13
4.1.15 seSpeechActivateFlashProgramMode	14
4.1.16 seSpeechEraseFlash	14
4.1.17 seSpeechWriteFlash	14
4.1.18 seSpeechReadFlash	15
4.1.19 seSpeechEnterSleep	15
4.1.20 seSpeechExitSleep	15
4.1.21 seSpeechSetKeyCode	15
4.1.22 seSpeechEraseSettingFlash	16
4.1.23 seSpeechWriteSettingFlash	16
4.1.24 seSpeechReadSettingFlash	16
4.1.25 seSpeechExecuteSelfCheck	16
4.1.26 seSpeechExecuteFlashCrcCheck	17
4.2. Serial I/F function	18
4.2.1 seSerialInitialize	18
4.2.2 seSpiInitialize	19
4.2.3 seI2cInitialize	19
4.2.4 seUartInitialize	19
4.2.5 seSerialSendRecvMessage	20
4.2.6 seSpiSendRecvMessage	20
4.2.7 seI2cSendRecvMessage	20
4.2.8 seUartSendRecvMessage	21
4.3. IND Messages Function	22

4.3.1	m_ISC_STATUS_IND	22
4.3.2	m_IND_ERROR_WARNING.....	22
4.3.3	m_IND_SOUND_OPERATION_STATE	22
4.3.4	m_IND_SOUND_OUTPUT_STATE	23
4.3.5	m_IND_CRC_SETTING.....	23
4.3.6	m_IND_SOUND_EFFECT_SETTINGS	23
4.3.7	m_IND_SOUND_ROM_SETTINGS.....	23
5.	Structure	24
5.1	SPEECH	24
5.2	msgResult	25
5.3	msgIndResult	25
5.4	msgErrorWarning	25
5.5	msgSoundOperationState.....	26
5.6	msgSoundOutputState.....	26
5.7	msgCrcSetting	26
5.8	msgSoundEffectSettings	27
5.9	msgSoundRomSettings	27
	Revision History	28

1. Overview

S1V3F351/S1V3F352 Sample Software Manual describes the sample software that controls S1V3F351/S1V3F352 from the host. It also describes the specifications of the API functions used in the sample software.

This document contains references to other documents. Please also refer to the following documents available from our website.

- S1V3F351/S1V3F352 Technical Manual
- S5U1V3F351T1 Manual or S5U1V3F352T1 Manual

1.1 Precautions for use

The S1V3F351/S1V3F352 sample software is reference material for controlling the S1V3F351/S1V3F352 from the host CPU. Our company is not responsible for any problems caused by this software. When using it for your product development, please conduct sufficient operation verification.

2. File Structure

2. File Structure

Table 2.1 lists the files included in the sample software.

Table 2.1 File List

File Name	Description	Note
main.c	Main source file (Please enable the function that you want to use)	
main_flash_prog.c	Flash programming example source file	
ROMImage.h ROMImage_test.h	Sound ROM file Sound ROM file for flash programming test	
main_play.c	Sound playback example source file	
main_rec_play.c	Sound recording/playback example source file	
main_sleep.c	Standby mode example source file	
main_tone_play.c	Tone playback example source file	
speech.c/.h	S1V3F351/S1V3F352 control function source file	
serial.c/.h	Serial I/F functions definition source file	
serial_drv.c/.h platfom.h port_config.h sePeripherallibrary	Serial I/F driver functions definition source file	
crc8.c/h	CRC value calculation function source file	

3. S1V3F351/S1V3F352 Control Sequence

The main source file contains a set of control sequence programs to perform various functions of the S1V3F351/S1V3F352, such as sound playback and flash programming.

This section describes the control sequence flow of S1V3F351/S1V3F352.

3.1 Sound Playback Control Sequence (main_play())

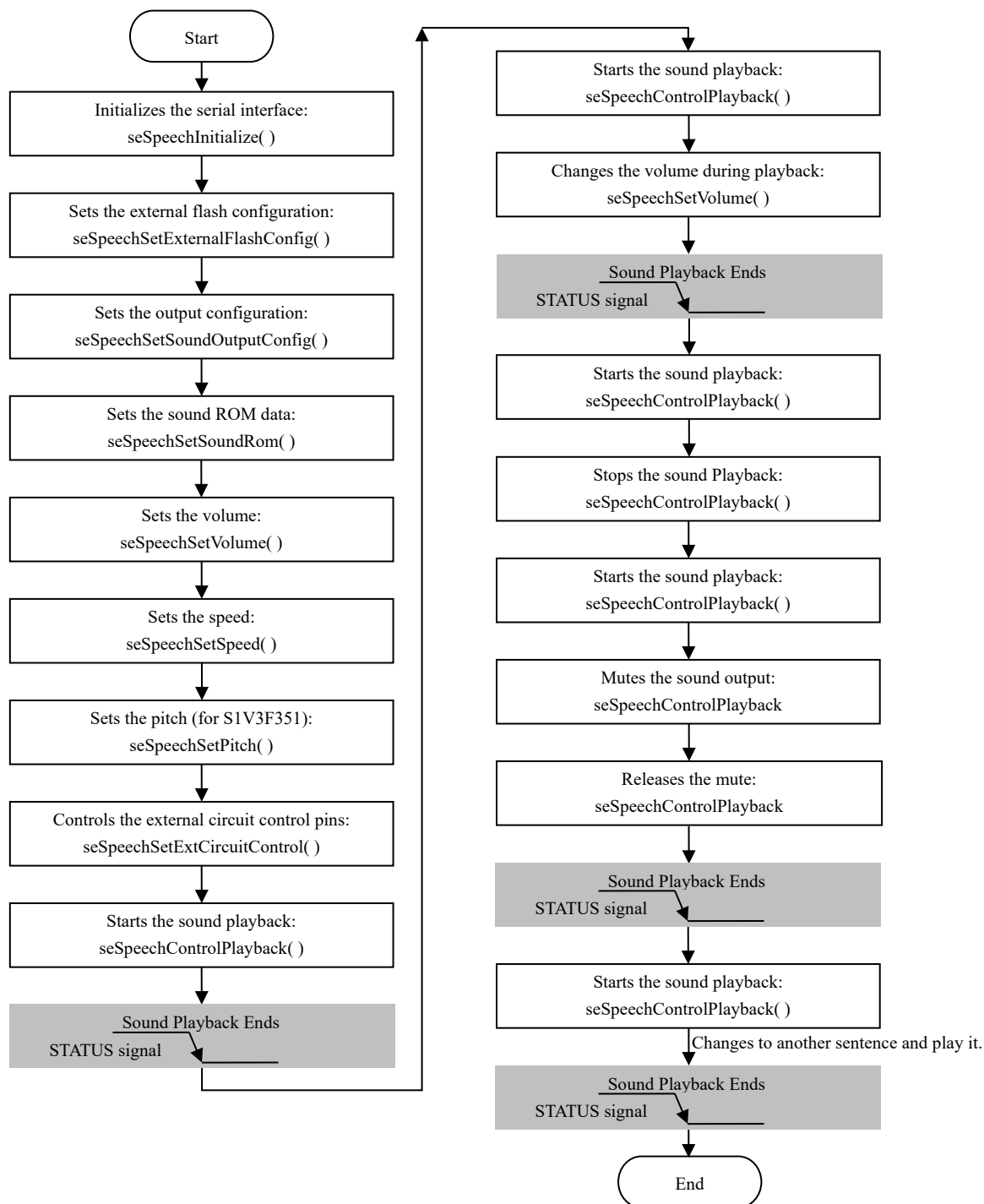


Figure 3.1 Sound Playback Control Sequence

3. S1V3F351/S1V3F352 Control Sequence

3.2 Sound Recording/Playback Control Sequence (main_rec_play())

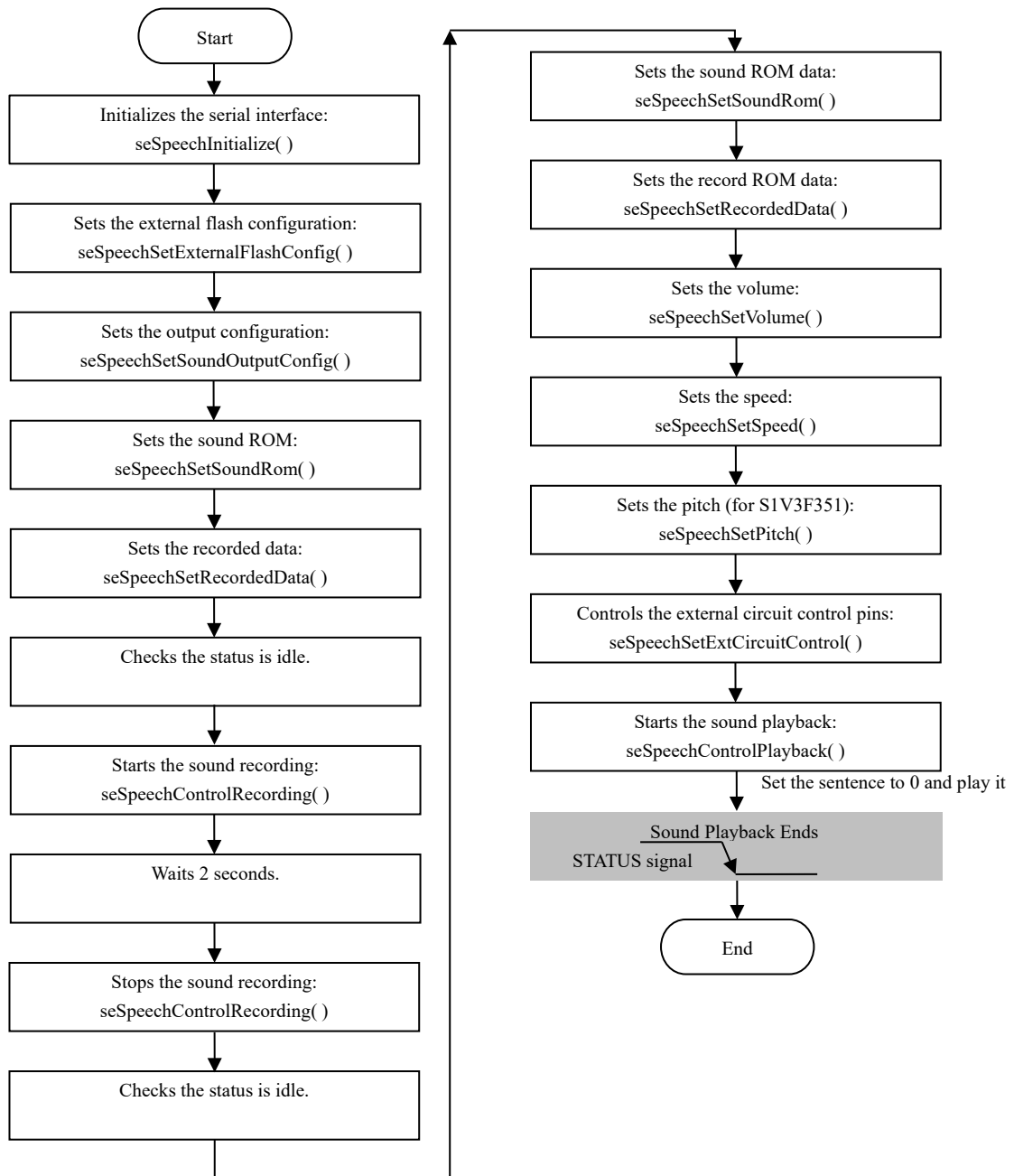


Figure 3.2 Sound Recording/Playback Control Sequence

3.3 Standby Mode Control Sequence (main_sleep())

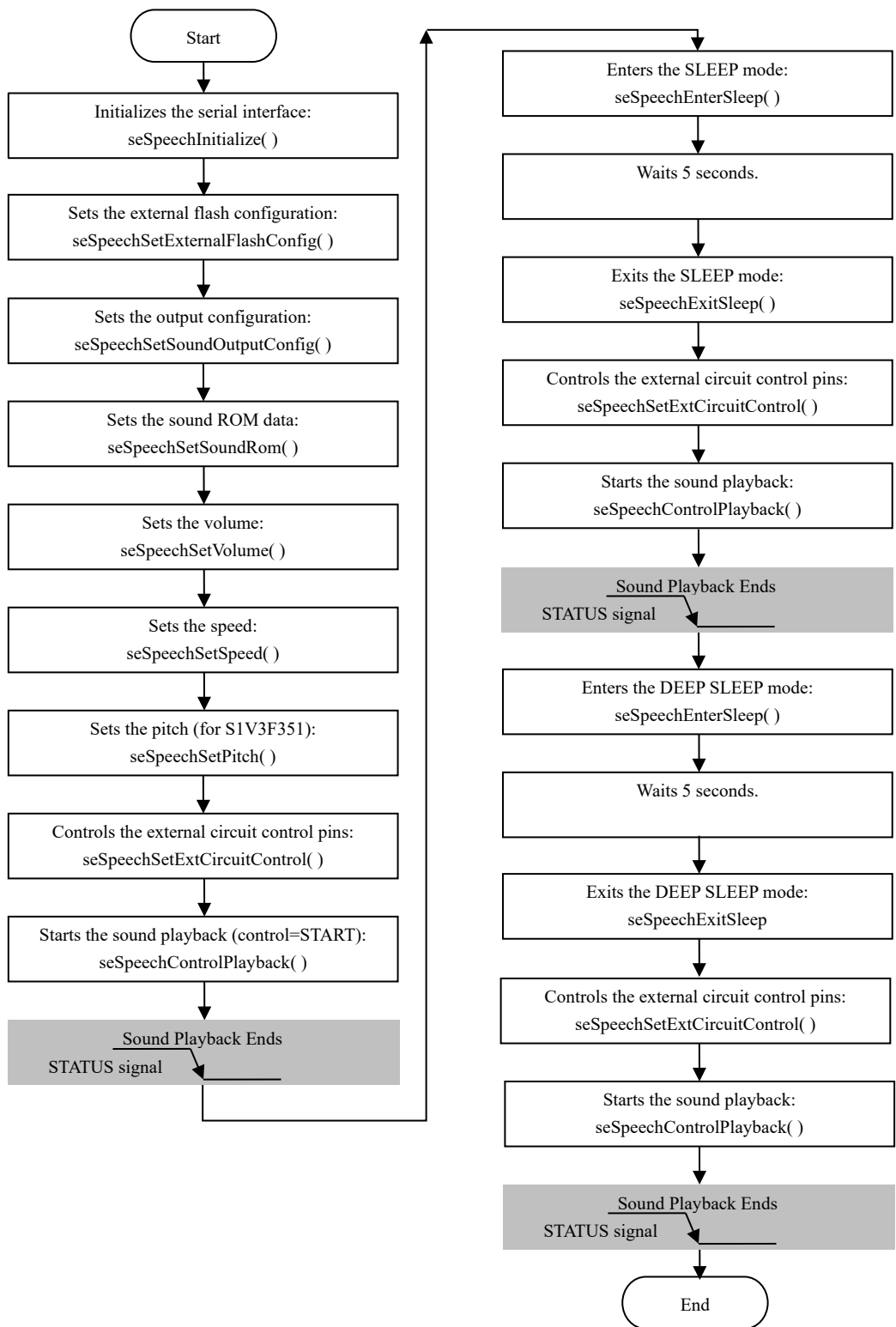


Figure 3.3 Standby Mode Control Sequence

3. S1V3F351/S1V3F352 Control Sequence

3.4 Flash Programming Control Sequence (main_flash())

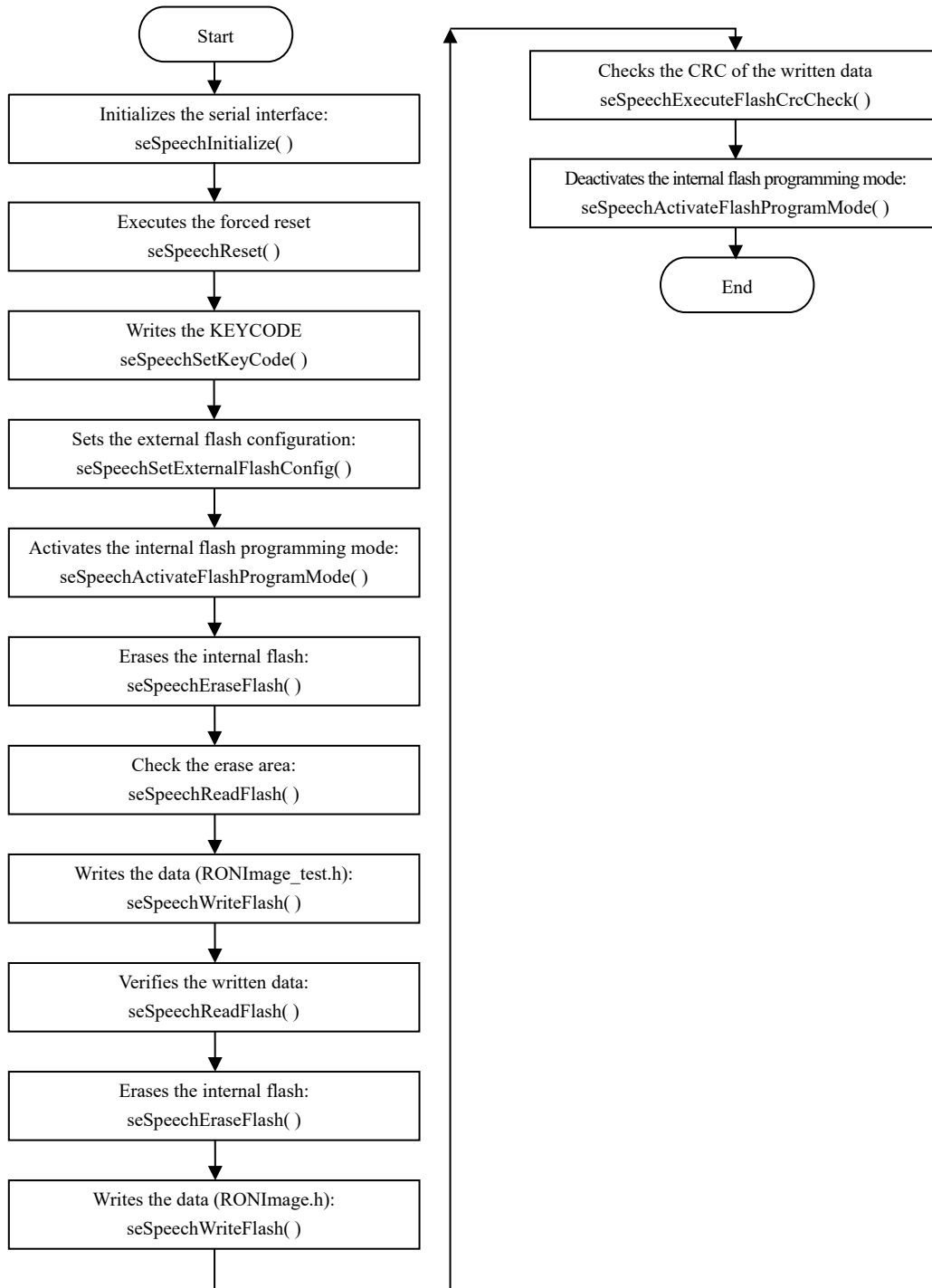


Figure 3.4 Flash Programming Control Sequence

3.5 Tone Playback Control Sequence (main_tone_play())

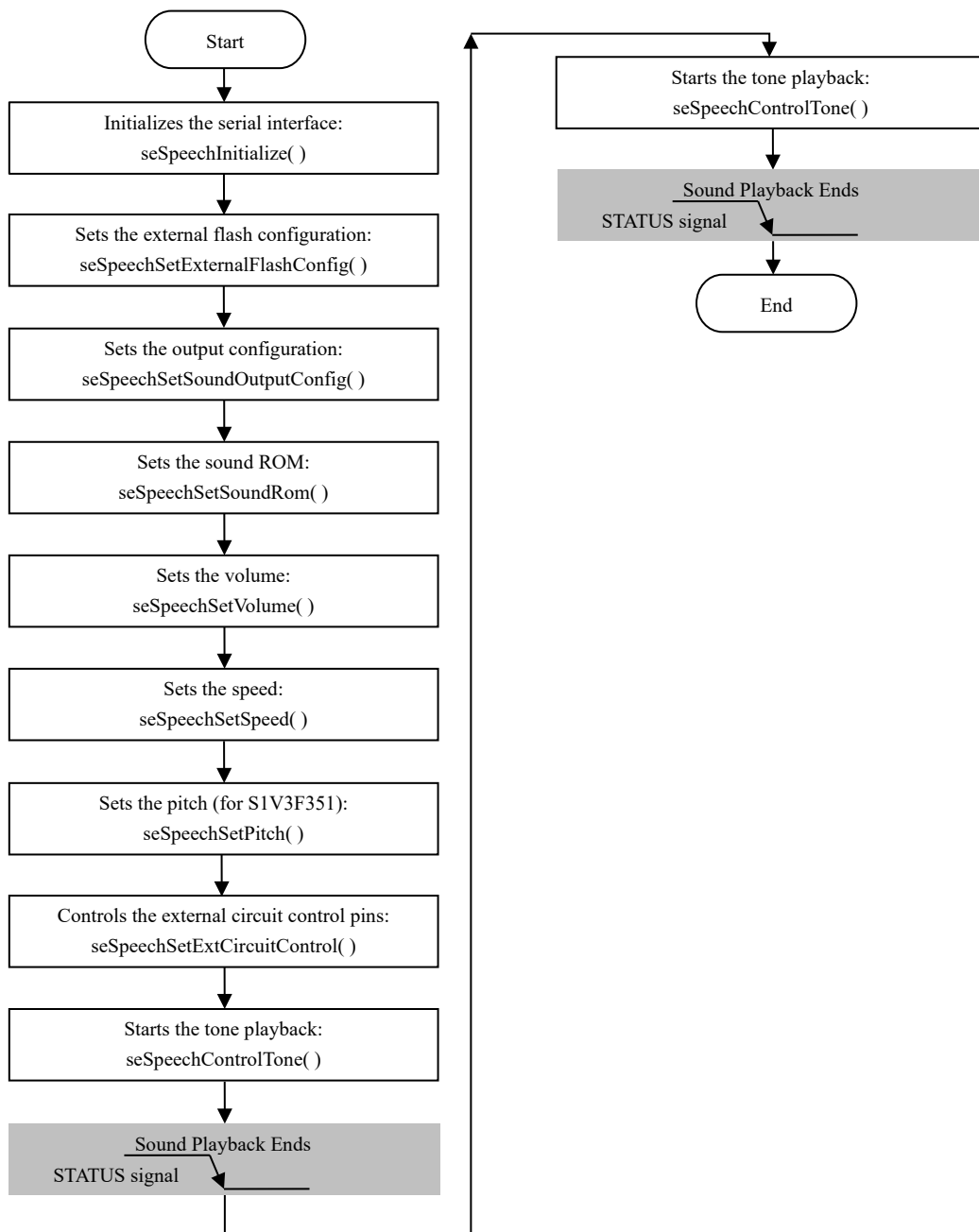


Figure 3.5 Tone Playback Control Sequence

4. Function Specification

4. Function Specification

This section describes the function specifications used in the S1V3F351/S1V3F352 sample software.

4.1. Speech LSI Control Function

The speech LSI control functions are API function for controlling the S1V3F351/S1V3F352.

These functions are defined in the speech.c/h file.

4.1.1 seSpeechInitialize

Function	int seSpeechInitialize (SPEECH *speech, uint8_t serial_type, uint32_t serial_clock, uint16_t uart_config)		
Argument	SPEECH_T	*speech	Pointer to the SPEECH structure
	uint8_t	serial_type	Serial I/F type. 0: SPI / 1: UART / 2: I2C
	uint32_t	serial_clock	Serial I/F clock. SPI: Max.400,000 Hz I2C: Max.300,000 Hz UART: 9600,19200,38400,57600,115200,230400 bps
	uint16_t	uart_config	UART communication parameters. [10:9] Parity 0: none / 1: even / 2: none / 3: odd [8] Stop Bit 0: 1bit / 1: 2bit Note1: Please set 0 to bits other than those listed above. Note2: Please set 0x0000 for SPI and I2C.
Return value	int	Returns 0 for normal termination, otherwise returns an error code.	
Description	This function initializes the serial I/F.		
Note			

4.1.2 seSpeechReset

Function	int seSpeechReset (uint8_t reset_type)		
Argument	SPEECH_T	*speech	Pointer to the SPEECH structure
	uint8_t	reset_type	Reset type. 0: Normal reset (clearing Non-fatal error) 1: Forced reset (clearing Fatal/Non-fatal error)
Return value	int	Returns 0 for normal termination, otherwise returns an error code.	
Description	This function resets S1V3F351/S1V3F352.		
Note			

4. Function Specification

4.1.3 seSpeechSetSerialConfig

Function	int seSpeechSetSerialConfig (SPEECH *speech, uint8_t serial_type, uint32_t serial_clock, uint16_t uart_config)		
Argument	SPEECH_T	*speech	Pointer to the SPEECH structure
	uint8_t	serial_type	Serial I/F type. 0: SPI / 1: UART / 2: I2C
	uint32_t	serial_clock	Serial I/F clock. SPI: Max.400,000 Hz I2C: Max.300,000 Hz UART: 9600,19200,38400,57600,115200,230400 bps
	uint16_t	uart_config	UART communication parameters. [10:9] Parity 0: none / 1: even / 2: none / 3: odd [8] Stop Bit 0: 1bit / 1: 2bit Note1: Please set 0 to bits other than those listed above. Note2: Please set 0x0000 for SPI and I2C.
Return value	int	Returns 0 for normal termination, otherwise returns an error code.	
Description	This function sets the serial I/F communication parameters.		
Note			

4.1.4 seSpeechSetOutputConfig

Function	int seSpeechSetOutputConfig (SPEECH *speech, uint8_t output_select, uint8_t sampling_rate)		
Argument	SPEECH_T	*speech	Pointer to the SPEECH structure
	uint8_t	output_select	Sets the sound output mode. <ul style="list-style-type: none"> • S1V3F351 0: speaker output 1: 2 pin buzzer output mode 2 2: 4 pin buzzer output mode 2 3: 2 pin buzzer output mode 1 4: 4 pin buzzer output mode 1 5: 2 pin buzzer output mode 3 6: 4 pin buzzer output mode 3 7: speaker output / 2pin buzzer output mode 3 • S1V3F352 0: speaker output 1: 2 pin buzzer output mode 1 2-6: Reserved 7: speaker output / 2pin buzzer output mode 3
	uint8_t	sampling_rate	Sets the sampling frequency. 0: 16kHz 1: 8kHz
Return value	int	Returns 0 for normal termination, otherwise returns an error code.	
Description	This function sets the sound output mode and the sampling frequency.		
Note			

4. Function Specification

4.1.5 seSpeechSetSoundRom

Function	int seSpeechSetSoundRom (SPEECH *speech, uint8_t flash_type, uint32_t rom_address, uint32_t rom_size)		
Argument	SPEECH_T	*speech	Pointer to the SPEECH structure
	uint8_t	flash_type	Sets the type of flash to store the sound ROM. 0: Internal Flash 1: External Serial Flash
	uint32_t	rom_address	Sets the address of the sound ROM storage area
	uint32_t	rom_size	Sets the size of the sound ROM
Return value	int	Returns 0 for normal termination, otherwise returns an error code.	
Description	This function sets the storage area and size of the sound ROM.		
Note			

4.1.6 seSpeechSetExternalFlashConfig

Function	int seSpeechSetExternalFlashConfig (uint8_t xip_activate, uint8_t xip_terminate, uint8_t xip_dummy_cycles)		
Argument	SPEECH_T	*speech	Pointer to the SPEECH structure
	uint8_t	xip_activate,	Sets the XIP mode activation bits described in the external serial flash memory specifications.
	uint8_t	xip_terminate	Sets the XIP mode termination bits described in the external serial flash memory specifications.
	uint8_t	xip_dummy_cycles	Sets the XIP dummy cycles described in the external serial flash memory specifications.
Return value	int	Returns 0 for normal termination, otherwise returns an error code.	
Description	This function sets the parameters to access external serial flash in XIP mode.		
Note	When using the external serial flash, please execute this function to set the XIP mode parameters.		

4.1.7 seSpeechSetExtCircuitControl

Function	int seSpeechSetExtCircuitControl (uint8_t control);		
Argument	uint8_t	control	Sets the value to control the EXT_CIRCUIT_CTRL pin.. 0: off 1: on
Return value	int	Returns 0 for normal termination, otherwise returns an error code.	
Description	This function controls the EXT_CIRCUIT_CONTROL pin to turn on/off the external circuit.		
Note			

4.1.8 seSpeechSetVolume

Function	int seSpeechSetVolume (SPEECH *speech, uint8_t ch0_volume, uint8_t ch1_volume)		
Argument	SPEECH_T	*speech	Pointer to the SPEECH structure
	uint8_t	ch0_volume	Sets the volume for channel 0. 0x00~0x7F * Settings other than the above are prohibited.
	uint8_t	ch1_volume	Sets the volume for channel 1. 0x00~0x7F * Settings other than the above are prohibited.
Return value	int	Returns 0 for normal termination, otherwise returns an error code.	
Description	This function sets sound volume for channel 0 and 1.		
Note	The volume can be changed in each channel individually, regardless of whether in Idle state or sound playback state.		

4.1.9 seSpeechSetSpeed

Function	int seSpeechSetSpeed (SPEECH *speech, uint8_t speed)		
Argument	SPEECH_T	*speech	Pointer to the SPEECH structure
	uint8_t	speed	Sets the playback speed for channel 0. 0x4B - 0x7D (75 - 125) * Settings other than the above are prohibited. * For details, refer to Technical Manual Chapter 8.5.
Return value	int	Returns 0 for normal termination, otherwise returns an error code.	
Description	This function sets the playback speed for channel 0.		
Note	Please set the playback speed as necessary before starting playback. These setting cannot be changed during playback.		

4.1.10 seSpeechSetPitch (S1V3F351 only available)

Function	int seSpeechSetPitch (SPEECH *speech, uint8_t pitch)		
Argument	SPEECH_T	*speech	Pointer to the SPEECH structure
	uint8_t	pitch	Sets the playback pitch for channel 0. 0x4B - 0x7D (75 - 125) * Settings other than the above are prohibited. * For details, refer to Technical Manual Chapter 8.5.
Return value	int	Returns 0 for normal termination, otherwise returns an error code.	
Description	This function sets the playback pitch for channel 0.		
Note	Please set the playback pitch as necessary before starting playback. These setting cannot be changed during playback.		

4. Function Specification

4.1.11 seSpeechControlPlayback

Function	int seSpeechControlPlayback (SPEECH *speech, uint8_t channel, uint8_t control, uint16_t ch0_sentence_no, uint16_t ch1_sentence_no, uint8_t ch0_repeat, uint8_t ch1_repeat)		
Argument	SPEECH_T	*speech	Pointer to the SPEECH structure
	uint8_t	channel	Sets the sound playback channel. 0: Send command only on channel 0 1: Send command only on channel 1 2: Send command on channel 0 and 1 simultaneously
	uint8_t	control	Sets the sound control command. 0x01: sound start 0x02: sound stop immediately 0x03: sound stop after current phrase 0x07: mute immediately 0x08: mute after current phrase 0x09: release mute * Settings other than the above are prohibited.
	uint16_t	ch0_sentence_no	Sets the sentence number of channel 0.
	uint16_t	ch1_sentence_no	Sets the sentence number of channel 1.
	uint8_t	ch0_repeat	Sets the number of repeats for channel 0. 0 or 1: 1 time (No repetition) 2~254: 2~254 times 255: endless loop (Continues until commanded to stop.)
	uint8_t	ch1_repeat	Sets the number of repeats for channel 1. 0 or 1: 1 time (No repetition) 2~254: 2~254 times 255: endless loop (Continues until commanded to stop.)
Return value	int	Returns 0 for normal termination, otherwise returns an error code.	
Description	This functions plays sound on specified channel.		
Note			

4.1.12 seSpeechControlTone

Function	int seSpeechControlTone (SPEECH *speech, uint16_t freq[4], uint8_t duration[4], uint8_t delay[4], uint8_t pattern_repeat, uint8_t control)		
Argument	SPEECH_T	*speech	Pointer to the SPEECH structure.
	uint16_t	freq[4]	Sets the tone frequency. 31 - 16000 Hz]
	uint8_t	duration[4]	Sets the tone playback duration. 0 - 2550[msec] (10 msec step) Note: If 0 is set, settings after TONE_FREQ1 are ignored.
	uint8_t	delay[4]	Sets the delay duration before next tone starts playback. 0 - 2550[msec] (10 msec step)
	uint8_t	pattern_repeat	Sets the number of repeats of the tone pattern. 0 or 1: 1 time (No repetition) 2 - 254: 2 - 254 times 255: endless loop (Continues until commanded to stop.)
	uint8_t	control	Sets the tone control command. 0: tone pattern playback start 1: single tone playback start (freq[0]) (Continues until commanded to stop.) 2: stop
Return value	int	Returns 0 for normal termination, otherwise returns an error code.	
Description	This function plays tones with a specified frequency combination pattern.		
Note			

4.1.13 seSpeechSetRecordedData

Function	int seSpeechSetRecordedData (uint32_t rec_address, uint16_t rec_size)		
Argument	uint32_t	rec_address	Sets the address of the recorded data storage area in external serial flash.
	uint16_t	rec_size	Sets the size of the recorded data.
Return value	int	Returns 0 for normal termination, otherwise returns an error code.	
Description	This function sets the storage area and size of the recorded data in external serial flash.		
Note			

4.1.14 seSpeechControlRecording

関数名	int seSpeechControlRecording (SPEECH *speech, uint8_t control)		
引数	SPEECH_T	*speech	Pointer to the SPEECH structure.
	uint8_t	control	Sets the recording command. 0: recording start 1: recording stop
Return value	int	Returns 0 for normal termination, otherwise returns an error code.	
Description	This function starts and stops recording.		
Note			

4. Function Specification

4.1.15 seSpeechActivateFlashProgramMode

Function	int seSpeechActivateFlashProgramMode (SPEECH *speech, uint8_t flash_type)		
Argument	SPEECH_T	*speech	Pointer to the SPEECH structure.
	uint8_t	flash_type	Sets the activation/deactivation for the fash programming mode. 0x00: deactivating flash programming mode. 0x10: activating internal flash programming mode. 0x11: activating external flash programming mode.
Return value	int	Returns 0 for normal termination, otherwise returns an error code.	
Description	This function activates/deactivates internal/external flash programming mode.		
Note			

4.1.16 seSpeechEraseFlash

Function	int seSpeechEraseFlash (SPEECH *speech, uint8_t erase_type, uin32_t address)		
Argument	SPEECH_T	*speech	Pointer to the SPEECH structure.
	uint8_t	erase_type	Sets the erase type. 0: chip erase 1: sector erase
	uint32_t	address	Set thea address of the sector erase area Note: Set to 0 for chip erase.
Return value	int	Returns 0 for normal termination, otherwise returns an error code.	
Description	This function erases the specified flash area.		
Note	This function can only be used in internal/external flash programming mode.		

4.1.17 seSpeechWriteFlash

Function	int seSpeechWriteFlash (SPEECH *speech, uint32_t address, uint8_t *wdata, uint16_t size)		
Argument	SPEECH_T	*speech	Pointer to the SPEECH structure.
	uint32_t	address	Sets the address of the write flash area
	uint8_t	*wdata	Sets the pointer to write data
	uint16_t	size	Sets the size of write data MAX: 1024 bytes
Return value	int	Returns 0 for normal termination, otherwise returns an error code.	
Description	This function writes data to the specified flash area.		
Note	This function can only be used in internal/external flash programming mode.		

4.1.18 seSpeechReadFlash

Function	int seSpeechReadFlash (SPEECH *speech, uint32_t address, uint8_t *rdata, uint16_t size)		
Argument	SPEECH_T	*speech	Pointer to the SPEECH structure.
	uint32_t	address	Sets the address of the read flash area
	uint8_t	*rdata	Sets the pointer to memory to store read data
	uint16_t	size	Sets the size of read data MAX:1024
Return value	int	Returns 0 for normal termination, otherwise returns an error code.	
Description	This function reads data from the specified flash area.		
Note	This function can only be used in internal/external flash programming mode.		

4.1.19 seSpeechEnterSleep

Function	int seSpeechEnterSleep (uint8_t sleep_type)		
Argument	uint8_t	sleep_type	Sets the type of stand-by mode 0: sleep 1: deep sleep
Return value	int	Returns 0 for normal termination, otherwise returns an error code.	
Description	This function enters sleep mode.		
Note			

4.1.20 seSpeechExitSleep

Function	int seSpeechExitSleep (SPEECH *speech)		
Argument	SPEECH_T	*speech	Pointer to the SPEECH structure.
Return value	int	Returns 0 for normal termination, otherwise returns an error code.	
Description	This function exits sleep mode.		
Note			

4.1.21 seSpeechSetKeyCode

Function	int seSpeechSetKeyCode (SPEECH *speech, uint32_t keycode)		
Argument	SPEECH_T	*speech	Pointer to the SPEECH structure.
	uint32_t	keycode	Sets the Keycode (32bits value required to read sound data from sound ROM)
Return value	int	Returns 0 for normal termination, otherwise returns an error code.	
Description	This function sets keycode in internal flash.		
Note	This function also erases the upper 128 bytes of the setting information area.		

4. Function Specification

4.1.22 seSpeechEraseSettingFlash

Function	int seSpeechEraseSettingFlash (SPEECH *speech)		
Argument	SPEECH_T	*speech	Pointer to the SPEECH structure.
Return value	int	Returns 0 for normal termination, otherwise returns an error code.	
Description	This function erases the setting information area.		
Note	This function can only be used in internal flash programming mode.		

4.1.23 seSpeechWriteSettingFlash

Function	int seSpeechWriteSettingFlash (SPEECH *speech , uint8_t *wdata)		
Argument	SPEECH_T	*speech	Pointer to the SPEECH structure.
	uint8_t	*wdata	Sets the pointer to write data (256 bytes of setting information).
Return value	int	Returns 0 for normal termination, otherwise returns an error code.	
Description	This function writes the setting information for standalone mode to the setting information area in internal flash.		
Note	This function can only be used in internal flash programming mode.		

4.1.24 seSpeechReadSettingFlash

Function	int seSpeechReadSettingFlash (SPEECH *speech, uint8_t *rdata)		
Argument	SPEECH_T	*speech	Pointer to the SPEECH structure.
	uint8_t	*rdata	Sets the pointer to memory to read data (256 bytes of setting information).
Return value	int	Returns 0 for normal termination, otherwise returns an error code.	
Description	This function reads the setting information from the setting information area in internal flash.		
Note	This function can only be used in internal flash programming mode.		

4.1.25 seSpeechExecuteSelfCheck

Function	int seSpeechSelfCheck (SPEECH *speech)		
Argument	SPEECH_T	*speech	Pointer to the SPEECH structure.
Return value	int	Returns 0 for normal termination, otherwise returns an error code.	
Description	This function executes the self-check.		
Note			

4.1.26 seSpeechExecuteFlashCrcCheck

Function	int seSpeechExecuteFlashCrcCheck (SPEECH *speech, uint32_t address, uint32_t numbytes, uint8_t crc)		
Argument	SPEECH_T	*speech	Pointer to the SPEECH structure.
	uint32_t	address	Sets the start address of crc check area
	uint32_t	numbytes	Sets the number of bytes
	uint8_t	crc	Sets the CRC value of data
Return value	int	Returns 0 for normal termination, otherwise returns an error code.	
Description	This function executes the CRC check on the data from the specified address to the specified size.		
Note	To use this function, the CRC value must be set to a pre-calculated value. This function can only be used in internal/external flash programming mode.		

4. Function Specification

4.2. Serial I/F function

The serial I/F function are used to execute serial communication between the host and S1V3F351/S1V3F352.

These functions are defined in the serial.c/h file.

4.2.1 seSerialInitialize

Function	int seSerialInitialize (uint8_t serial_type, uint32_t serial_clock, uint16_t uart_config)		
Argument	uint8_t	serial_type	Serial I/F type. 0: SPI / 1: UART / 2: I2C
	uint32_t	serial_clock	Serial I/F clock. SPI: Max.400,000 Hz I2C: Max.300,000 Hz UART: 9600,19200,38400,57600,115200,230400 bps
	uint16_t	uart_config	UART communication parameters. [10:9] Parity 0: none / 1: even / 2: none / 3: odd [8] Stop Bit 0: 1bit / 1: 2bit Note1: Please set 0 to bits other than those listed above. Note2: Please set 0x0000 for SPI and I2C.
Return value	int	Returns 0 for normal termination, otherwise returns an error code.	
Description	This function initializes the serial I/F control circuit on host CPU.		
Note			

4.2.2 seSpilnitialize

Function	int seSpilnitialize (uint32_t clock)		
Argument	uint32_t	clock	SPI clock Max.400,000 Hz
Return value	int	Returns 0 for normal termination, otherwise returns an error code.	
Description	This function initializes the SPI control circuit on the host.		
Note			

4.2.3 sel2cInitalize

Function	int sel2cInitalize (uint32_t clock)		
Argument	uint32_t	clock	I2C clock Max.300,000 Hz
Return value	int	Returns 0 for normal termination, otherwise returns an error code.	
Description	This function initializes the I2C control circuit on the host.		
Note			

4.2.4 seUartInitalize

Function	int seUartInitalize (uint32_t clock, uint16_t uart_config)		
Argument	uint32_t	clock	UART clock(baudrate) 9600,19200,38400,57600,115200,230400 bps
	uint16_t	uart_config	UART communication parameters. [10:9] Parity 0: none / 1: even / 2: none / 3: odd [8] Stop Bit 0: 1bit / 1: 2bit Note: Please set 0 to bits other than those listed above.
Return value	int	Returns 0 for normal termination, otherwise returns an error code.	
Description	This function initializes the UART control circuit on the host.		
Note			

4. Function Specification

4.2.5 seSerialSendRecvMessage

Function	int seSerialSendRecvMessage (uint8_t serial_type, uint8_t *msg, uint16_t msgsize, uint8_t *resp, uint16_t respsize)		
Argument	uint8_t	serial_type	Serial I/F type 0: SPI / 1: UART / 2: I2C
	uint8_t	*msg	Message to send
	uint16_t	msgsize	Size of send message
	uint8_t	*resp	Memory address to store received response data
	uint16_t	respsize	Size of received response data
Return value	int	Returns 0 for normal termination, otherwise returns an error code.	
Description	This function sends and receives messages between the host and S1V3F351/S1V3F352.		
Note			

4.2.6 seSpiSendRecvMessage

Function	int seSpiSendRecvMessage (uint8_t *msg, int msgsize, uint8_t *resp, int respsize)		
Argument	uint8_t	*msg	Message to send
	int	msgsize	Size of send message
	uint8_t	*resp	Memory address to store received response data
	int	respsize	Size of received response data
Return value	int	Returns 0 for normal termination, otherwise returns an error code.	
Description	This function sends and receives messages between the host and S1V3F351/S1V3F352 using SPI.		
Note			

4.2.7 sel2cSendRecvMessage

Function	int sel2cSendRecvMessage (uint8_t *msg, int msgsize, uint8_t *resp, int respsize)		
Argument	uint8_t	*msg	Message to send
	int	msgsize	Size of send message
	uint8_t	*resp	Memory address to store received response data
	int	respsize	Size of received response data
Return value	int	Returns 0 for normal termination, otherwise returns an error code.	
Description	This function sends and receives messages between the host and S1V3F351/S1V3F352 using I2C.		
Note			

4.2.8 seUartSendRecvMessage

Function	int seUartSendRecvMessage (uint8_t *msg, int msgsize, uint8_t *resp, int respsize)		
Argument	uint8_t	*msg	Message to send
	int	msgsize	Size of send message
	uint8_t	*resp	Memory address to store received response data
	int	respsize	Size of received response data
Return value	int	Returns 0 for normal termination, otherwise returns an error code.	
Description	This function sends and receives messages between the host and S1V3F351/S1V3F352 using UART.		
Note			

4. Function Specification

4.3. IND Messages Function

The IND messages function used by the host to request status and data back from S1V3F351/S1V3F352.

These functions are defined in the serial.c/.h file.

4.3.1 m_ISC_STATUS_IND

関数名	msgIndResult m_ISC_STATUS_IND (uint8_t indType, uint32_t rxSize)		
引数	uint8_t	indType	Indication Type (IND) 0x00: Error/Warning Status 0x01: Sound Operation State 0x02: CRC Setting 0x03: Sound Effect Settings 0x04: Sound ROM Settings 0x06: Serial Flash Read ID Status 0x07: Serial Flash Read Register Status * Settings other than the above are prohibited.
	uint32_t	rxSize	Size of received response data
戻り値	msgIndResult	Returns the received data.	
機能	This function requests status or data according to the specified IND type.		
備考			

4.3.2 m_IND_ERROR_WARNING

関数名	msgErrorWarning m_IND_ERROR_WARNING ()		
引数			
戻り値	msgErrorWarning	Returns the received data.	
機能	This function reads Error/Warning Status.		
備考			

4.3.3 m_IND_SOUND_OPERATION_STATE

関数名	msgSoundOperationState m_IND_SOUND_OPERATION_STATE ()		
引数			
戻り値	msgSoundOperationState	Returns the received data.	
機能	This function reads Sound Operation State.		
備考			

4.3.4 m_IND_SOUND_OUTPUT_STATE

関数名	msgSoundOutputState m_IND_SOUND_OUTPUT_STATE ()	
引数		
戻り値	msgSoundOutputState	Returns the received data.
機能	This function reads Sound Output State.	
備考		

4.3.5 m_IND_CRC_SETTING

関数名	msgCrcSetting m_IND_CRC_SETTING ()	
引数		
戻り値	msgCrcSetting	Returns the received data.
機能	This function reads CRC Setting.	
備考		

4.3.6 m_IND_SOUND_EFFECT_SETTINGS

関数名	msgSoundEffectSettings m_IND_SOUND_EFFECT_SETTINGS ()	
引数		
戻り値	msgSoundEffectSettings	Returns the received data.
機能	This function reads Sound Effect Settings.	
備考		

4.3.7 m_IND_SOUND_ROM_SETTINGS

関数名	msgSoundRomSettings m_IND_SOUND_ROM_SETTINGS ()	
引数		
戻り値	msgSoundRomSettings	Returns the received data.
機能	This function reads SoundROM Settings.	
備考		

5. Structure

5. Structure

5.1 SPEECH

Structure	<pre> typedef struct { uint8_t serial_type; uint32_t serial_clock; uint16_t uart_config; uint8_t output_mode; uint8_t sampling_rate; uint8_t flash_type; uint32_t rom_address; uint32_t rom_size; uint32_t keyocode; uint8_t volume[2]; uint8_t speed; uint8_t pitch; uint16_t sentence_no[2]; uint8_t repeat[2]; uint16_t freq[4]; uint8_t duration[4]; uint8_t delay[3]; uint8_t pattern_repeat; } SPEECH; </pre>		
Member variables	uint8_t	serial_type	Serial I/F type
	uint32_t	serial_clock	Serial I/F clock
	uint16_t	uart_config	UART communication parameters
	uint8_t	output_mode	Sound output mode
	uint8_t	sampling_rate	Sampling Frequency
	uint8_t	flash_type	Type of sound ROM storage area
	uint32_t	rom_address	Address of sound ROM storage area
	uint32_t	rom_size	Size of sound ROM
	uint32_t	keyocode	Keycode
	uint8_t	volume[2]	Sound volume
	uint8_t	speed	Sound playback speed
	uint8_t	pitch	Sound playback pitch
	uint16_t	sentence_no[2]	Sentence number
	uint8_t	repeat[2]	Repeat count
	uint16_t	freq[4]	Tone frequency
	uint8_t	duration[4]	Tone playback duration
	uint8_t	delay[4]	Delay duration before next tone starts playback
	uint8_t	pattern_repeat	Repeat count of tone pattern
Description	This structure is required to used Speech LSI control function. Various information such as serial I/F, audio output mode, audio ROM data address, and audio playback parameters, etc. are stored.		
Note			

5.2 msgResult

Structure	typedef struct { uint8_t RECV_STATUS; uint8_t STATUS; uint8_t ERROR; } msgResult;		
Member variables	uint8_t	RECV_STATUS	Receiving status
	uint8_t	STATUS	STATUS signal state
	uint8_t	ERROR	ERROR signal state
Description	This structure is used in communication with S1V3F351/S1V3F352. The results of communication are stored.		
Note			

5.3 msgIndResult

Structure	typedef struct { uint8_t RECV_STATUS; uint8_t STATUS; uint8_t ERROR; uint8_t* RECV_DATA; } msgResult;		
Member variables	uint8_t	RECV_STATUS	Receiving status
	uint8_t	STATUS	STATUS signal state
	uint8_t	ERROR	ERROR signal state
	uint8_t	RECV_DATA	Received data
Description	This structure is used in communication with S1V3F351/S1V3F352. The results of IND communication are stored.		
Note			

5.4 msgErrorWarning

Structure	typedef struct { msgResult Result; uint16_t Error0; uint16_t Error1; } msgErrorWarning;		
Member variables	msgResult	Result	Communication result
	uint16_t	Error0	ERROR0 status
	uint16_t	Error1	ERROR1 status
Description	This structure is used in communication with S1V3F351/S1V3F352. Various information obtained through IND communication is stored.		
Note			

5. Structure

5.5 msgSoundOperationState

Structure	<pre>typedef struct { msgResult Result; uint16_t StateCh0; uint16_t StateCh1; } msgSoundOperationState;</pre>		
Member variables	msgResult	Result	Communication result
	uint16_t	StateCh0	Ch.0 operation state
	uint16_t	StateCh1	Ch.1 operation state
Description	This structure is used in communication with S1V3F351/S1V3F352. Various information obtained through IND communication is stored.		
Note			

5.6 msgSoundOutputState

Structure	<pre>typedef struct { msgResult Result; uint16_t StateCh0; uint16_t StateCh1; uint8_t Toneon; uint8_t Status; } msgSoundOutputState;</pre>		
Member variables	msgResult	Result	Communication result
	uint16_t	StateCh0	Ch.0 operation state
	uint16_t	StateCh1	Ch.1 operation state
	uint8_t	ToneOn	Tone output state
	uint8_t	Status	STATUS signal state
Description	This structure is used in communication with S1V3F351/S1V3F352. Various information obtained through IND communication is stored.		
Note			

5.7 msgCrcSetting

Structure	<pre>typedef struct { msgResult Result; uint8_t CrcSetting; } msgCrcSetting;</pre>		
Member variables	msgResult	Result	Communication result
	uint8_t	CrcSetting	CRC check setting state
Description	This structure is used in communication with S1V3F351/S1V3F352. Various information obtained through IND communication is stored.		
Note			

5.8 msgSoundEffectSettings

Structure	<pre>typedef struct { msgResult Result; uint16_t VolumeCh0; uint16_t VolumeCh1; uint16_t SpeedCh0; uint16_t PitchCh0; uint16_t ToneFreq; uint16_t ToneOn; uint16_t SndOutSelect; } msgSoundEffectSettings;</pre>		
Member variables	msgResult	Result	Communication result
	uint8_t	VolumeCh0	Ch.0 Volume setting state
	uint8_t	VolumeCh1	Ch.1 Volume setting state
	uint8_t	SpeedCh0	Ch.0 Playback speed setting state
	uint8_t	PitchCh0	Ch.0 Playback pitch setting state
	uint16_t	ToneFreq	Currently / previously output tone frequency
	uint8_t	ToneOn	Tone output state
	uint8_t	SndOutSelect	Sound output selection state
Description	This structure is used in communication with S1V3F351/S1V3F352. Various information obtained through IND communication is stored.		
Note			

5.9 msgSoundRomSettings

Structure	<pre>typedef struct { msgResult Result; uint16_t RomAddr; uint16_t RomSize; uint16_t FlashSelect; } msgSoundRomSettings;</pre>		
Member variables	msgResult	Result	Communication result
	uint32_t	RomAddr	Sound ROM start address
	uint32_t	RomSize	Sound ROM size
	uint8_t	FlashSelect	Selected flash memory
Description	This structure is used in communication with S1V3F351/S1V3F352. Various information obtained through IND communication is stored.		
Note			

America

Epson America, Inc.

Headquarter:
3131 Katella Ave.
Los Alamitos, CA 90720, USA
Phone: +1-800-463-7766

San Jose Office:
2860 Zanker Road Suite 204
San Jose, CA 95134, USA
Phone: +1-800-463-7766

Europe

Epson Europe Electronics GmbH

Riesstrasse 15, 80992 Munich,
Germany
Phone: +49-89-14005-0 FAX: +49-89-14005-110

Asia

Epson (China) Co., Ltd.

4th Floor, Tower 1 of China Central Place, 81 Jianguo Road, Chaoyang
District, Beijing 100025 China
Phone: +86-10-8522-1199 FAX: +86-10-8522-1120

Shanghai Branch

Room 601-603, Building A One East, No.325 East Longhua Road,
Shanghai 200023, China
Phone: +86-21-5330-4888 FAX: +86-21-5423-4677

Shenzhen Branch

Room 804-805, 8 Floor, Tower 2, Ali Center, No.3331
Keyuan South RD(Shenzhen bay), Nanshan District, Shenzhen
518054, China
Phone: +86-755-3299-0588 FAX: +86-755-3299-0560

Epson Taiwan Technology & Trading Ltd.

15F, No.100, Songren Rd, Sinyi Dist, Taipei City 110. Taiwan
Phone: +886-2-8786-6688

Epson Singapore Pte., Ltd.

438B Alexandra Road,
Block B Alexandra TechnoPark, #04-01/04, Singapore 119968
Phone: +65-6586-5500 FAX: +65-6271-7066

Epson Korea Co.,Ltd

10F Posco Tower Yeoksam, Teheranro 134 Gangnam-gu,
Seoul, 06235, Korea
Phone: +82-2-3420-6695

Seiko Epson Corp.

Sales & Marketing Division

MD Sales & Marketing Department

JR Shinjuku Miraina Tower, 4-1-6 Shinjuku,
Shinjuku-ku, Tokyo 160-8801, Japan