EPSON
EXCEED YOUR VISION

S1V30080 Series
Message Protocol
Specification

SEIKO EPSON CORPORATION

Rev. 1.00

NOTICE

Table of Contents

1. Introduction 1

1.1 Scope 1

1.2 Document structure 1

1.3 Terminology 1

2. Features 2

2.1 Audio playback 2

2.2 Synthesizer playback 4

2.3 Audio playback and sound synthesizer playback mixing 4

2.4 One-command playback function 4

2.5 SPI interface 4

2.6 I2C interface 4

2.7 Standalone interface 5

2.8 External serial flash memory interface 5

2.9 Standby mode 5

2.10 Sleep mode 5

2.11 10-bit DAC 5

3. Interface Specifications 6

3.1 Common settings 6

3.1.1 Initialization information 6

3.1.2 Memory storage commands 7

3.2 SPI interface mode 8

3.3 I2C interface mode 9

3.4 Standalone 1 interface mode 11

3.5 Standalone 2 interface mode 12

3.6 External serial flash memory 13

4. Message Protocol 14

4.1 ISC message basic specifications 14

4.2 ISC message basic structure 16

4.3 ISC message basic flow 17

4.3.1 Updating initialization information 17

4.3.2 One-command playback 17

4.3.3 Audio playback 18

4.3.4 Synthesizer playback 18

4.3.5 Synthesizer melody playback (streaming playback) 19

4.3.6 Mixing playback 21

4.3.7 Mixing playback (using one-command playback) 22
4.4 Checksum 23
5. Message Details 24
5.1 Message identifier list 24
5.2 System control messages 25
5.2.1 ISC_CLKDIV_CONFIG_REQ 25
5.2.2 ISC_CHKSUM_CONFIG_REQ 26
5.3 Sound output setting messages 27
5.3.1 ISC_AUDIO_CONFIG_REQ 27
5.4 Audio playback setting messages 28
5.4.1 ISC_SEQUENCER_MAIN_CONFIG_REQ 28
5.4.2 ISC_SEQUENCER_MULTI_CONFIG_REQ 29
5.5 Synthesizer playback setting messages 31
5.5.1 ISC_SYNTHESIZER_MELODY_CONFIG_REQ 31
5.5.2 ISC_SYNTHESIZER_MELODY_START_REQ 33
5.5.3 ISC_SYNTHESIZER_MELODY_SET_REQ 34
5.5.4 ISC_SYNTHESIZER_BUZZ_TONE_SET_REQ 35
5.6 Sound playback control messages 37
5.6.1 ISC_EASY_START_REQ 37
5.6.2 ISC_SOUND_START_REQ 38
5.6.3 ISC_SOUND_STOP_REQ 39
5.6.4 ISC_AUDIO_VOLUME_REQ 40
5.6.5 ISC_AUDIO_MUTE_REQ 41
5.7 Status confirmation messages 42
5.7.1 ISC_STATUS_REQ 42
5.7.2 ISC_STATUS_RESP 43
Revision History 47

1. Introduction

1.1 Scope

This document specifies the interface and message protocol used to control the S1V30080.

If the interface mode used is Standalone 1 or 2, refer in particular to Section 2, “Features” and Section 3, “Interface Specifications.”

1.2 Document structure

Section 2 describes the features of the S1V30080.

Section 3 describes the specifications for the interface selected by the user.

Section 4 describes the basic specifications of the message protocol used to communicate between the host and the S1V30080 and outlines the basic message flow.

Section 5 describes the details of the messages supported by the S1V30080.

1.3 Terminology

The terminology used in this specification manual is defined as shown below.

Term	Explanation
Audio main playback	Playback set by ISC_SEQUENCER_MAIN_CONFIG_REQ or memory storage main command
Audio multi playback	Playback set by ISC_SEQUENCER_MULTI_CONFIG_REQ or memory storage multi command
Audio playback	General term for audio main and audio multi playback
Synthesizer melody playback	Playback set by ISC_SYNTHESIZER_MELODY_*_REQ or memory storage melody command
Synthesizer buzzer/tone playback	Playback set by ISC_SYNTHESIZER_BUZZ_TONE_SET_REQ or memory storage buzzer/tone command
Synthesizer playback	General term for synthesizer melody and synthesizer buzzer/tone playback
Audio data	ERV (Epson Raw Voice) data
Sound	General term for audio output by the S1V30080
File_Index_Number	Index number for the audio data or memory storage commands
Mixing playback	State in which at least two of audio main playback, audio multi playback, and synthesizer playback are in progress

2. Features

The S1V30080 includes the following features:

- Audio playback
- Synthesizer playback
- Audio playback and synthesizer playback mixing
- One-command playback function
- SPI interface
- I2C interface
- Standalone interface
- External serial flash memory interface
- Standby mode
- Sleep mode
- 10-bit DAC

This section describes these features.

2.1 Audio playback

Audio playback allows the following settings:

- Multiple audio data playback sequence
- Mute interval between individual phrases
- Number of repeats

The S1V30080 internal ROM contains the data provided in advance by the customer. The host device selects the audio data to be played back from among this data, and specifies the playback sequence to the S1V30080. The audio data specified is automatically played back continuously once the host device instructs the S1V30080 to start playback. This playback method is defined as *audio playback*.

The S1V30080 supports the following two types of audio playback.

- Audio main playback (1ch playback)
- Audio multi playback (5ch playback)

Audio main playback supports 1-channel playback, while audio multi playback supports 5-channel playback.

These two playback methods are independent, enabling the volume setting and playback start timing to be controlled separately. The five channels for audio multi playback use the same playback method, and so the same volume and playback start timing settings apply to all five channels.

Figure 2.1 illustrates the audio main playback arrangement.

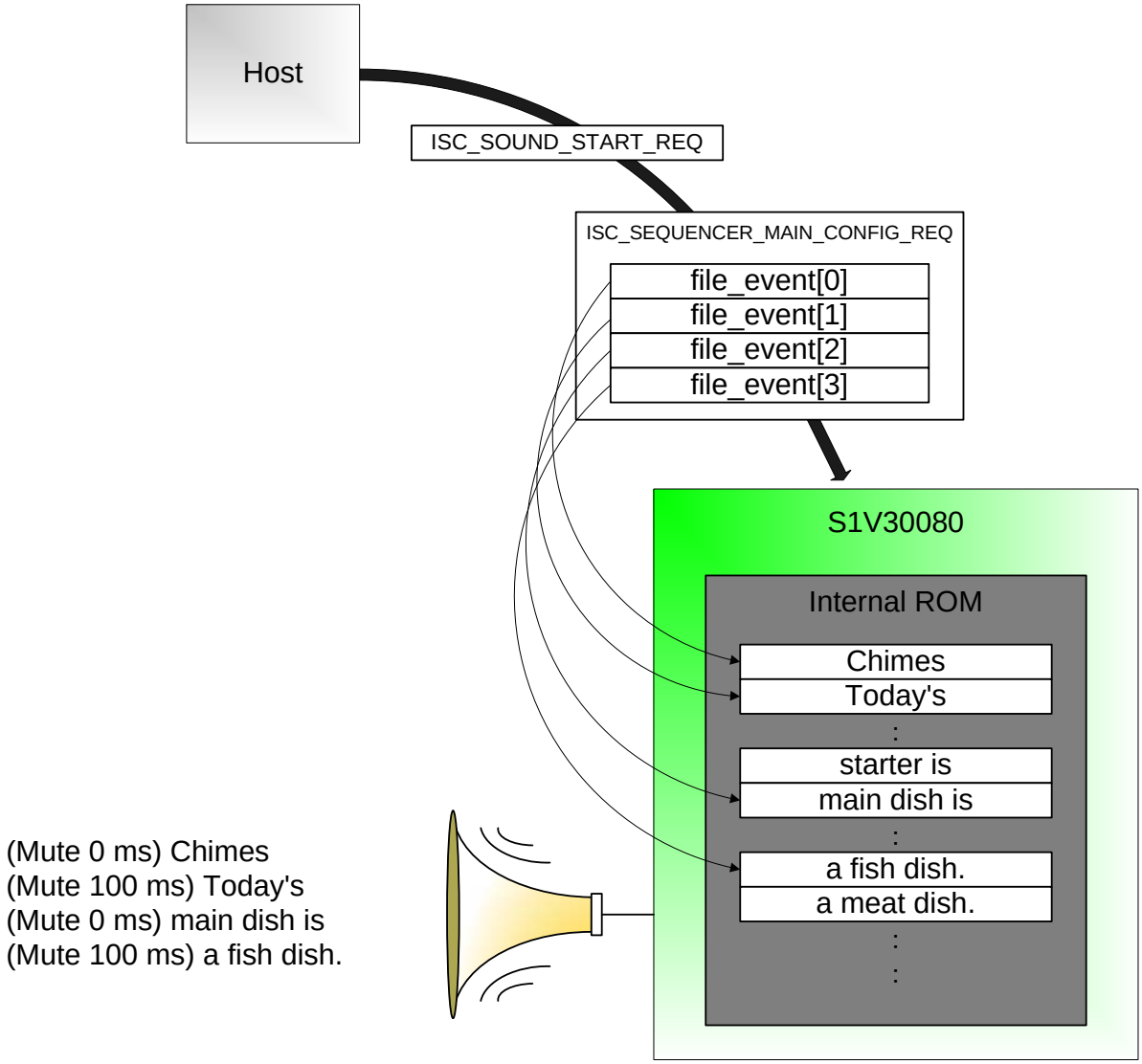


Figure 2.1 Audio main playback

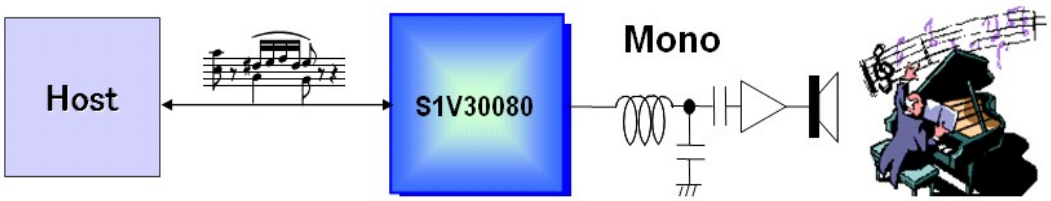
2.2 Synthesizer playback

The S1V30080 includes a built-in synthesizer function. The synthesizer function generates melodies or buzzer tones, and supports playback of the following:

- Synthesizer melody playback
- Synthesizer buzzer/tone playback

Synthesizer melody playback involves generating a melody from score data (1 byte per note) sent from the host device or from score data specified by the memory storage melody command. Five channels of simultaneous playback are possible, with individual volume settings for each channel. Parameters such as melody tempo and envelope can also be set.

Synthesizer buzzer/tone playback involves generating the desired buzzer tone from frequency data (2 bytes per playback tone) sent from the host device or from frequency data specified by the memory storage buzzer/tone command. Up to five channels can be played back at once, with individual volume settings for each channel.



2.3 Audio playback and sound synthesizer playback mixing

The audio playback and synthesizer playback functions are completely independent, enabling both to be mixed with separate volume settings.

2.4 One-command playback function

The S1V30080 features a one-command playback function to reduce the load on the host device. This function starts playback when the ISC_EASY_START_REQ message is sent. The one-command playback function is used with memory storage commands created by the “S1V30080 Series Sound Tool.” For more details, refer to Section 3.1.2, “Memory storage commands.”

2.5 SPI interface

The 3-line (SCKS, SIS, and SOS) SPI (slave) interface mode is supported for communication with the host. The MSG_RECEIVE signal, indicating message receipt, and the SOUND_PLAYING signal, indicating sound output in progress, are also used.

For more details, refer to Section 3.2, “SPI interface mode.”

2.6 I2C interface

I2C (slave) is supported as an interface mode with the host. The MSG_RECEIVE signal, indicating message receipt, and the SOUND_PLAYING signal, indicating sound output in progress, are also used.

For more details, refer to Section 3.3, “I2C interface mode.”

2.7 Standalone interface

A standalone interface is also supported for easy playback based on data stored in the memory, by setting input pin SET_PLAY[3:0]. The standalone interface includes one of two different modes, depending on the customer's usage circumstances.

For more details, refer to Section 3.4, "Standalone 1 interface mode" and Section 3.5, "Standalone 2 interface mode."

2.8 External serial flash memory interface

The S1V30080 allows access to the external serial flash memory (for the QFP12-48 package only). Setting the FLASH_EN pin to High allows playback from the external serial flash memory instead of the internal ROM.

For more details, refer to Section 3.6, "External serial flash memory."

2.9 Standby mode

Setting the CE pin to High and the SYSTEM_EN pin to Low invokes Standby mode. The input clock can also be gated off to reduce current consumption while the SYSTEM_EN pin is set to Low.

Resetting from Standby mode is possible only by setting the SYSTEM_EN pin to High.

For details of the SYSTEM_EN pin and CE pin control timing, refer to the *S1V30080 Series Hardware Specifications*.

2.10 Sleep mode

Sleep mode can be set by setting the CE pin to Low and the SYSTEM_EN pin to Low. The input clock is gated while the SYSTEM_EN pin is set to Low.

The internal regulator is turned off while the CE pin is set to Low, allowing the current consumption to be further reduced compared to Standby mode.

For details of the SYSTEM_EN pin and CE pin control timing, refer to the *S1V30080 Series Hardware Specifications*.

2.11 10-bit DAC

A 10-bit DAC is built in for sound output. 10-bit or 8-bit resolution can be selected using the parameters set by ISC_AUDIO_CONFIG_REQ.

3. Interface Specifications

3.1 Common settings

The S1V30080 supports four different interface types: SPI, I2C, Standalone 1, and Standalone 2. These interfaces can be selected using the "S1V30080 Series Sound Tool," and are achieved by storing interface information in the S1V30080 internal ROM or external serial flash memory.

The initialization information and memory storage commands described below are stored in the memory as settings common for each interface.

3.1.1 Initialization information

Initialization information is necessary to ensure that the S1V30080 operates correctly. Initialization information encompasses the items listed in Table 3.1, and the settings created using the "S1V30080 Series Sound Tool" are stored in the memory.

The initialization information can be updated using the ISC message shown in Table 3.1 when using the SPI or I2C interface (refer to 4.3.1). The updated initialization information is enabled provided the SYSTEM_EN pin is not set to Low. A hardware reset will occur if the SYSTEM_EN pin is set to Low.

Note that initialization information cannot be updated when using Standalone 1 or Standalone 2, since ISC messages are not used.

Table 3.1 Initialization information

Initialization information	Corresponding ISC message
Clock division	ISC_CLKDIV_CONFIG_REQ
Sampling frequency	ISC_AUDIO_CONFIG_REQ
Channel volume	ISC_AUDIO_CONFIG_REQ
DAC bit width	ISC_AUDIO_CONFIG_REQ
Synthesizer melody playback tempo	ISC_SYNTHESIZER_MELODY_CONFIG_REQ
Synthesizer melody playback envelope	ISC_SYNTHESIZER_MELODY_CONFIG_REQ

3.1.2 Memory storage commands

Memory storage commands are commands that have been pre-stored in the internal ROM or external flash memory, corresponding to the following four ISC message types:

ISC_SEQUENCER_MAIN_CONFIG_REQ

ISC_SEQUENCER_MULTI_CONFIG_REQ

ISC_SYNTHESIZER_MELODY_SET_REQ

ISC_SYNTHESIZER_BUZZ_TONE_SET_REQ

Table 3.2 describes the function of each command. The method for invoking a memory storage command depends on the interface mode, as described below:

For SPI or I2C interface

Specifying a particular command's File_Index_Number in the ISC_EASY_START_REQ message starts playback corresponding to that command. The ISC_SOUND_START_REQ message is not required here.

For Standalone 1 or 2 interface

Specifying a particular command's File_Index_Number using the SET_PLAY[0] to [3] pins starts playback corresponding to that command. (Refer to Section 3.4 or Section 3.5.)

The "S1V30080 Series Sound Tool" should be used to create memory storage commands. Up to 255 memory storage commands and audio data can be created. Create data so that the number of memory storage commands plus audio data is within 255.

For details of the File_Index_Number associated with each memory storage command, refer to the "ROMImage_yymmdd_hhmmss.csv" file output by the "S1V30080 Series Sound Tool."

Table 3.2 Memory storage command types

Memory storage command type	Operation
Memory storage main command	Starts audio main playback.
Memory storage multi command	Starts audio multi playback.
Memory storage melody command	Starts synthesizer melody playback.
Memory storage buzzer/tone command	Starts synthesizer buzzer/tone playback.

3.2 SPI interface mode

When using SPI interface mode, the S1V30080 operates as a slave device. Therefore, a clock signal must be fed to the SCKS pin by the host device. Data transfer uses 8-bit word transfer. The non-active state for the SCKS pin is the logical value 1.

The host device can send messages to the S1V30080 via the SIS pin, and can receive messages from the S1V30080 via the SOS pin when necessary (for ISC_STATUS_REQ message). A clock signal must be fed to the SCKS pin for the host device to receive messages from the S1V30080.

For physical details, such as data setup time, hold time, and transfer rates, refer to the *S1V30080 Series Hardware Specifications*.

Table 3.3 Pins supporting SPI interface mode

Interface mode	SCKS	SIS	SOS	MSG_RECIEVE	Data output sequence
SPI	Serial clock input	Data input	Data output	MSG_RECEIVE output	msb first

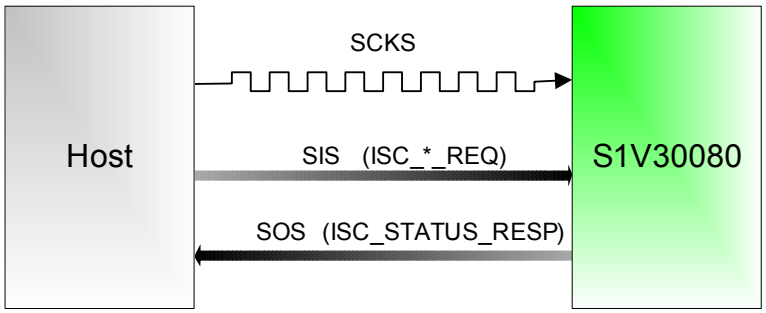


Figure 3.1 Host – S1V30080 SPI connection

3.3 I2C interface mode

When using I2C interface mode, data transfer uses MSB-first 8-bit word transfer. The SCL non-active state and SDA initial values are the logical value 1.

I2C transfer starts when the host changes SDA from High to Low when SCL is High. These are the start conditions. SDA can normally be changed only when SCL is Low, but this is overruled when the start conditions are satisfied. Once the start conditions have been satisfied, the host should send the S1V30080 7-bit slave ID (0110110) to the SDA. The next bit should be Low if the host is sending data, and High if receiving data. Concequently, if the S1V30080 is ready to receive or send, set the SDA to Low. This switching of the SDA to Low after receiving data by the device forms an Ack. Setting the SDA instead to High forms a Nack.

■ REQ message transmission (Refer to Figure 3.3)

When sending a REQ message, the host can start to send the REQ message stipulated in this specification manual after the 1-byte padding word (0x00) and 1-byte start message (0xAA).The S1V30080 returns an Ack for each 8 bits of data received.

To end transmission of REQ messages by the host, change the SDA from Low to High while the SCL is High after the S1V30080 has returned an Ack. These are the stop conditions. SDA can normally be changed only when SCL is Low, but this is overruled when the stop conditions are satisfied.

■ ISC_STATUS_RESP message receipt (Refer to Figure 3.4)

When receiving an ISC_STATUS_RESP message, the S1V30080 starts transmission of the ISC_STATUS_RESP message stipulated in this specification manual after transmitting the 1-byte padding word (0x00) and 1-byte start message (0xAA). The host should return an Ack for each 8 bits of data received. To end receipt of ISC_STATUS_RESP messages by the host, return a Nack after receiving 8 bits of data. In addition, the stop conditions must be met.

For physical details, such as data setup time, hold time, and transfer rates, refer to the *S1V30080 Series Hardware Specifications*.

Table 3.4 Pins supporting I2C interface mode

Interface mode	SCKS	SIS	SOS	MSG_RECIEVE	Data output sequence
I2C	Serial clock input (SCL)	Data input/output (SDA)	Input (Fixed at Low for IC external)	MSG_RECEIVE output	msb first

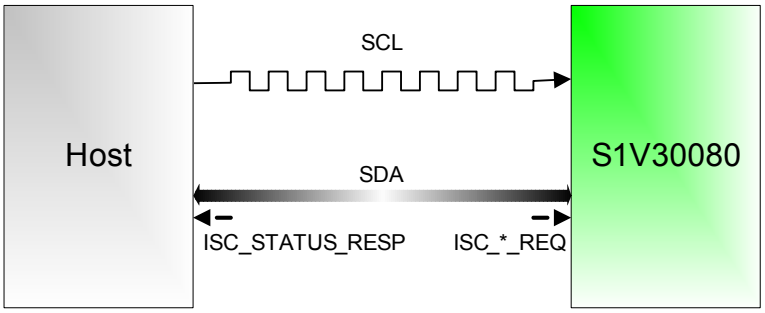


Figure 3.2 Host – S1V30080 I2C connection

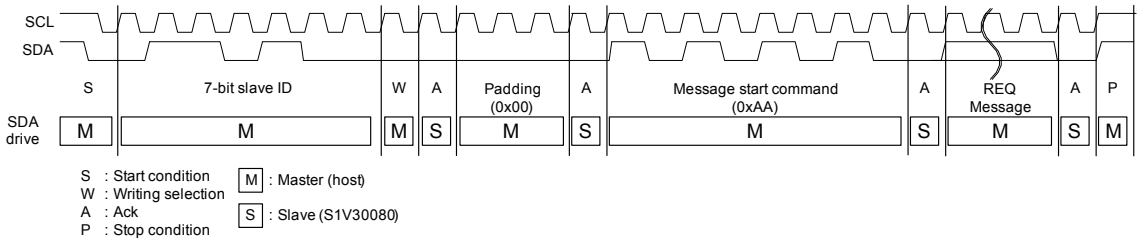


Figure 3.3 I2C REQ message timing chart

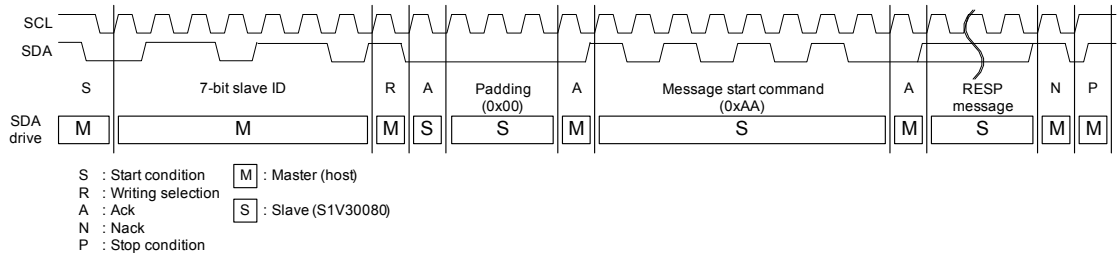


Figure 3.4 I2C ISC_STATUS_RESP message timing chart

3.4 Standalone 1 interface mode

When using the Standalone 1 interface mode, ISC messages are not required. This mode allows playback by controlling the SIS, SCKS, SOS, and MSG_RECEIVE pins shown in Table 3.5. The initial state for these four pins is logical value 1. Enter the File_Index_Number to be played back as shown in Figure 3.5 and Table 3.6, and specify the memory storage command to start playback. Note the following points in Standalone 1 mode:

- File_Index_Number is confirmed by Low input for 50 ms or more, followed by a rising edge.
- File_Index_Number can be specified in the range 0 to 14.
- Sound that can be played back consists of memory storage commands only.
- Entering File_Index_Number during playback will produce mixing playback. Note that overwrite playback will result if the same type of memory storage command is specified. (Memory storage melody command and memory storage buzzer/tone command are considered the same).
- Initialization information cannot be changed.

Table 3.5 Pins supporting Standalone 1

Interface mode	SIS	SCKS	SOS	MSG_RECIEVE	Outline
Standalone 1	SET_PLAY[0] 0x1 (hex) (input)	SET_PLAY[1] 0x2 (hex) (input)	SET_PLAY[2] 0x4 (hex) (input)	SET_PLAY[3] 0x8 (hex) (input)	Simple decode setting

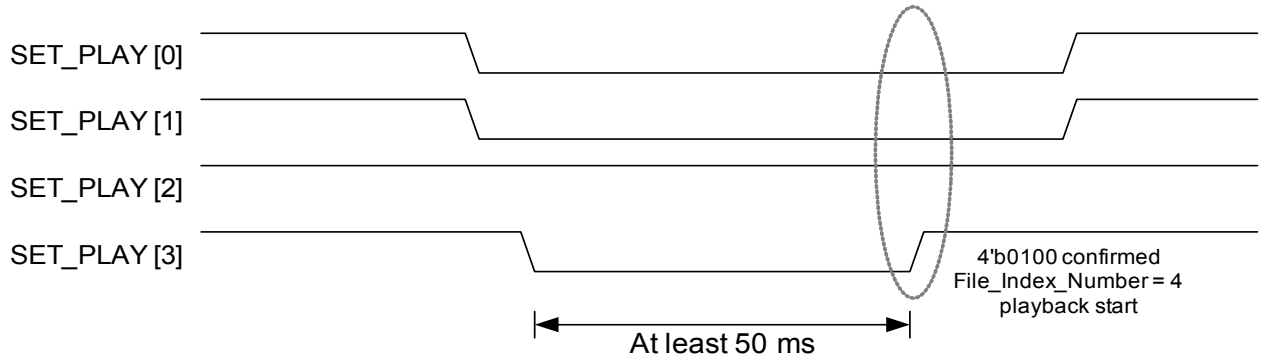


Figure 3.5 Standalone 1 File_Index_Number 4 playback

Table 3.6 Standalone 1 File_Index_Number setting

File_Index_Number	SET_PLAY3	SET_PLAY2	SET_PLAY1	SET_PLAY0
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
.				
.				
14	1	1	1	0
Initial state	1	1	1	1

3.5 Standalone 2 interface mode

When using the Standalone 2 interface mode, ISC messages are not required. This mode allows playback by controlling the SIS, SCKS, SOS, and MSG_RECEIVE pins shown in Table 3.7. The initial state for these four pins is logical value 1. Set the File_Index_Number to be played back in hexadecimal format, and set START to Low to start playback of the memory storage command.

As shown in Figure 3.6, entering Low for at least 50 ms for the SET_PLAY[1] pin and entering Low for at least 50 ms twice for the SET_PLAY[2] pin sets the hexadecimal value 21 in the internal register. Playback is started by subsequently entering Low for at least 50 ms for the SET_PLAY[3] pin. The SET_PLAY[0] pin functions as STOP, and should be used to stop playback.

Note the following points in Standalone 2 mode:

- File_Index_Number count, STOP, and START are recognized by Low input for 50 ms or more, followed by a rising edge.
- File_Index_Number can be specified in the range 0 to 254.
- Sound that can be played back consists of memory storage commands only.
- Setting File_Index_Number during playback and then using START will produce mixing playback. Note that overwrite playback will result if the same type of memory storage command is specified. (Memory storage melody command and memory storage buzzer/tone command are considered the same).
- Initialization information cannot be changed.

Table 3.7 Pins supporting Standalone 2

Interface mode	SIS	SCKS	SOS	MSG_RECIEVE	Outline
Standalone 2	SET_PLAY[0] STOP (input)	SET_PLAY[1] 0x01 (hex) (input)	SET_PLAY[2] 0x10 (hex) (input)	SET_PLAY[3] START (input)	Hexadecimal setting

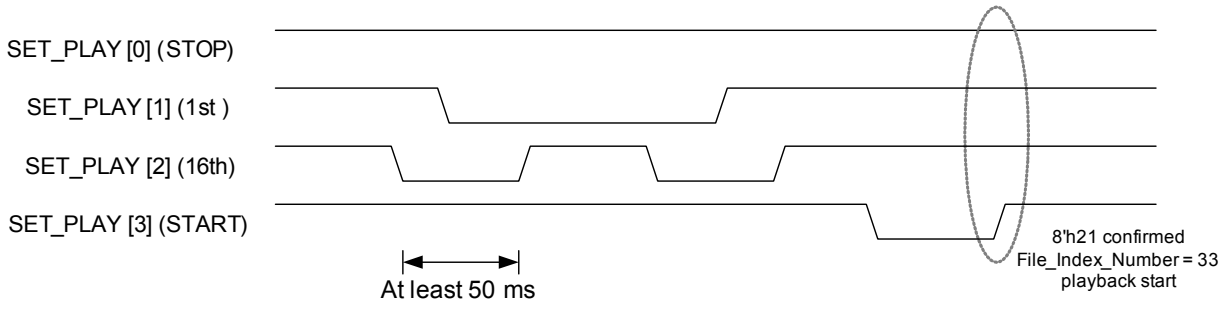


Figure 3.6 Standalone 2 File_Index_Number 33 playback

3.6 External serial flash memory

The S1V30080 can access the external serial flash memory by setting the FLASH_EN pin to High (QFP12-48 only). The following points must be noted when using flash memory:

- Flash memory capacity of up to 512 KB is supported.
- When writing to flash memory, set the FLASH_EN pin to Low. This causes the S1V30080 FLASH related pins to change to Hi-Z.
- Audio multi playback is possible for up to two channels.

  When using ISC_SEQUENCER_MULTI_CONFIG_REQ, only ch.0 and ch.1 are used. 0xFF (channel off) cannot be set to both ch.0 and ch.1 in this case.
- Audio data for storage in flash memory is limited to 10 bits.

  0x01 must be set when setting DAC_bit_width using ISC_AUDIO_CONFIG_REQ due to the 10 bit limit.

For details about connecting serial flash memory to the S1V30080, refer to the *S1V30080 Series Hardware Specifications*.

4. Message Protocol

4.1 ISC message basic specifications

The ISC message basic specifications are as follows.

- The S1V30080 is operated by receiving messages from the host device.
- Messages are made up of fixed-length header portions and variable-length data portions. The length is specified in bytes.
- There are two kinds of message, as follows.
  1. Request (REQuest)

     ...Request messages to the S1V30080 from the host device
  2. Response (RESPonse)

     ...Response messages in which the host device reads out information from the S1V30080

The MSG_RECEIVE pin indicates correct receipt of an ISC message. This pin outputs a pulse (Low → High → Low) after the message (*_REQ) sent from the host device has been received correctly. (See Figure 4.1.)

When using ISC_SYNTHESIZER_MELODY_START_REQ, the pulse Low → High → Low from the MSG_RECEIVE pin must always be confirmed before sending the next ISC_SYNTHESIZER_MELODY_START_REQ.

The pulse Low → High → Low is output on completion of transmission of ISC_STATUS_RESP from the S1V30080 only when the message sent from the host device was ISC_STATUS_REQ. (See Figure 4.2.)

The SOUND_PLAYING pin indicates that playback is in progress. This pin switches to High when the S1V30080 is playing back audio or synthesizer sound. (See Figure 4.3.)

The S1V30080 operates only when the SYSTEM_EN pin is High.

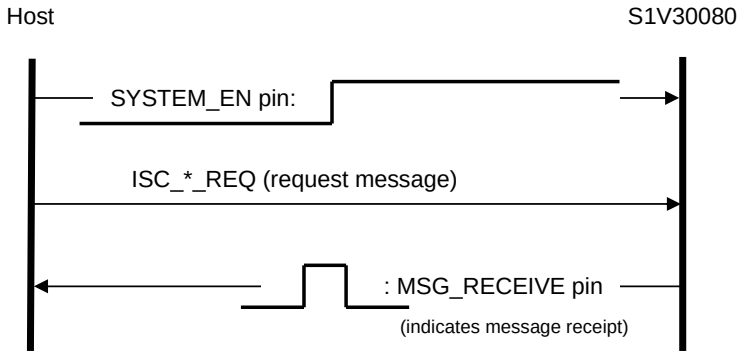


Figure 4.1 General message flow

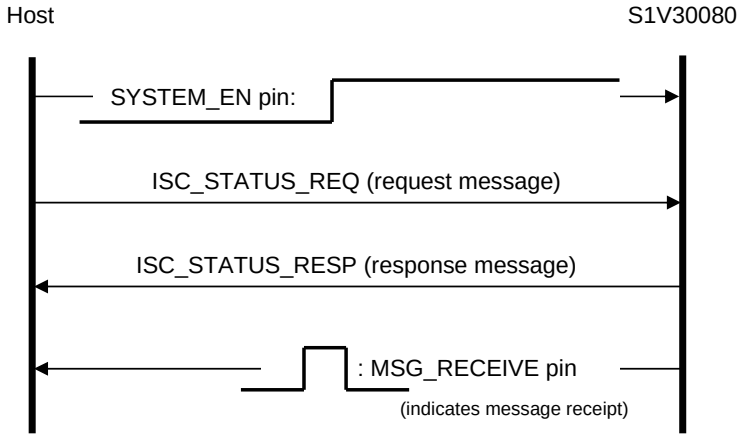


Figure 4.2 Status message flow

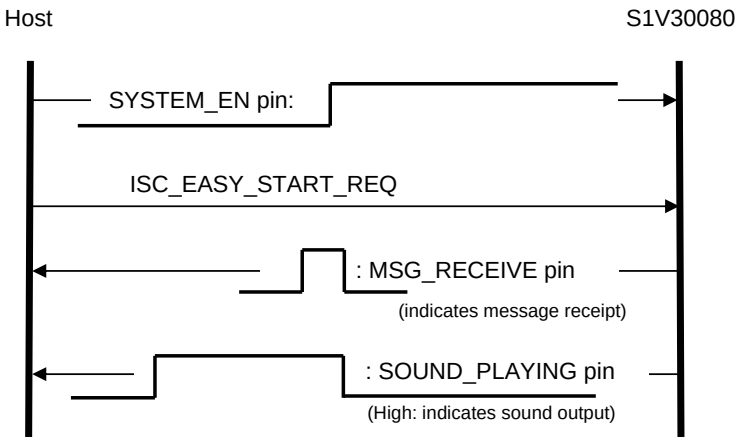


Figure 4.3 Audio playback message flow

4.2 ISC message basic structure

ISC messages are made up of fixed-length header portions and variable-length data portions. The ISC message structure is shown in Table 4.1. The S1V30080 can receive messages of up to 2,058 bytes. All ISC messages must comply with this length restriction. Note that correct operation cannot be guaranteed for messages exceeding the length restriction. Similarly, messages with incorrect length values cannot be guaranteed, and these may lead to problems such as inability to play back or subsequent communication problems.

The host device must send 0x00AA to notify the S1V30080 of the message start when starting message transmission. This 0x00AA is called the start message.

The S1V30080 interprets the data after the message start command 0x00AA as a request message.

Table 4.1 Message structure

	Byte	Description
Start message	-2, -1	0x00AA
Header portion	0, 1	length: Message length (number of bytes including header)
	2, 3	msg_id: Message ID (Specifies the data included in the payload.)
Data portion	4 ... n	Variable-length data payload

4.3 ISC message basic flow

4.3.1 Updating initialization information

If initialization information needs to be updated (refer to Section 3.1.1), this should be done before sending an audio playback start message (ISC_EASY_START_REQ or ISC_SOUND_START_REQ). The updated initialization information remains enabled until the SYSTEM_EN pin is set to Low.

Note that setting the SYSTEM_EN pin to Low resets with a HW reset, thus disabling the updated initialization information.

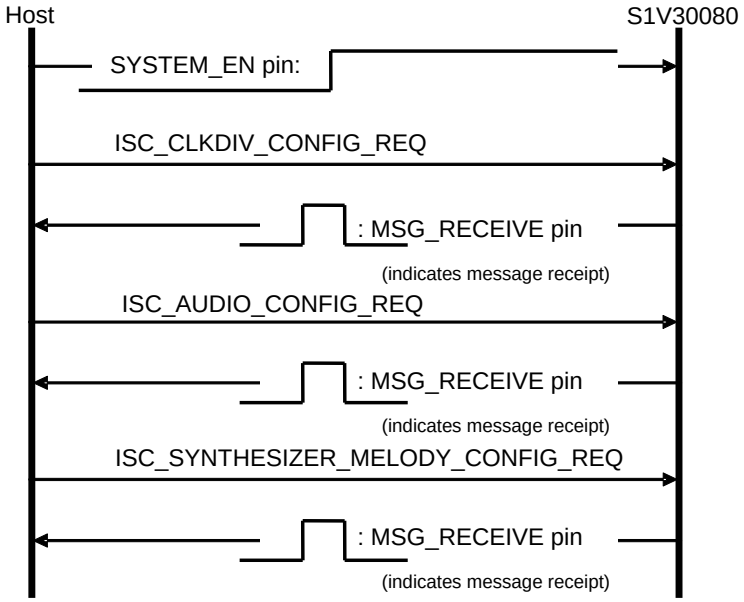


Figure 4.4 Initialization information update flow

4.3.2 One-command playback

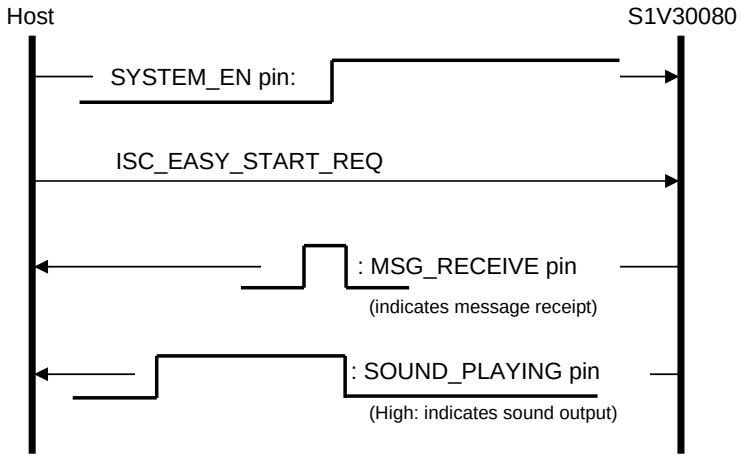


Figure 4.5 One-command playback flow

4.3.3 Audio playback

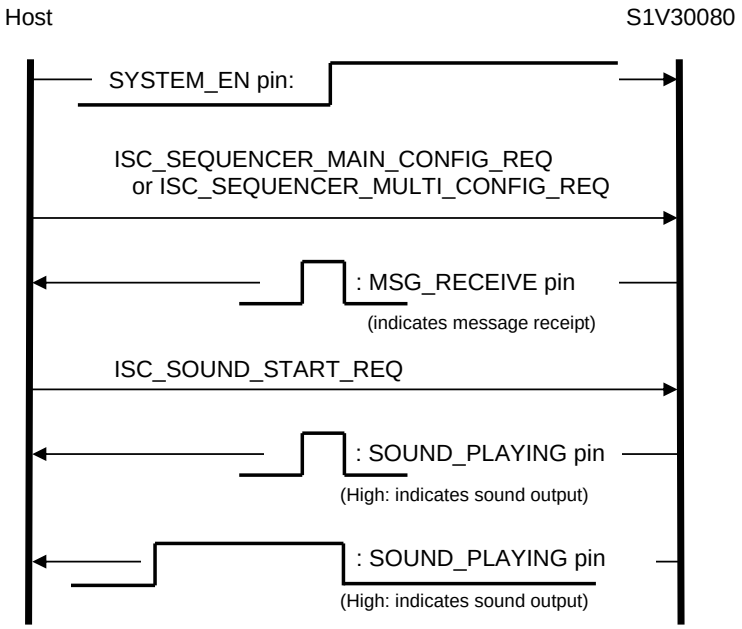


Figure 4.6 Audio playback flow

4.3.4 Synthesizer playback

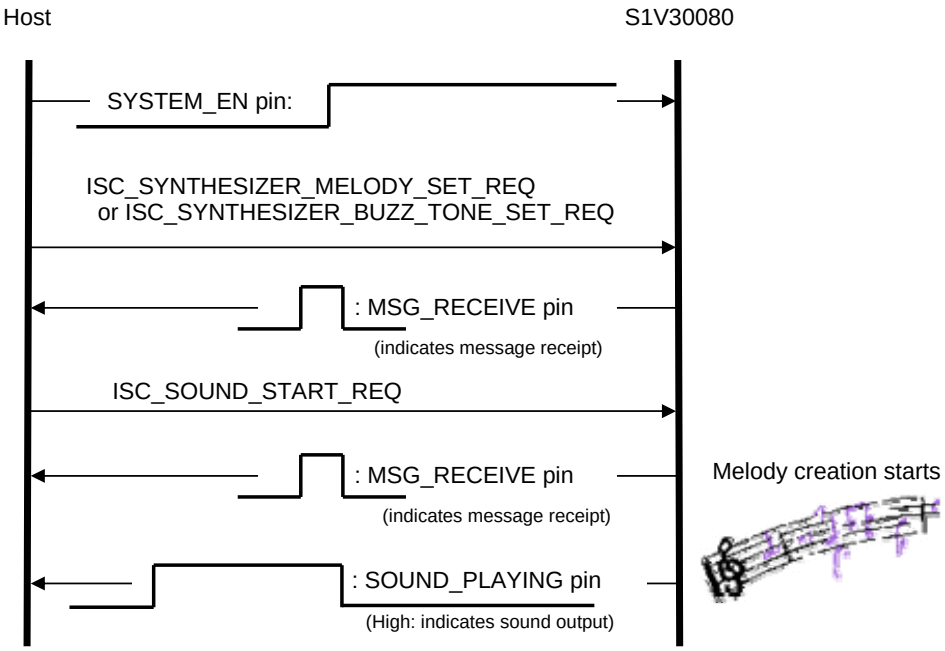


Figure 4.7 Synthesizer playback flow

4.3.5 Synthesizer melody playback (streaming playback)

When score information is sent from the host device using an ISC_SYNTHESIZER_MELODY_START_REQ message, melody playback starts automatically, allowing extended playback. Figure 4.8 illustrates the playback flow when using ISC_SYNTHESIZER_MELODY_START_REQ.

When using this message, note that the MSG_RECEIVE pin must output the pulse Low → High → Low before the next ISC_SYNTHESIZER_MELODY_START_REQ can be sent by the host.

Note also that for synthesizer melody playback and mixing using this message, commands other than MSG_RECEIVE for this message will be masked as shown in Figure 4.9. Mixing playback is possible by sending ISC_SOUND_START_REQ at the appropriate time.

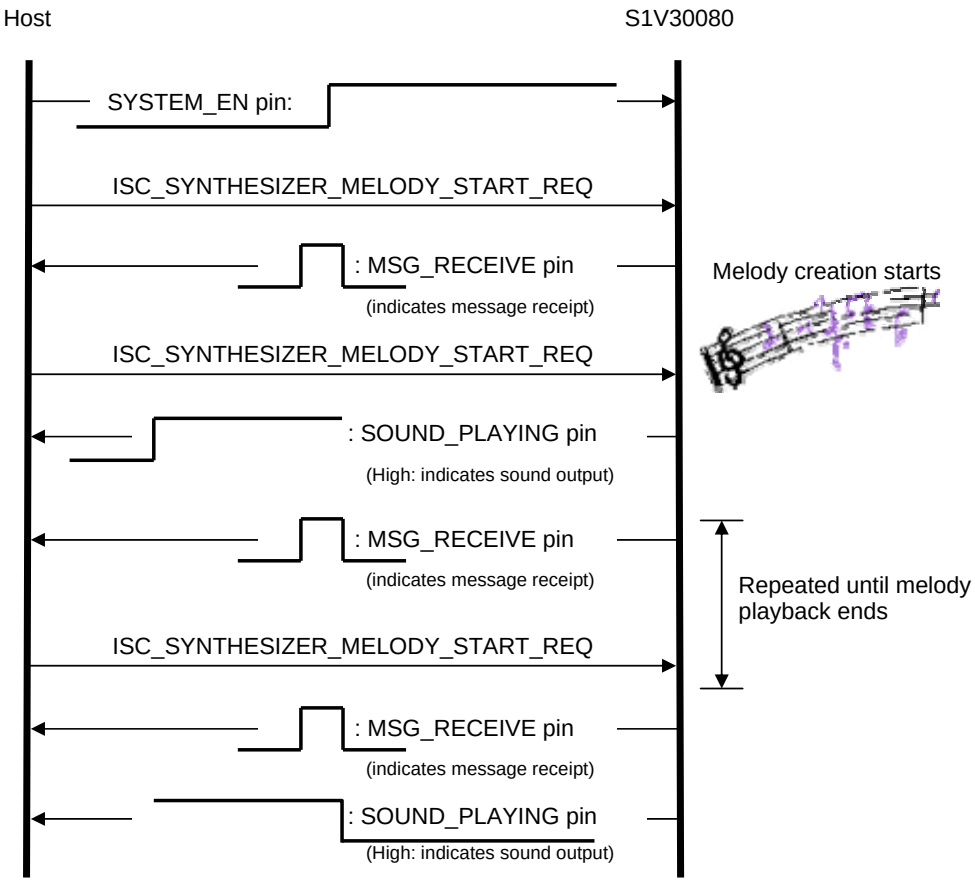


Figure 4.8 Synthesizer melody playback flow

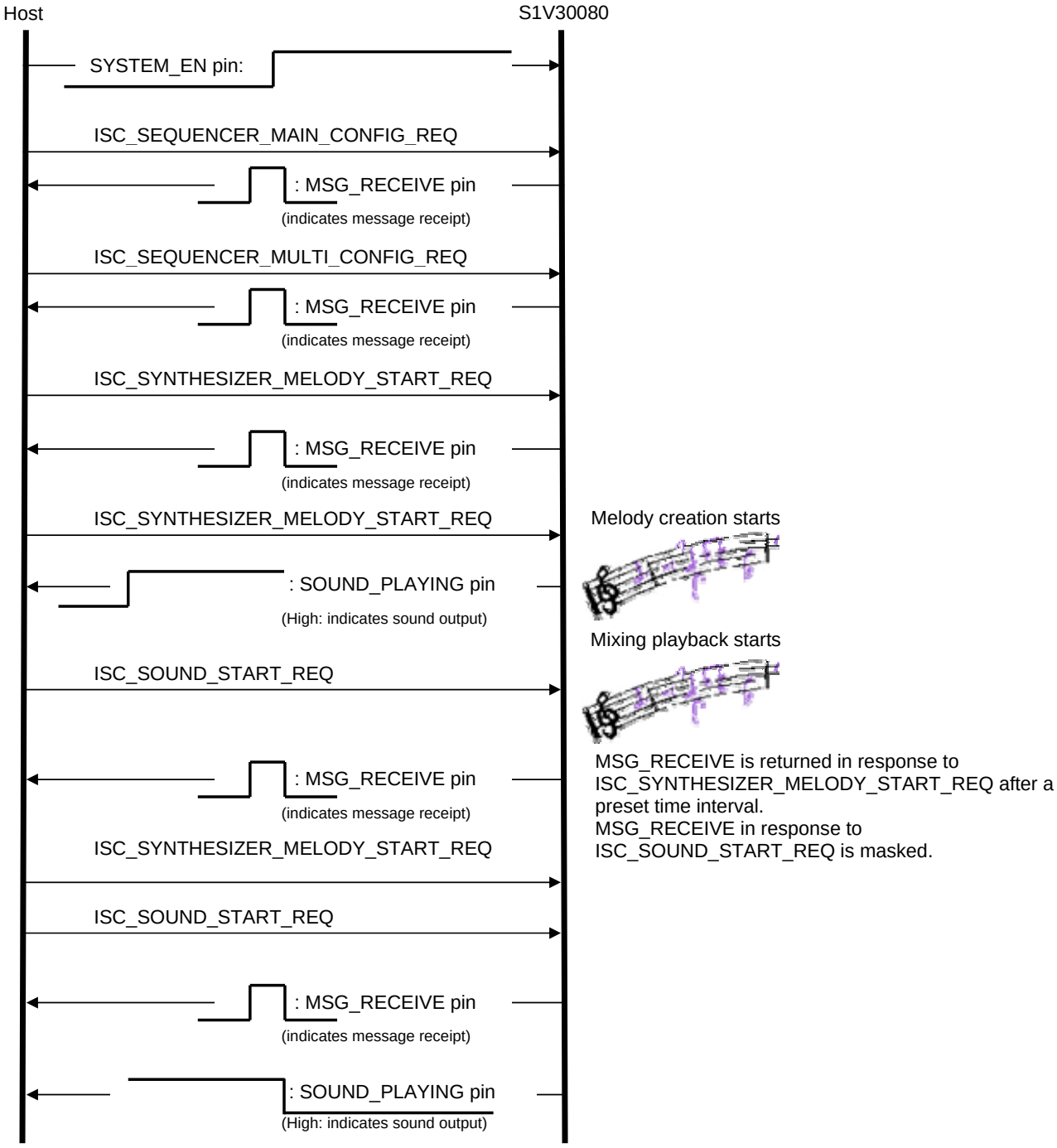


Figure 4.9 Synthesizer melody playback flow (mixing playback)

4.3.6 Mixing playback

Mixing playback is possible using ISC_SOUND_START_REQ.

Mixing playback can use up to three playback types: audio main playback, audio multi playback, and synthesizer playback. Note that mixing is not possible using the same type of playback.

Mixing playback timing can be adjusted by sending ISC_SOUND_START_REQ for each playback.

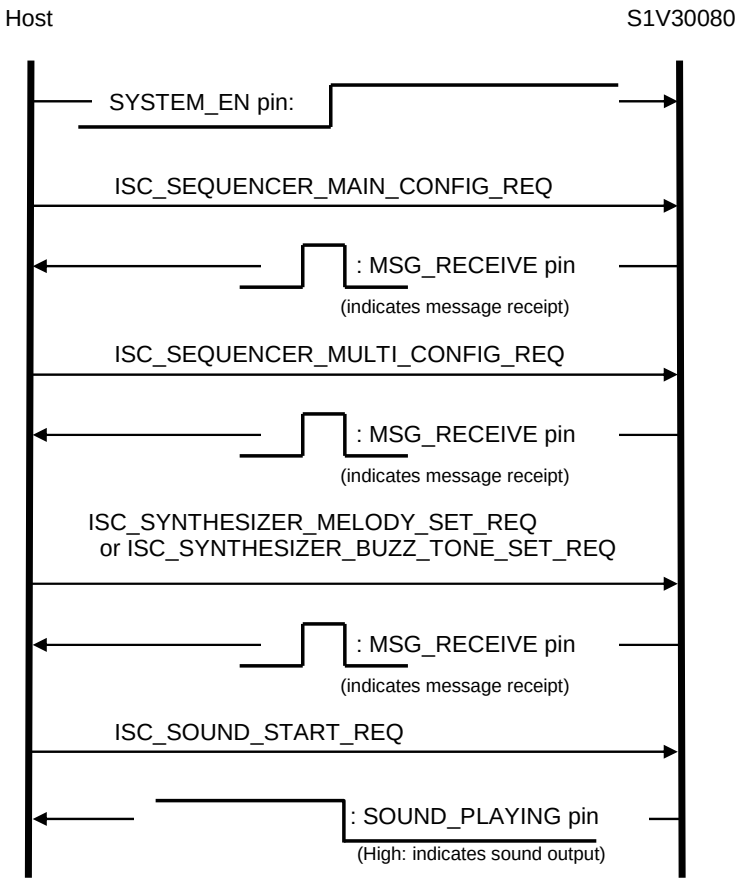


Figure 4.10 Mixing playback flow

4.3.7 Mixing playback (using one-command playback)

Mixing playback is also possible using ISC_EASY_START_REQ.

Mixing playback can use up to three playback types: audio main playback, audio multi playback, and synthesizer playback. Note that mixing is not possible using the same type of playback.

Errors of up to 2 seconds may occur in the mixing playback timing when using ISC_EASY_START_REQ. If mixing timing is particularly important, mixing playback is recommended using ISC_SOUND_START_REQ as described in Section 4.3.6, “Mixing playback.”

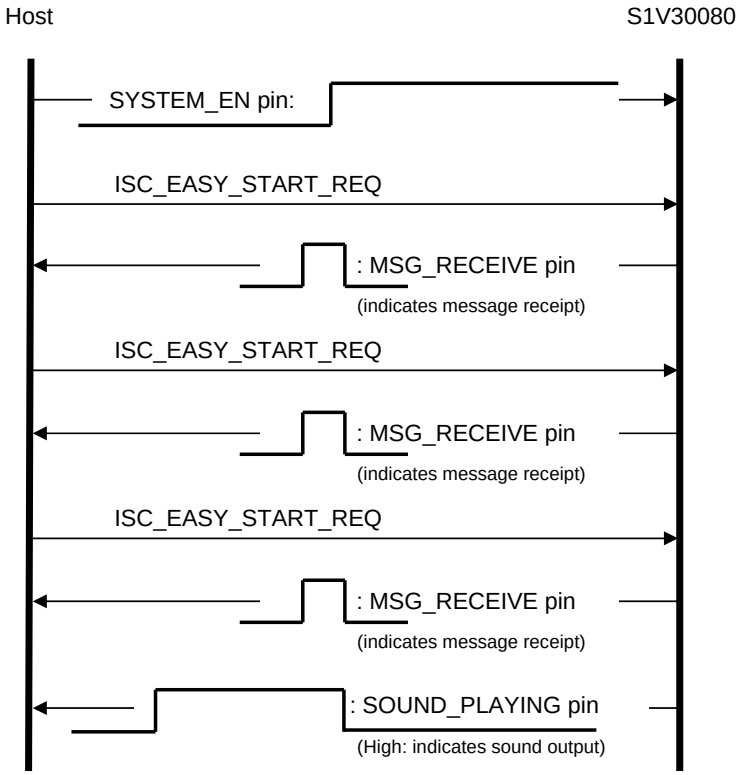


Figure 4.11 One-command mixing playback flow

4.4 Checksum

The S1V30080 features a checksum function for detecting transfer data errors arising from noise in communication lines between the host and itself when using the I2C or SPI interface. This function can be enabled or disabled using ISC_CHKSUM_CONFIG_REQ. (It is not necessary to add checksum data in ISC_CHKSUM_CONFIG_REQ messages even if the checksum function is enabled.)

When the function is enabled, the host adds checksum data expressing the sum of the data sent as one byte at the end of the message sent. (The message Length field must not include the 1 byte of checksum data.) If the sum of the data sent is two or more bytes, add only the last byte. Discard any additional bytes.

Checksum data should be added as described above (refer to Figure 4.12) when using the I2C interface.

When using the SPI interface, add the checksum data as described above, and add 0xFF to the last byte (refer to Figure 4.13) before transmission.

The S1V30080 sums data sent from the host for each byte and detects data errors by comparing this against the final checksum data.

For details of how to detect checksum errors, refer to Section 5.7.1, "ISC_STATUS_REQ."

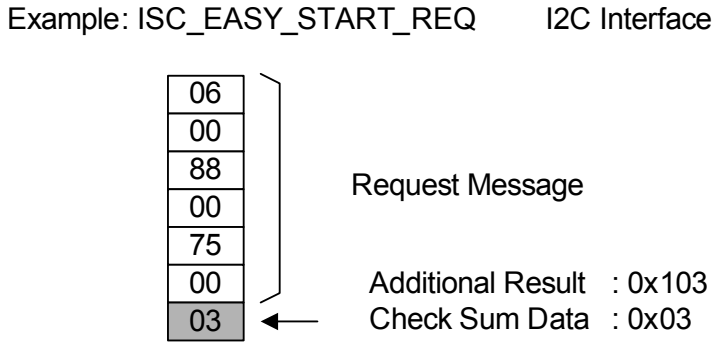


Figure 4.12 Adding checksum data (I2C interface)

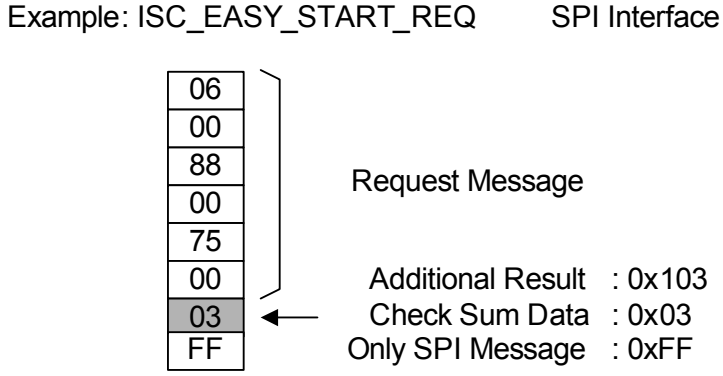


Figure 4.13 Adding checksum data (SPI interface)

5. Message Details

5.1 Message identifier list

Table 5.1 lists the message identifiers supported by the S1V30080.

The individual message details are described later.

Table 5.1 Message identifier list

Message	Purpose	Length	ID	Reference
System control message				
ISC_CLKDIV_CONFIG_REQ	Clock division ratio setting	0x0006	0x0014	Table 5.2
ISC_CHKSUM_CONFIG_REQ	Checksum function setting	0x0006	0x0016	Table 5.4
ISC_AUDIO_CONFIG_REQ	Sampling frequency, volume setting	0x000E	0x0018	Table 5.5
ISC_SEQUENCER_MAIN_CONFIG_REQ	Audio main playback setting	variable	0x0020	Table 5.6
ISC_SEQUENCER_MULTI _CONFIG_REQ	Audio multi playback setting	variable	0x0024	Table 5.7
ISC_ SYNTHESIZER_MELODY_CONFIG_REQ	Synthesizer melody timbre setting	0x000A	0x0060	Table 5.8
ISC_ SYNTHESIZER_MELODY_START_REQ	Synthesizer melody playback start	variable	0x0064	Table 5.9
ISC_ SYNTHESIZER_MELODY_SET_REQ	Synthesizer melody playback setting	variable	0x0068	Table 5.10
ISC_ SYNTHESIZER_BUZZ_TONE_SET_REQ	Synthesizer buzzer/tone playback setting	variable	0x006C	Table 5.11
ISC_EASY_START_REQ	One-command playback start	0x0006	0x0088	Table 5.12
ISC_SOUND_START_REQ	Sound playback start	0x0008	0x0080	Table 5.13
ISC_SOUND_STOP_REQ	Sound playback stop	0x0008	0x0084	Table 5.14
ISC_AUDIO_VOLUME_REQ	Volume control	0x000C	0x0090	Table 5.15
ISC_AUDIO_MUTE_REQ	Audio output mute	0x0006	0x0094	Table 5.16
ISC_STATUS_REQ	Internal status confirmation request	0x0008	0x00E0	Table 5.17
ISC_STATUS_RESP	Internal status response	0x000C	0x00E1	Table 5.18

5.2 System control messages

5.2.1 ISC_CLKDIV_CONFIG_REQ

Table 5.2 ISC_CLKDIV_CONFIG_REQ

Direction	Host device to S1V30080	
Purpose	Used to set input clock division	
Byte	**Field**	**Value**
0	Length (LSB)	0x0006
1	Length (MSB)	
2	msg_id (LSB)	0x0014: ISC_CLKDIV_CONFIG_REQ
3	msg_id (MSB)	
4	clock_divide	0x00: No division 0x01: No division 0x02: 2 divisions 0x04: 4 divisions 0x08: 8 divisions *Note 1
5	reserved	Set to 0x00

Note 1: 0x00 (no division) will be set if an unsupported value (i.e. any value other than 0x00, 0x01, 0x02, 0x04, 0x08) is sent.

Note 2: This message should be sent with a baud rate of at least 20 times the internal clock.

Note 3: The default clock division ratio will be the value set by the "S1V30080 Series Sound Tool."

The clock division ratio is determined by the input clock frequency to the S1V30080, the audio data sampling frequency, and the DAC_bit_width, and is calculated using the equation below:

$$\textit{Input clock division ratio} = \textit{Input clock frequency (kHz)} / \{\textit{Sampling frequency (kHz)} \times 2^{(\textit{DAC bit width})}\}$$

Table 5.3 shows an example.

Table 5.3 Clock division ratio setting example

Input clock division ratio	Input clock frequency (kHz)	Sampling frequency (kHz)	DAC_bit_width	Internal system clock (MCLK) frequency (kHz)
4	4096	4	8	1024
1	8192	8	10	8192
1	16384	16	10	16384

5.2.2 ISC_CHKSUM_CONFIG_REQ

Table 5.4 ISC_CHKSUM_CONFIG_REQ

Direction	Host device to S1V30080	
Purpose	Used to enable and disable checksum function	
Byte	**Field**	**Value**
0	Length (LSB)	0x0006
1	Length (MSB)	
2	msg_id (LSB)	0x0016: ISC_CHKSUM_CONFIG_REQ
3	msg_id (MSB)	
4	chksum_set	0x00: Checksum function off 0x01: Checksum function on *Note 1
5	reserved	Set to 0x00

Note 1: 0x00 (checksum off) will be set if an unsupported value (i.e. any value other than 0x00 or 0x01) is sent.

Note 2: The default checksum setting is 0x00 (checksum off).

5.3 Sound output setting messages

5.3.1 ISC_AUDIO_CONFIG_REQ

Table 5.5 ISC_AUDIO_CONFIG_REQ

Direction	Host device to S1V30080	
Purpose	Used to set sound output	
Byte	**Field**	**Value**
0	Length (LSB)	0x000E
1	Length (MSB)	
2	msg_id (LSB)	0x0018: ISC_AUDIO_CONFIG_REQ
3	msg_id (MSB)	
4	audio_sample_rate	0x04 ... 4 [kHz] 0x08 ... 8 [kHz] 0x0C ... 12 [kHz] 0x10 ... 16 [kHz] *Note 1
5	audio_gain_sequencer_main	0x00 ... 0 dB 0x01 ... -0.5 dB 0x02 ... -1.0 dB 0x03 ... -1.5 dB .... 0x7D ... -62.5 dB 0x7E ... -63.0 dB 0x7F ... Mute *Note2
6	audio_gain_sequencer_multi	
7	audio_gain_synth0	
8	audio_gain_synth1	
9	audio_gain_synth2	
A	audio_gain_synth3	
B	audio_gain_synth4	
C	DAC_bit_width	0x00 ... 8 bit 0x01 ... 10 bit *Note3
D	reserved	Set to 0x00

Note 1: 0x10 (16 kHz) will be set if an unsupported value (i.e. any value other than 0x04, 0x08, 0x0C, or 0x10) is sent.

Note 2: 0x7F (Mute) will be set if a value of 0x7F or greater is sent.

Note 3: 0x01 (10 bit) will be set if an unsupported value (i.e. any value other than 0x00 or 0x01) is sent.
DAC_bit_width should be the same value as the number of quantization bits for the data stored in the internal ROM or external flash memory.
Select 0x01 when using external serial flash memory, as this is limited to 10 bits.
If using external serial flash memory, refer to Section 3.6, “External serial flash memory.”

Note 4: The default setting will be the value created using the “S1V30080 Series Sound Tool.”

5.4 Audio playback setting messages

5.4.1 ISC_SEQUENCER_MAIN_CONFIG_REQ

Table 5.6 ISC_SEQUENCER_MAIN_CONFIG_REQ

Direction	Host device to S1V30080	
Purpose	Used to set sound audio main playback	
Byte	**Field**	**Value**
0	length (LSB)	VARIABLE (max 0x0108)
1	length (MSB)	
2	msg_id (LSB)	0x0020 : ISC_SEQUENCER_MAIN_CONFIG_REQ
3	msg_id (MSB)	
4	play_count (LSB)	0xFFFF: Unlimited 0x0001 – 0xFFFE: Number of playback times
5	play_count (MSB)	
6	num_files	0x01-0x7F: Number of file_event structures *Note 1
7	reserved	Set to 0x00
8, 9	file_event[0]	file_event structure 0
A, B	file_event[1]	file_event structure 1
...	...	...
104, 105	file_event[126]	file_event structure 126
length - 2	reserved	Set to 0x00
length - 1	reserved	Set to 0x00

file_event structure (ISC_SEQUENCER_MAIN_CONFIG_REQ)

0	delay_ms	Length of mute interval inserted before playback 0x00: No mute interval (gapless playback) 0x01: 10 ms 0x02: 20 ms ... 0x64: 1,000 ms *Note 2
1	file_index_number	Index number of audio data stored in internal ROM or external flash memory *Note 3

Note 1: Not guaranteed if an unsupported value (i.e. any value other than 0x01 to 0x7F) is sent.

Note 2: 0x64 (1,000 ms) will be set if an unsupported value (i.e. 0x64 or more) is sent.

Note 3: Only audio data can be specified; memory storage commands cannot be specified. 0xFF cannot be specified either. For details of the association between audio data and File_Index_Number, refer to "ROMImage_yymmdd_hhmmss.csv" output from the "S1V30080 Series Sound Tool."

Note 4: This message can be sent divided into multiple parts if necessary by the host device. Note that the file_event field bytes should be set to 4, 8, 16, 32, 64, 128, 256, 512, 1024, or 2048 for divided transmission. This does not apply to the final transmission. The individual divided message lengths can be entered in the length field for divided transmission, but the total value should be entered for num_files.

5.4.2 ISC_SEQUENCER_MULTI_CONFIG_REQ

Table 5.7 ISC_SEQUENCER_MULTI_CONFIG_REQ

Direction	Host device to S1V30080	
Purpose	Used to set sound audio multi playback	
Byte	**Field**	**Value**
0	length (LSB)	VARIABLE (max 0x0304)
1	length (MSB)	
2	msg_id (LSB)	0x0024 : ISC_SEQUENCER_MULTI_CONFIG_REQ
3	msg_id (MSB)	
4	play_count (LSB)	0xFFFF: Unlimited 0x0001 – 0xFFFE: Number of playback times
5	play_count (MSB)	
6	num_files (LSB)	0x01–0x7F: Number of file_event structures *Note 1
7	num_files (MSB)	Set to 0x00
8 - D	file_event[0]	file_event structure 0
E - 13	file_event[1]	file_event structure 1
...	...	...
2FC - 301	file_event[126]	file_event structure 126
length - 2	reserved	Set to 0x00
length - 1	reserved	Set to 0x00

file_event structure (ISC_SEQUENCER_MULTI_CONFIG_REQ)

0	delay_ms	Length of mute interval inserted before playback 0x00: No mute interval (gapless playback) 0x01: 10 ms 0x02: 20 ms ... 0x64: 1,000 ms *Note 2
1	file_index_number_ch0	Index number of audio data stored in internal ROM or external flash memory (Specify 0xFF for channels not used.) *Note 3
2	file_index_number_ch1	
3	file_index_number_ch2	
4	file_index_number_ch3	
5	file_index_number_ch4	

Note 1: Not guaranteed if an unsupported value (i.e. any value other than 0x01 to 0x7F) is sent.

Note 2: 0x64 (1,000 ms) will be set if an unsupported value (i.e. 0x64 or more) is sent.

Note 3: Only audio data can be specified; memory storage commands cannot be specified. Specifying 0xFF disables that channel alone. Note that 0xFF cannot be specified for all channels. If using external serial flash memory, channels 2, 3, and 4 cannot be used, for reasons related to transfer rates. For details of the association between audio data and File_Index_Number, refer to "ROMImage_yymmdd_hhmmss.csv" output from the "S1V30080 Series Sound Tool."
If using external serial flash memory, refer to Section 3.6, "External serial flash memory."

Note 4: This message can be sent divided into multiple parts if necessary by the host device. Note that the file_event field bytes should be set to 4, 8, 16, 32, 64, 128, 256, 512, 1024, or 2048 for divided transmission. This does not apply to the final transmission. The individual divided message lengths can be entered in the length field for divided transmission, but the total value should be entered for num_files.

5.5 Synthesizer playback setting messages

5.5.1 ISC_SYNTHESIZER_MELODY_CONFIG_REQ

Table 5.8 ISC_SYNTHESIZER_MELODY_CONFIG_REQ

Direction	Host device to S1V30080		
Purpose	Used to set synthesizer playback tempo and envelope		
Byte	**Field**	**Value**	
0	length (LSB)	0x000A	
1	length (MSB)		
2	msg_id (LSB)	0x0060 : ISC_SYNTHESIZER_MELODY_CONFIG_REQ	
3	msg_id (MSB)		
4	tempo	0x00 ... Prohibited 0x01 ... ♩=10/min 0x02 ... ♩=20/min 0x03 ... ♩=30/min 0x04 ... ♩=40/min (Largo) 0x05 ... ♩=50/min 0x06 ... ♩=60/min (Larghetto) 0x07 ... ♩=70/min 0x08 ... ♩=80/min 0x09 ... ♩=90/min 0x0A ... ♩=100/min	0x0B ... ♩=110/min 0x0C ... ♩=120/min (Allegro) 0x0D ... ♩=130/min 0x0E ... ♩=140/min 0x0F ... ♩=150/min 0x10 ... ♩=160/min 0x11 ... ♩=170/min 0x12 ... ♩=180/min 0x13 ... ♩=190/min 0x14 ... ♩=200/min (Prestissimo) *Note 1
5	attack_level	0x00: Prohibited ..... 0x3E: Prohibited 0x3F: Mute → -32.0dB (Level 63) 0x40: Mute → -31.5dB (Level 64) ..... 0x7F: Mute → 0.0dB (Level 127)	
6	attack_step	0x00: Prohibited 0x01: 1PCM sample step 0x02: 2PCM sample step ..... 0x7F: 127PCM step	
7	decay_level	0x00: Prohibited 0x01: attack_level – 0.5db (Level 1) 0x02: attack_level – 1.0db (Level 2) ..... 0x7F: attack_level – 63.5db (Level 127)	
8	decay_step	Same as attack_step	
9	release_step	Same as attack_step	

Note 1: 0x06 (♩ = 60/min) will be set if a non-supported value is specified. (♩ is a crotchet.)

Note 2: The default settings will be the values created by the "S1V30080 Series Sound Tool."

Note 3: It is recommended that the message output from the "S1V30080 Series Sound Tool" be used for this message. For details of tempo and envelope settings, see below.

The envelope refers to the line enclosing the sound wave, and is made up of Attack, Decay, Sustain, and Release parameters (see Figure 5.1). The sound timbre can be altered by changing these parameters. The values shown below can be set for the S1V30080.

tempo : Number of crotchets per minute. For example, for ♩ = 60/min, a semiquaver will be 250 ms long.

attack_level : Attack level.

attack_step : The number of PCM samples necessary for a 0.5 dB increase in the Attack period.

decay_level : Attenuation level from the attack level.

decay_step : The number of PCM samples necessary for a 0.5 dB decrease in the Decay period.

release_step : The number of PCM samples necessary for a 0.5 dB decrease in the Release period.

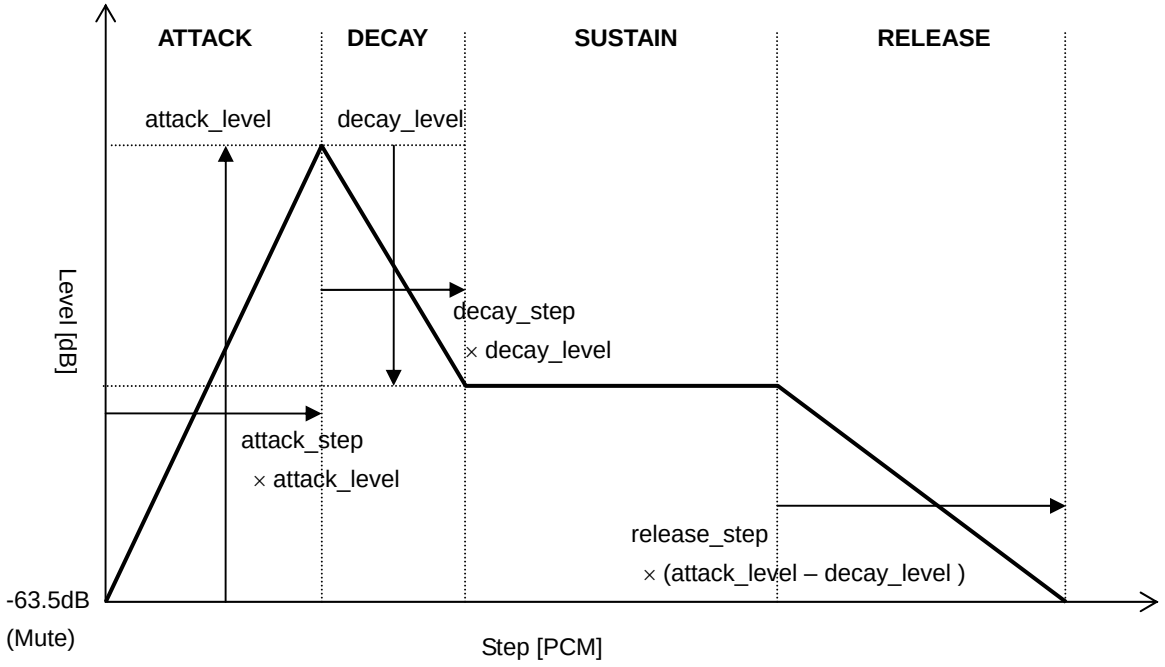


Figure 5.1 Envelope settings

The envelope is applied for each semiquaver. (If the file_event structure continuous bit is set to 1, the envelope is applied with the Sustain period extended for the note set.) Therefore, if envelope values are set exceeding the number of semiquaver PCM samples, the will become saturated, preventing playback as desired. Values should be set which satisfy the equation below.

fs [Hz] × (60/tempo) × (1/4)

> (attack_step × attack_level)

+ (decay_step × decay_level)

+ (release_step × (attack_level – decay_level))

The volume output by the S1V30080 is the sum of the envelope value set by ISC_SYNTHESIZER_MELODY_CONFIG_REQ and the values (dB) set by ISC_AUDIO_CONFIG_REQ or ISC_AUDIO_VOLUME_REQ.

5.5.2 ISC_SYNTHESIZER_MELODY_START_REQ

Table 5.9 ISC_SYNTHESIZER_MELODY_START_REQ

Direction	Host device to S1V30080	
Purpose	Used for synthesizer melody playback (streaming)	
Byte	**Field**	**Value**
0	length (LSB)	VARIABLE (max 0x2808)
1	length (MSB)	
2	msg_id (LSB)	0x0064 : ISC_SYNTHESIZER_MELODY_START_REQ
3	msg_id (MSB)	
4	num_files (LSB)	Number of file_event structures (0x019A – 0x0800) *Note 1
5	num_files(MSB)	
6	reserved	Set to 0x00
7	reserved	Set to 0x00
8 ... C	file_event[0]	file_event structure 0
D ... 11	file_event[1]	file_event structure 1
...	...	...
...	...	...

file_event structure (ISC_SYNTHESIZER_MELODY_START_REQ)

0	Ch0_sound	Bit[7] 1: Sound ON 0: Sound OFF *Note 2 Bit[6] 1: Continuous 0: Independent	Bit[5:0] 0x00: Do 2 0x01: Do #2 0x02: Re 2 0x03: Re #2 0x04: Mi 2 0x05: Fa 2 0x06: Fa #2 0x07: Sol 2 0x08: Sol #2 0x09: La 2 0x0A: La #2 0x0B: Ti 2	0x0C: Do 3 0x0D: Do #3 0x0E: Re 3 0x0F: Re #3 0x10: Mi 3 0x11: Fa 3 0x12: Fa #3 0x13: Sol 3 0x14: Sol #3 0x15: La 3 0x16: La #3 0x17: Ti 3	0x18: Do 4 0x19: Do #4 0x1A: Re 4 0x1B: Re #4 0x1C: Mi 4 0x1D: Fa 4 0x1E: Fa #4 0x1F: Sol 4 0x20: Sol #4 0x21: La 4 0x22: La #4 0x23: Ti 4	0x24: Do 5 0x25: Do #5 0x26: Re 5 0x27: Re #5 0x28: Mi 5 0x29: Fa 5 0x2A: Fa #5 0x2B: Sol 5 0x2C: Sol #5 0x2D: La 5 0x2E: La #5 0x2F: Ti 5	0x30: Do 6 0x31: Do #6 0x32: Re 6 0x33: Re #6 0x34: Mi 6 0x35: Fa 6 0x36: Fa #6 0x37: Sol 6 0x38: Sol #6 0x39: La 6 0x3A: La #6 0x3B: Ti 6 0x3C: reserved 0x3D: reserved 0x3E: reserved 0x3F: reserved
1	Ch1_sound						
2	Ch2_sound						
3	Ch3_sound						
4	Ch4_sound						

Note 1: If the num_files value is 0x199 or less, use ISC_SYNTHESIZER_MELODY_SET_REQ. This message can be used only when the num_files value is 0x19A or more and 0x800 or less.

Note 2: Set 0x00 for Sound OFF.

Note 3: This message must be sent divided into multiple parts. The file_event field bytes should be set to 4, 8, 16, 32, 64, 128, 256, 512, 1024, or 2048 for divided transmission. This does not apply to the final transmission. The individual divided message lengths can be entered in the length field for divided transmission, but the total value should be entered for num_files.

Note 4: It is recommended that the message output from the “S1V30080 Series Sound Tool” be used for this message.

5.5.3 ISC_SYNTHESIZER_MELODY_SET_REQ

Table 5.10 ISC_SYNTHESIZER_MELODY_SET_REQ

Direction	Host device to S1V30080	
Purpose	Used for setting synthesizer melody playback	
Byte	**Field**	**Value**
0	length (LSB)	VARIABLE (max 0x805)
1	length (MSB)	
2	msg_id (LSB)	0x0068 : ISC_SYNTHESIZER_MELODY_SET_REQ
3	msg_id (MSB)	
4	play_count(LSB)	0x001–0xFFFE: Number of playback times 0xFFFF: Unlimited
5	play_count(MSB)	
6	num_files (LSB)	Number of file_event structures (0x0199) *Note 1
7	num_files (MSB)	
8 ... C	file_event[0]	file_event structure 0
D ... 11	file_event[1]	file_event structure 1
...	...	...

file_event structure (ISC_SYNTHESIZER_MELODY_START_REQ)

0	Ch0_sound	Bit[7] 1: Sound ON 0: Sound OFF *Note 2 Bit[6] 1: Continuous 0: Independent	Bit[5:0] 0x00: Do 2 0x01: Do #2 0x02: Re 2 0x03: Re #2 0x04: Mi 2 0x05: Fa 2 0x06: Fa #2 0x07: Sol 2 0x08: Sol #2 0x09: La 2 0x0A: La #2 0x0B: Ti 2	0x0C: Do 3 0x0D: Do #3 0x0E: Re 3 0x0F: Re #3 0x10: Mi 3 0x11: Fa 3 0x12: Fa #3 0x13: Sol 3 0x14: Sol #3 0x15: La 3 0x16: La #3 0x17: Ti 3	0x18: Do 4 0x19: Do #4 0x1A: Re 4 0x1B: Re #4 0x1C: Mi 4 0x1D: Fa 4 0x1E: Fa #4 0x1F: Sol 4 0x20: Sol #4 0x21: La 4 0x22: La #4 0x23: Ti 4	0x24: Do 5 0x25: Do #5 0x26: Re 5 0x27: Re #5 0x28: Mi 5 0x29: Fa 5 0x2A: Fa #5 0x2B: Sol 5 0x2C: Sol #5 0x2D: La 5 0x2E: La #5 0x2F: Ti 5	0x30: Do 6 0x31: Do #6 0x32: Re 6 0x33: Re #6 0x34: Mi 6 0x35: Fa 6 0x36: Fa #6 0x37: Sol 6 0x38: Sol #6 0x39: La 6 0x3A: La #6 0x3B: Ti 6 0x3C:reserved 0x3D:reserved 0x3E:reserved 0x3F:reserved
1	Ch1_sound						
2	Ch2_sound						
3	Ch3_sound						
4	Ch4_sound						

Note 1: Playback may not be possible as desired if the num_files value is 0x199 or more.

Note 2: Set 0x00 for Sound OFF.

Note 3: This message must be sent divided into multiple parts. The file_event field bytes should be set to 4, 8, 16, 32, 64, 128, 256, 512, 1024, or 2048 for divided transmission. This does not apply to the final transmission. The individual divided message lengths can be entered in the length field for divided transmission, but the total value should be entered for num_files.

Note 4: It is recommended that the message output from the “S1V30080 Series Sound Tool” be used for this message.

5.5.4 ISC_SYNTHESIZER_BUZZ_TONE_SET_REQ

Table 5.11 ISC_SYNTHESIZER_BUZZ_TONE_SET_REQ

Direction	Host device to S1V30080	
Purpose	Used for setting synthesizer buzzer/tone playback	
Byte	**Field**	**Value**
0	length (LSB)	VARIABLE (max 0x800)
1	length (MSB)	
2	msg_id (LSB)	0x006C : ISC_SYNTHESIZER_BUZZ_TONE_SET_REQ
3	msg_id (MSB)	
4	play_count (LSB)	0x001–0xFFFE: Number of playback times 0xFFFF: Unlimited
5	play_count (MSB)	
6	num_files (LSB)	Number of file_event structures (max 0x00AA) *Note 1
7	num_files (MSB)	
8 ... C	file_event[0]	file_event structure 0
D ... 11	file_event[1]	file_event structure 1
...	...	...

file_event structure (ISC_SYNTHESIZER_BUZZ_TONE_SET_REQ)

0	delay_ms	Bit[7] ... 1: Continuous 0: Independent Bit[6:0]... Length of mute interval inserted before playback 0x00: No mute interval (gapless playback) 0x01: 10 ms 0x02: 20 ms ... 0x64: 1,000 ms (max) *Note 2
1	playtime_ms	0x00: reserved 0x01: reserved 0x02: 20 ms ... 0x64: 1,000 ms ... 0xFF: 2,550 ms (max)
2	Ch0_frequency (LSB)	0x0000: 0 Hz ... 0x01F4: 500 Hz ... 0x03E8: 1,000 Hz ... 0x05DC: 1,500 Hz ... 0x07D0: 2,000 Hz (max) *Note 3
3	Ch0_frequency (MSB)	
4	Ch1_frequency (LSB)	
5	Ch1_frequency (MSB)	
6	Ch2_frequency (LSB)	
7	Ch2_frequency (MSB)	
8	Ch3_frequency (LSB)	
9	Ch3_frequency (MSB)	
A	Ch4_frequency (LSB)	
B	Ch4_frequency (MSB)	

Note 1: Playback may not be possible as desired if 0xAA or more is set.

Note 2: Bit[6:0] must be set to “0x00: No mute interval” if Bit[7] is set to “1: Continuous.”

Note 3: Playback may not be possible as desired if a value of Fs/2 or more is set. No sound (mute) is output if 0x00 is set. Not guaranteed if set to 0x07D1.

Note 4: This message can also be sent divided into multiple parts if required by the host device. The file_event field bytes should be set to 4, 8, 16, 32, 64, 128, 256, 512, 1024, or 2048 for divided transmission. This does not apply to the final transmission. The individual divided message lengths can be entered in the length field for divided transmission, but the total value should be entered for num_files.

Note 5: It is recommended that the message output from the “S1V30080 Series Sound Tool” be used for this message.

5.6 Sound playback control messages

5.6.1 ISC_EASY_START_REQ

Table 5.12 ISC_EASY_START_REQ

Direction	Host device to S1V30080	
Purpose	Used for starting memory storage command playback	
Byte	**Field**	**Value**
0	length (LSB)	0x0006
1	length (MSB)	
2	msg_id (LSB)	0x0088: ISC_EASY_START_REQ
3	msg_id (MSB)	
4	file_index_number	Index number of memory storage command stored in internal ROM or external flash memory *Note 1
5	reserved	Set to 0x00

Note 1: Only memory storage commands can be specified, and audio data cannot be specified. 0xFF cannot be specified either. For details of the correlation between memory storage commands and File_Index_Number, refer to “ROMImage_yymmdd_hhmmss.csv” output from the “S1V30080 Series Sound Tool.”

5.6.2 ISC_SOUND_START_REQ

Table 5.13 ISC_SOUND_START_REQ

Direction	Host device to S1V30080	
Purpose	Used for starting playback set by ISC_*_SET_REQ	
Byte	**Field**	**Value**
0	length (LSB)	0x0008
1	length (MSB)	
2	msg_id (LSB)	0x0080: ISC_SOUND _START_REQ
3	msg_id (MSB)	
4	sequencer_main_start	0x00: Retain state 0x01: Start playback *Note 1
5	sequencer_multi_start	0x00: Retain state 0x01: Start playback *Note 1
6	synthesizer_start	0x00: Retain state 0x01: Start playback *Note 1 *Note 2
7	reserved	Set to 0x00

Note 1: 0x00 (Retain state) will be set if an unsupported value (i.e. any value other than 0x00 or 0x01) is sent. “Retain state” continues playback if playback is in progress or maintains the non-operative state if non-operative. If “Start playback” is set during playback, this is treated as “Retain state.”

Note 2: If “Start playback” is set, synthesizer melody playback is started if the last message was ISC_ SYNTHESIZER_MELODY_SET_REQ, and synthesizer buzzer/tone playback is started if the last message was ISC_SYNTHESIZER_BUZZ_TONE_SET_REQ.

5.6.3 ISC_SOUND_STOP_REQ

Table 5.14 ISC_SOUND_STOP_REQ

Direction	Host device to S1V30080	
Purpose	Used for stopping playback	
Byte	**Field**	**Value**
0	length (LSB)	0x0008
1	length (MSB)	
2	msg_id (LSB)	0x0084: ISC_SOUND _STOP_REQ
3	msg_id (MSB)	
4	sequencer_main_stop	0x00: Retain state 0x01: Stop playback *Note 1
5	sequencer_multi_stop	0x00: Retain state 0x01: Stop playback *Note 1
6	synthesizer_stop	0x00: Retain state 0x01: Stop playback *Note 1
7	reserved	Set to 0x00

Note 1: 0x00 (Retain state) will be set if an unsupported value (i.e. any value other than 0x00 or 0x01) is sent. “Retain state” continues playback if playback is in progress or maintains the non-operative state if non-operative.

5.6.4 ISC_AUDIO_VOLUME_REQ

Table 5.15 ISC_AUDIO_VOLUME_REQ

Direction	Host device to S1V30080	
Purpose	Used for real-time volume control	
Byte	**Field**	**Value**
0	length (LSB)	0x000C
1	Length (MSB)	
2	msg_id (LSB)	0x90: ISC_AUDIO_VOLUME_REQ
3	msg_id (MSB)	
4	reserved	set to 0x00
5	audio_gain_seq_main_inc	0x82: +63.0 dB 0x84: +62.0 dB ... ... 0xFC: +2.0 dB 0xFE: +1.0 dB 0x00: No change (Uses value set by ISC_AUDIO_CONFIG_REQ) 0x02: -1.0 dB 0x04: -2.0 dB ... ... 0x7C: -62.0 dB 0x7E: -63.0 dB *Note 1
6	audio_gain_seq_multi_inc	
7	audio_gain_synthe_ch0_inc	
8	audio_gain_synthe_ch1_inc	
9	audio_gain_synthe_ch2_inc	
A	audio_gain_synthe_ch3_inc	
B	audio_gain_synthe_ch4_inc	

Note 1: This message uses a 1 dB step resolution. Settings with 0.5 dB step cannot be guaranteed. Note also that the settings for this message are summed for the ISC_AUDIO_CONFIG_REQ gain settings.

Note 2: 0 dB is set if the sum of the ISC_AUDIO_CONFIG_REQ gain settings with the settings for this message is 0 db or more, and is set to Mute if -63.5 dB or less.

5.6.5 ISC_AUDIO_MUTE_REQ

Table 5.16 ISC_AUDIO_MUTE_REQ

Direction	Host device to S1V30080	
Purpose	Used for muting audio output	
Byte	**Field**	**Value**
0	length (LSB)	0x0006
1	length (MSB)	
2	msg_id (LSB)	0x0094: ISC_AUDIO_MUTE_REQ
3	msg_id (MSB)	
4	audio_mute_enable	0x00: Disable mute 0x01: Enable mute *Note 1
5	reserved	Set to 0x00

Note 1: 0x00 (Disable mute) will be set if an unsupported value (i.e. any value other than 0x00 or 0x01) is sent.

5.7 Status confirmation messages

5.7.1 ISC_STATUS_REQ

Table 5.17 ISC_STATUS_REQ

Direction	Host device to S1V30080	
Purpose	Used for checking S1V30080 internal state	
Byte	**Field**	**Value**
0	length (LSB)	0x0008
1	length (MSB)	
2	msg_id (LSB)	0x00E0: ISC_STATUS_REQ
3	msg_id (MSB)	
4	status_code	0x01: Operating status 0x02: Error confirmation 0x05: Previously correctly recognized message 0x10: clock_divide setting confirmation 0x11: chksum_set setting confirmation 0x12: audio_sample_rate setting confirmation 0x13: audio_gain setting confirmation 0x15: DAC_bit_width setting confirmation 0x16: sequencer_main setting confirmation 0x18: sequencer_multi setting confirmation 0x1A: synthesizer_melody_config setting confirmation 0x1C: synthesizer_melody_set/start setting confirmation 0x1E: synthesizer_buzz_tone_set setting confirmation 0x20: mute setting status confirmation
5	reserved	Set to 0x00
6	reserved	Set to 0x00
7	reserved	Set to 0x00

5.7.2 ISC_STATUS_RESP

Table 5.18 ISC_STATUS_RESP

Direction	S1V30080 to Host device	
Purpose	Used for providing S1V30080 internal state in response	
Byte	**Field**	**Value**
0	length (LSB)	0x000C
1	length (MSB)	
2	msg_id (LSB)	0x00E1: ISC_STATUS_RESP
3	msg_id (MSB)	
4	status(LSB)	See Table 5.19.
5	status	
6	status	
7	status	
8	status	
9	status	
A	status	
B	status(MSB)	

Table 5.19 ISC_STATUS_RESP details

Status_req code (hex)	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
Status confirmation								
Operating status (0x01)	0x00000001: Audio main playback in progress 0x00000002: Audio multi playback in progress 0x00000004: Synthesizer melody playback in progress 0x00000008: Synthesizer buzzer/tone playback in progress				0x00	0x00	0x00	0x00
Error status (0x02)	0x0000: No error 0x0001: Error		Error code *Note 1		0x00	0x00	0x00	0x00
Previous message confirmation (0x05)	Previous message code *Note 2		0x00	0x00	0x00	0x00	0x00	0x00
System control setting confirmation								
Clock_divide setting (0x10)	clock _divide	0x00	0x00	0x00	0x00	0x00	0x00	0x00
Check_sum setting (0x11)	chksum _set	0x00	0x00	0x00	0x00	0x00	0x00	0x00
Sound output setting confirmation								
Audio_sample_rate setting (0x12)	audio _sample _rate	0x00	0x00	0x00	0x00	0x00	0x00	0x00
Audio_gain setting (0x13) *Note 3	sequencer _main	sequencer _multi	synth0	synth1	synth2	synth3	synth4	0x00
DAC_bit_width (0x15)	DAC_bit _width	0x00	0x00	0x00	0x00	0x00	0x00	0x00
Audio playback setting confirmation								
Sequencer_main setting (0x16)	play_count		num_files	0x00	0x00	0x00	0x00	0x00
Sequencer_multi setting (0x18)	play_count		num_files	0x00	0x00	0x00	0x00	0x00
Synthesizer playback setting confirmation								
Synthesizer_melody_ config setting (0x1A)	tempo	Attack _level	Attack _step	Decay _level	Decay _step	Release _step	0x00	0x00
Synthesizer_melody_ set/Start setting (0x1C)	play_count		num_files		0x00	0x00	0x00	0x00
Synthesizer_buzz_tone _set setting (0x1E)	play_count		Num_files		0x00	0x00	0x00	0x00
Mute setting confirmation								
Audio_mute setting (0x20)	0x00 : disable 0x01 : enable	0x00	0x00	0x00	0x00	0x00	0x00	0x00

Note 1: For error code details, refer to Table 5.20. The first occurring error code is retained and output. To clear error codes, set the SYSTEM_EN pin to Low.

Note 2: Outputs the previously received message ID.

Note 3: Outputs the volume set internally. Note that 0x7F will be output for values over 0x7F, and 0x00 will be output for value less than 0x00.

Note 4: This message is output using LSB first.

Table 5.20 Error codes

Error code	Associated message	Meaning
General error codes		
0x4001	Message error	A message ID not in the protocol was specified.
0x4002	Checksum error	The calculated data sum does not match the checksum value.
0x4003	Flash read error	The data written to the flash memory is illegal data.
System control error codes		
0x4010	ISC_CLKDIV_CONFIG_REQ	This message was sent during playback. The message becomes invalid.
0x4011	ISC_CHKSUM_CONFIG_REQ	This message was sent during playback. The message becomes invalid.
Sound output setting error codes		
0x4018	ISC_AUDIO_CONFIG_REQ	This message was sent during playback. The message becomes invalid.
Audio playback setting error codes		
0x4020	ISC_SEQUENCER_MAIN_CONFIG_REQ	This message was sent during audio main playback. The message becomes invalid.
0x4021	ISC_SEQUENCER_MAIN_CONFIG_REQ	file_index_number is invalid (0xFF or outside range). To reset this error, set the SYSTEM_EN pin to Low.
0x4022	ISC_SEQUENCER_MAIN_CONFIG_REQ	Memory storage command was specified by this message. To reset this error, set the SYSTEM_EN pin to Low.
0x4028	ISC_SEQUENCER_MULTI_CONFIG_REQ	This message was sent during audio multi playback. The message becomes invalid.
0x4029	ISC_SEQUENCER_MULTI_CONFIG_REQ	file_index_number is invalid (all 5 channels are 0xFF or outside range). To reset this error, set the SYSTEM_EN pin to Low.
0x402A	ISC_SEQUENCER_MULTI_CONFIG_REQ	Memory storage command was specified by this message. To reset this error, set the SYSTEM_EN pin to Low.

Synthesizer playback setting error code		
0x4060	ISC_ SYNTHESIZER_MELODY_CONFIG_REQ	This message was sent during playback. The message becomes invalid.
0x4064	ISC_ SYNTHESIZER_MELODY_SET_REQ	This message was sent during synthesizer melody playback. The message becomes invalid.
0x4066	ISC_ SYNTHESIZER_BUZZ_TONE_SET_REQ	This message was sent during synthesizer buzzer/tone playback. The message becomes invalid.
Sound output error code		
0x4080	ISC_SOUND_START_REQ	Audio main playback was set using this message before setting ISC_SEQUENCER_MAIN_CONFIG_REQ. In this case, audio main playback is not started.
0x4081	ISC_SOUND_START_REQ	Audio multi playback was set using this message before setting ISC_SEQUENCER_ MULTI_CONFIG_REQ. In this case, audio multi playback is not started.
0x4082	ISC_SOUND_START_REQ	Synthesizer playback was set using this message before setting ISC_SYNTHESIZER_*_ SET_REQ. In this case, synthesizer playback is not started.
0x4090	ISC_ EASY_START_REQ	file_index_number is invalid (0xFF or outside range). The message becomes invalid.
0x4091	ISC_ EASY_START_REQ	The audio data file_index_number was specified. The message becomes invalid.
0x4092	ISC_EASY_START_REQ	This message was sent four or more times before sound playback. Messages after the third time become invalid.
0x4093	ISC_EASY_START_REQ	Memory storage main command was specified by this message during audio main playback. The message becomes invalid.
0x4094	ISC_EASY_START_REQ	Memory storage multi command was specified by this message during audio multi playback. The message becomes invalid.
0x4095	ISC_EASY_START_REQ	Memory storage melody command or memory storage buzzer/tone command was specified by this message during synthesizer playback. The message becomes invalid.

Revision History

Date	Revision details			
	Rev.	Page	Type	Details
03/10/2009	1.00	All	New	Newly established

EPSON

International Sales Operations

AMERICA

EPSON ELECTRONICS AMERICA, INC.
2580 Orchard Parkway,
San Jose, CA 95131, USA
Phone: +1-800-228-3964 FAX: +1-408-922-0238

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.
7F, Jinbao Bldg., No.89 Jinbao St.,
Dongcheng District,
Beijing 100005, CHINA
Phone: +86-10-6410-6655 FAX: +86-10-6410-7320

SHANGHAI BRANCH
7F, Block B, Hi-Tech Bldg., 900 Yishan Road,
Shanghai 200233, CHINA
Phone: +86-21-5423-5522 FAX: +86-21-5423-5512

SHENZHEN BRANCH
12F, Dawning Mansion, Keji South 12th Road,
Hi-Tech Park, Shenzhen 518057, CHINA
Phone: +86-755-2699-3828 FAX: +86-755-2699-3838

EPSON HONG KONG LTD.
20/F, Harbour Centre, 25 Harbour Road,
Wanchai, Hong Kong
Phone: +852-2585-4600 FAX: +852-2827-4346
Telex: 65542 EPSCO HX

EPSON TAIWAN TECHNOLOGY & TRADING LTD.
14F, No. 7, Song Ren Road,
Taipei 110, TAIWAN
Phone: +886-2-8786-6688 FAX: +886-2-8786-6660

EPSON SINGAPORE PTE., LTD.
1 HarbourFront Place,
#03-02 HarbourFront Tower One, Singapore 098633
Phone: +65-6586-5500 FAX: +65-6271-3182

SEIKO EPSON CORP.
KOREA OFFICE
50F, KLI 63 Bldg., 60 Yoido-dong,
Youngdeungpo-Ku, Seoul 150-763, KOREA
Phone: +82-2-784-6027 FAX: +82-2-767-3677

SEIKO EPSON CORP.
SEMICONDUCTOR OPERATIONS DIVISION

IC Sales Dept.
IC International Sales Group
421-8, Hino, Hino-shi, Tokyo 191-8501, JAPAN
Phone: +81-42-587-5814 FAX: +81-42-587-5117

Document Code: 411691200
First Issue April 2009 in JAPAN ⓓ