EPSON
EXCEED YOUR VISION

S1V30120
Message Protocol Specification

SEIKO EPSON CORPORATION

Rev.1.17

NOTICE

Table of Contents

1. Introduction

1.1 Scope

This document specifies the message protocol used to configure, control and transfer data to the S1V30120. It includes details of both the messages used in S1V30120 boot mode and the messages used in S1V30120 main mode.

This document will be updated for each release of the S1V30120 firmware that includes any changes to the message protocol.

1.2 Document Structure

Section 2 summarises the general features of the S1V30120 necessary for the description of the message protocol.

Section 3 summarises the general features of the message protocol used for communicating with the S1V30120 and the mapping of this to the S1V30120 hardware interface.

Section 4 describes the messages used in boot mode in order to download initialisation data and switch to S1V30120 main mode.

Section 5 describes the messages used in main mode. It includes procedural flows for operating the S1V30120 and also summarises error codes.

2. Feature Summary

2.1 S1V30120 Feature Summary

The S1V30120 supports the following applications.

- Unrestricted Text-To-Speech (TTS)
- ADPCM Decoding

2.2 Interface Summary

The S1V30120 has the following interfaces.

- Flexible Serial Interface (SPI-Slave)
- GPIO Interface
- 16-bit Full-digital amplifier

The Flexible Serial Interface is configured as an SPI Slave interface to support bi-directional transfer of control and data between the host and S1V30120.

The full digital amplifier interface supports output of audio data from S1V30120.

Through the message protocol, the GPIO interface may be used for the outputting of control signals on the S1V30120 GPIO pins.

3. Message Protocol

The S1V30120 operates as a companion device. It is connected to a Host using the Serial Peripheral Interface (SPI).

When connected to a Host, the operation of the S1V30120 is completely determined by control and data information provided by the Host. Control and data information is transferred to and from S1V30120 using a defined message protocol.

A model of host and S1V30120 communication is presented in Figure 1.

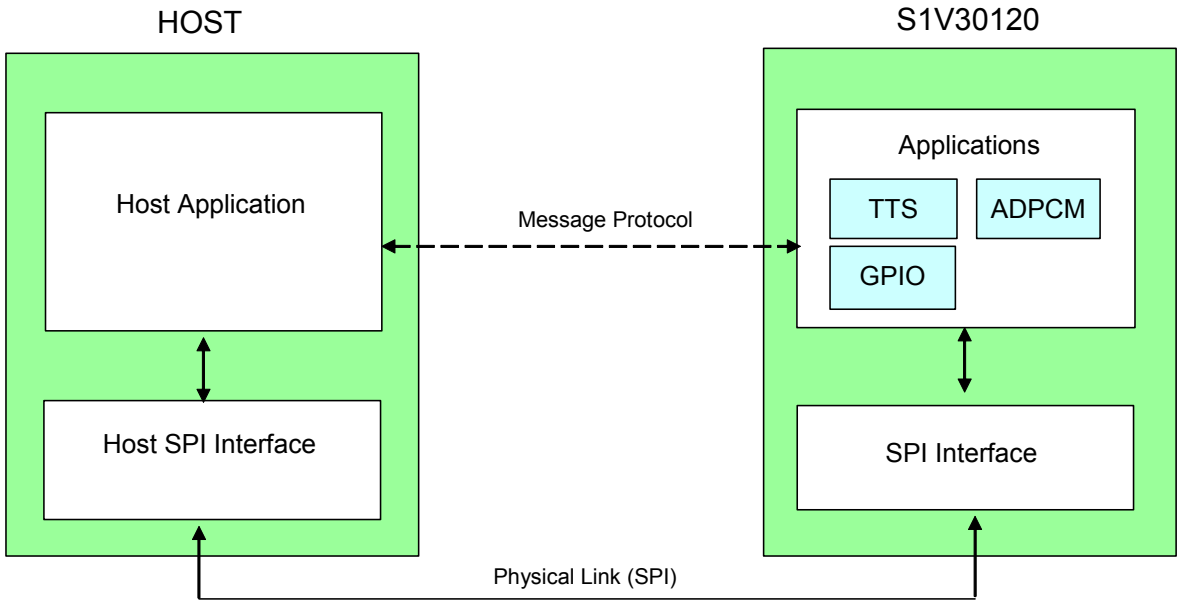


Figure 1 Host - S1V30120 Communication

3.1 General Features

The features of the messaging protocol are presented below:

- The S1V30120 is completely driven by the reception of messages from the Host.
- The message set is byte oriented, with a fixed length header part, and variable length payload
- The messages follow a REQuest –> RESPonse flow. All REQuest messages are sent by the host processor. All RESPonse messages are generated by the S1V30120 in response to a REQuest message.
- The host processor is required to wait for a RESPonse to a REQuest message, before issuing any subsequent REQuest. Any REQuest message sent before receiving the previous RESPonse will trigger an error state within the S1V30120.
- Some messages are termed INDication messages. Such messages require no confirmation from the recipient. INDication messages can only be sent by the S1V30120*

The general flow of messages is exemplified in the following diagram:

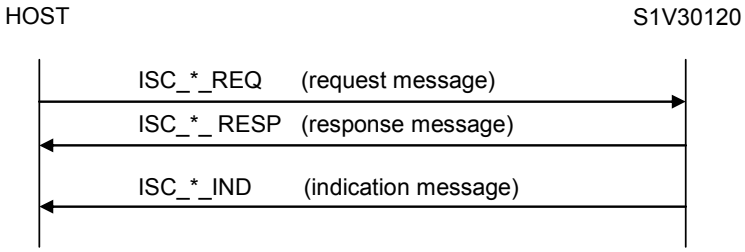


Figure 2 Generic Message Flow

* The one exception to this rule is the ISC_PMAN_STANDBY_EXIT_IND message.

In general, a RESPonse message is sent immediately on receiving its corresponding REQuest message. The sending of a RESPonse informs the host that the REQuest has been received but not that it has been actioned.

However for some messages, notably ISC_SPCODEC_START_REQ and STOP_REQ, the interval between the sending of a REQuest and receipt of a RESPonse can be considerable. In corner cases, the delay between the sending of a REQuest message by a Host and the receipt of the corresponding RESPonse message can be several hundred milliseconds.

For the purpose of implementing failsafe functionality the delay between REQuest and RESPonse is guaranteed not to exceed 500ms.

3.2 Hardware Configuration

Certain GPIO pins are reserved for device configuration or for use by the message protocol. The following table summarises the reserved GPIO pin usage for correct device operation and also pins used by the message protocol.

Table 1 GPIO Port A Usage

GPIO Pin	Purpose	Input / Output
GPIOA0	BOOT_MODE_RUN 0x00: reserved 0x01: run boot mode	Input
GPIOA2:A1	BOOT_MODE_SELECT 0x00: reserved 0x01: boot using SPI Slave Interface 0x02: reserved 0x03: reserved	Input
GPIOA3	MSG_READY (Active High) Indicates when a message is ready for transfer from the S1V30120 to the host.	Output
GPIOA4	Reserved: set to Low.	Input

3.3 Message Structure

All messages that are received and sent by the S1V30120 are termed ISC (Inter-System-Communication) messages.

An ISC message consists of a fixed length header part and a variable length payload. The following figure shows this, along with a definition of the structure of the header part.

Table 2 ISC Message Structure

	Byte	Interpretation
HEADER PART	0 (LSB)	Length of Message (Number of bytes including header)
	1 (MSB)	
	2 (LSB)	Message ID (Identifies Payload)
	3 (MSB)	
PAYLOAD PART	4	Variable Length Payload (Contains control fields and/or data)

3.3.1 Boot Mode Messages and Main Mode Messages

The S1V30120 runs in boot mode after hardware reset. This must be used to initialise the S1V30120 before sending any main mode messages. The procedure for initialisation and the messages used by boot mode are described in Section 4. Once initialised, main mode uses a second message set, the main mode messages; these messages are described in Section 5.

3.3.2 Message Length

A limit is placed on the maximum length of any ISC message that can be transferred to the S1V30120. For details of the maximum lengths of messages used in boot mode and main mode refer to sections 4 and 5 detailing the message sets.

3.4 Message Mapping for SPI Slave Interface Protocol

The Host is an SPI master and the S1V30120 a slave for all SPI communications. The SPI transfer uses an 8-bit word transfer.

The SPI Slave interface has two independent channels for transferring messages to and from S1V30120. The input channel to the S1V30120 is on the SIN pin, the output channel is on SOUT.

Message transfers on each channel are independent – i.e. messages can be transferred on both channels simultaneously. So for example whilst a REQ message is being transmitted on the input channel by the host, the S1V30120 could be sending an IND message on the output channel. This requires the Host to support full duplex communication.

In the SPI Slave Interface Protocol, an 8-bit word must always be transferred in both directions on transmit and receive channels. SPI padding words are used to flush the communication channel at the end of a message. They are also used by the Host and S1V30120 to pad the transmit channel while receiving a message. Start message words are used to mark the start of a new message. The SPI words used for control within the SPI Slave Interface Protocol are shown in Table 3.

Table 3 SPI Control Words

Control Word	Value	Description
Padding Word	0x00	Pad the channel.
Start Message Word	0xAA	Mark start of message

The transfer of a message is shown in Table 4. To begin a message the host must first send the Start Message Control Word. This will be followed by the ISC message header and contents. After the host has transmitted a message its serial clock is required to continue running for 128 serial clock cycles to flush the S1V30120 receive channel. This can be achieved by having the Host send a further 16 padding words on it's transmit data channel after it has sent each message.

The receive channel should also be flushed with 128 clock cycles after each message received. Again, this can be achieved by the Host sending 16 padding words on it's transmit data channel after it has received each message.

As transmit and receive channels are independent, during the sending of either the message or the subsequent padding words it is possible that either an IND message or a RESP message will be sent to the host.

The message flow for a generic message transfer is shown in Figure 3.

Table 4 SPI ISC Message Transfers

Words	Value	Description
-N to -1	0x00	N Padding Words
0	0xAA	Start Message Command
1 ... ISC Message Length	ISC Message	

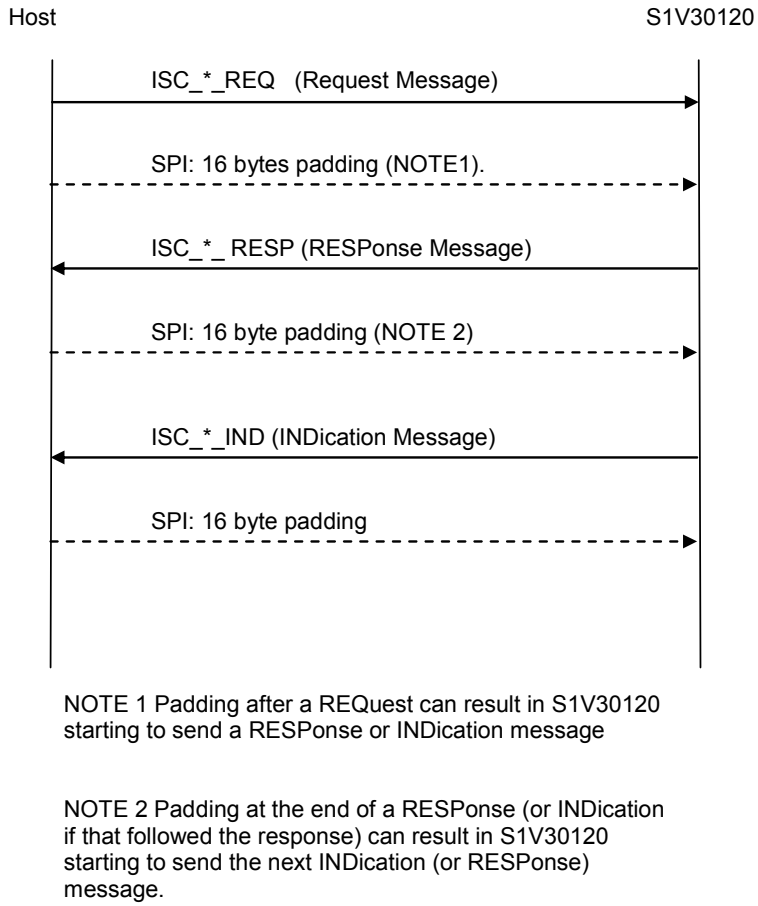


Figure 3 Generic Message Flow and Padding

The sending and receiving of padding bytes is understood to be implicit in all message flow diagrams presented in this document.

As well as SIN, SOUT and SCLK, S1V30120 SPI message transfers make use of the MSG_READY output pin. The MSG_READY output is asserted when the S1V30120 is ready to transmit a message to the host and will be de-asserted before the next message is ready to be transmitted. The recommendation is for the Host to detect the positive edge of MSG_READY and continue to clock the SPI link until the message and all padding has been received.

NOTE1: The ISC_BOOT_RUN_REQ/RESP messages and the ISC_PMAN_STANDBY_EXIT_IND message sent by the Host to S1V30120 have special padding requirements. For specification of these requirements refer to section 4.4.2.5.

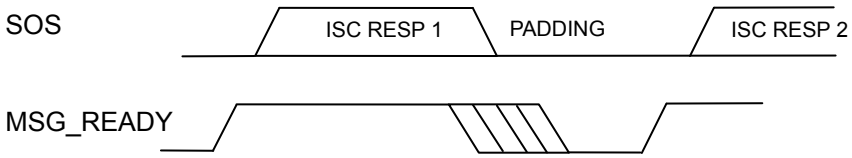


Figure 4 MSG_READY timing

3.5 Streaming Considerations

3.5.1 Input Buffering

The maximum amount of input data, text or encoded speech that can be accepted in any single data bearing ISC message (e.g. an ISC_SPCODEC_START_REQ or ISC_TTS_SPEAK_REQ) is **2048 bytes.** In main mode, the maximum ISC message length is greater than this to allow for the message header and any control fields in the payload of the ISC message.

3.5.2 Real Time Constraints

For ADPCM streaming, two key factors need to be considered to ensure that the host can stream data to the S1V30120 in a timely manner. The minimum SPI bit rate ($F_{s,min}$) and the maximum delay in the host in responding to an ISC_SPCODEC_READY_IND ($T_{h,max}$).

Maximum SPI frequency for S1V30120 is as follows:

$\boldsymbol{F_{s,max}}$ **= 1MHz**

Minimum SPI frequency for S1V30120 depends on the host response time to an ISC_SPCODEC_READY_IND (T_h).

This can be calculated as follows:

$$F_{s,min} = 8*N*F_d \quad / ((8*N) - (T_h*F_d))$$

where:

F_s : SPI frequency (Hz)

F_d : ADPCM data rate (bps)

N : Number of bytes transferred in a single data bearing message

T_h : worst case host response time to SPCODEC_READY_IND messages (s)

Alternatively, for a fixed SPI frequency, F_s, the maximum tolerated response time from the host can be calculated as follows:

$$T_{h,max} = \quad 8*N(F_s-F_d) / (F_s*F_d)$$

The following tables summarise $F_{s,min}$ and $T_{h,max}$ values for:

F_d = 64kps

N = 2048

F_s (kHz)	$T_{h,max}$ (ms)
100	92
150	146
200	174

T_h (ms)	$F_{s,min}$ (kHz)
10	67
50	80
100	105

NOTE: The calculated values for $F_{s,min}$ and $T_{h,max}$ are limit cases and are shown for illustrative purposes only. For reliable operation it is recommended that the host configuration does not result in T_h or F_s being set at these limit values.

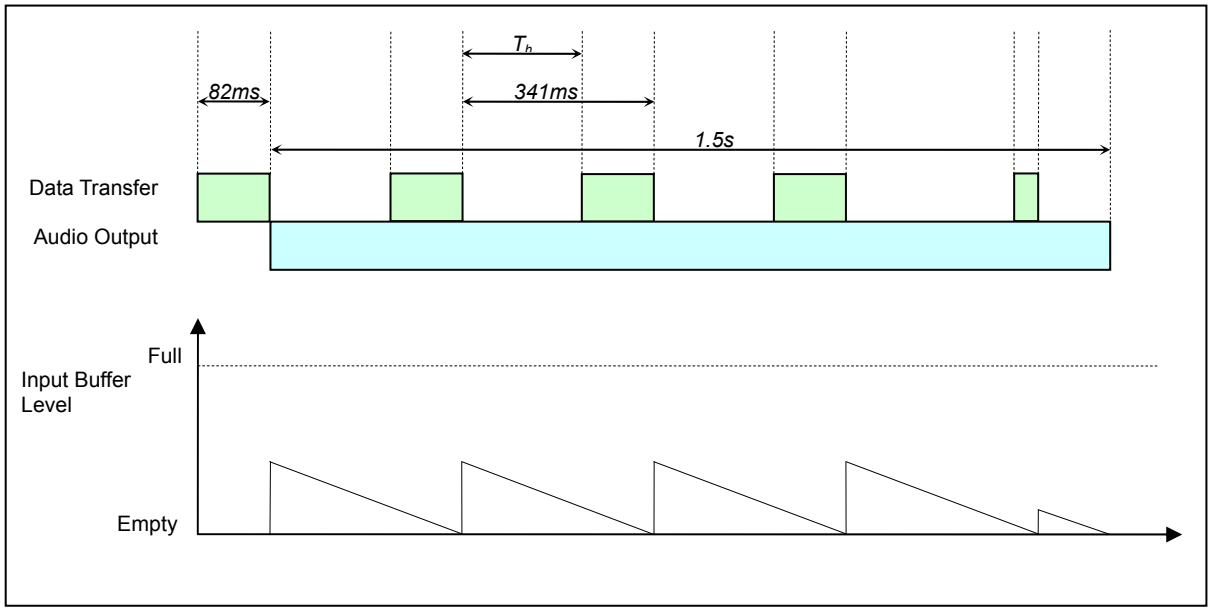


Figure 5 Example Streaming Case: ADPCM 48kbps, F_s=200kHz, 1.5s Audio Data

It is not necessary to consider the constraints associated with TTS streaming as the device is limited to processing single buffers of TTS text at a time.

4. Boot Mode Messages

4.1 Introduction

The S1V30120 boot mode is entered on hardware reset. The S1V30120 requires the download of initialisation data in boot mode before it is run in main mode. This initialisation data is contained in the S1V30120_INIT_DATA binary file included with this documentation.

The boot mode messages are used for the download of this S1V30120_INIT_DATA. This file should be separated into blocks for sending to the Host in the payload of ISC_BOOT_LOAD_RESP messages. The recommended size for these blocks is 2048 bytes. The final block will contain the remainder of the initialisation data after all 2048 byte blocks have been loaded by the Host.

4.2 Message Lengths

For boot mode, the maximum message length is 2048 bytes (Hex: 0x800 bytes).

4.3 Message Identifiers Summary

The boot mode messages are summarised in Table 5. Message contents are described in Section 4.4.2.

Table 5 Boot Mode Messages

Message ID	Purpose	Value	Table Reference
ISC_VERSION_REQ	Version Request Message	0x0005	Table 6
ISC_VERSION_RESP		0x0006	Table 7
ISC_BOOT_LOAD_REQ	Loads code image into SRAM	0x1000	Table 8
ISC_BOOT_LOAD_RESP		0x1001	Table 9
ISC_BOOT_RUN_REQ	Start executing image.	0x1002	Table 10
ISC_BOOT_RUN_RESP		0x1003	Table 11

4.4 Boot Mode Messages

4.4.1 Message Flows

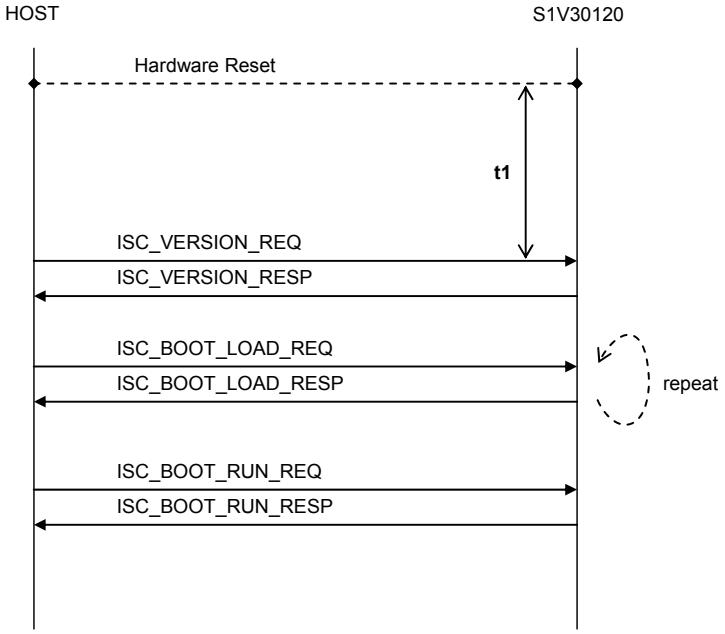
4.4.1.1 Boot Mode

Boot mode is run after hardware reset so the user can perform the mandatory download of the initialisation data.

After reset, time t1 (see Figure 6) must be observed to allow for system start up. After, this an ISC_VERSION_REQ/RESP message flow can be used to identify the hardware identifier of the part.

The ISC_BOOT_LOAD_REQ/RESP sequence is used to download initialisation data to the S1V30120.

Finally, an ISC_BOOT_RUN_REQ/RESP is used to enter main mode. The SPI padding requirements for these messages are different from those described in section 3.4. For details see sections 4.4.2.5and 4.4.2.6.



Symbol	Parameter	Min	Max	Unit
t1	Time from Reset Initialisation to communication with S1V30120 in boot mode.	120	n/a	ms
Note1	Sending padding bytes during period t1 is forbidden.			

Figure 6 BOOT LOADER Message Flow

4.4.2 Message Description

4.4.2.1 ISC_VERSION_REQ (Boot Mode)

Table 6 ISC_VERSION_REQ (Boot Mode)

Direction	HOST to S1V30120	
Purpose	Used to request information about the chip hardware version. The message format is the same as the ISC_VERSION_REQ message in main mode. See Table 17.	
Byte	**Field**	**Values**
0	length (lsb)	0x0004
1	length (msb)	
2	msg_id (lsb)	ISC_VERSION_REQ
3	msg_id (msb)	

4.4.2.2 ISC_VERSION_RESP (Boot Mode)

Table 7 ISC_VERSION_RESP (Boot Mode)

Direction	S1V30120 to HOST	
Purpose	Response to ISC_VERSION_REQ. The message format is the same as the ISC_VERSION_RESP message in main mode. See Table 18.	
Byte	**Field**	**Values**
0	length (lsb)	0x000F
1	length (msb)	
2	msg_id (lsb)	ISC_VERSION_RESP
3	msg_id (msb)	
4	hw_id_int	Hardware version identifier, integer part
5	hw_id_frac	Hardware version identifier, fractional part
6	fw_version_int	Not defined for boot mode.
7	fw_version_frac	Not defined for boot mode.
8	fw_features	Not defined for boot mode.
9		
10		
11		
12	fw_extended_features	Not defined for boot mode.
13		
14		
15		

4.4.2.3 ISC_BOOT_LOAD_REQ

Table 8 ISC_BOOT_LOAD_REQ

Direction	HOST to S1V30120	
Purpose	Loads program code into SRAM. Subsequent ISC_BOOT_LOAD_REQ messages will continue writing program code into SRAM, at the point where the previous ISC_BOOT_LOAD_REQ finished. The size of the message should not exceed 2048 bytes.	
Byte	**Field**	**Values**
0	length_lsb	0x0004 + length(boot_data) VARIABLE
1	length_msb	
2	msg_id_lsb	ISC_BOOT_LOAD_REQ
3	msg_id_msb	
	boot_data	

4.4.2.4 ISC_BOOT_LOAD_RESP

Table 9 ISC_BOOT_LOAD_RESP

Direction	S1V30120 to HOST	
Purpose	Response to ISC_BOOT_LOAD_REQ. Indicates whether initialisation data was successfully loaded. Note the receipt of this message indicates boot sequence completion.(see Figure 7)	
Byte	**Field**	**Values**
0	length_lsb	0x0006
1	length_msb	
2	msg_id_lsb	ISC_BOOT_LOAD_RESP
3	msg_id_msb	
4	boot_load_success (lsb)	0x0001 indicates success – all other values indicate failure.
5	boot_load_success (msb)	

4.4.2.5 ISC_BOOT_RUN_REQ

Table 10 ISC_BOOT_RUN_REQ

Direction	HOST to S1V30120	
Purpose	After completion of loading the initialisation data, ISC_BOOT_RUN_REQ is used to switch to running in main mode. Padding after this message must be completed within the time interval t1 shown in Figure 7. (NOTE 1).	
Byte	**Field**	**Values**
0	length_lsb	0x0004
1	length_msb	
2	msg_id_lsb	ISC_BOOT_RUN_REQ
3	msg_id_msb	

NOTE 1: The padding requirements after this message are different from those described in Section 3.4. After sending ISC_BOOT_RUN_REQ the host must send 8 padding words.

4.4.2.6 ISC_BOOT_RUN_RESP

Table 11 ISC_BOOT_RUN_RESP

Direction	S1V30120 to HOST	
Purpose	Response to ISC_BOOT_RUN_RESP. Note that this message signals boot sequence completion. Padding after this message must be completed within the time interval t1 shown in Figure 7. NOTE 1	
Byte	**Field**	**Values**
0	length_lsb	0x0006
1	length_msb	
2	msg_id_lsb	ISC_BOOT_RUN_RESP
3	msg_id_msb	
4	boot_run_success (lsb)	0x01 indicates success – all other values indicate failure.
5	boot_run_success (msb)	

NOTE 1: The padding requirements after this message are different from those described in Section 3.4. After receiving ISC_BOOT_RUN_RESP the host must send 8 padding bytes.

5. Main Mode Messages

5.1 Introduction

This section describes the message protocol used in main mode. The messages are summarised in Section 5.3. Standard error codes that a user may see generated in main mode are listed in Section 5.4.

In subsequent sections, the messages are grouped by function and described in detail. There is an overview section for each message group that describes the functions of the message group and a Message Flow section, which describes the general flow of messages within a message group.

The message flows presented in these sections are the typical flows in main mode. This message protocol is more flexible than the flows presented. In general, any request message can be sent at any time after the last response message has been received. If it is inappropriate to send a request message at a certain point, then the S1V30120 will indicate this with a non-fatal error code in the response message.

5.2 Maximum Message Lengths

The maximum length of a message in main mode is 2116 bytes (Hex: 0x844 bytes)

5.3 Message Identifiers Summary

Table 12 summarises the current supported messages and message ID values..

Table 12 Message Identifiers Summary

Message ID	Purpose	Value	Reference
System Messages			**Section 5.5**
ISC_TEST_REQ	Test & register host hw interface	0x0003	Table 15
ISC_TEST_RESP		0x0004	Table 16
ISC_VERSION_REQ	Version & support info	0x0005	Table 17
ISC_VERSION_RESP		0x0006	Table 18
ISC_ERROR_IND	Fatal Error Indication	0x0000	Table 19
ISC_MSG_BLOCKED_RESP	Request blocked.	0x0007	Table 20
Audio Messages			**Section 5.7**
ISC_AUDIO_CONFIG_REQ	Configure Audio Output	0x0008	Table 26
ISC_AUDIO_CONFIG_RESP		0x0009	Table 27
ISC_AUDIO_VOLUME_REQ	Set volume (analogue gain)	0x000A	Table 28
ISC_AUDIO_VOLUME_RESP		0x000B	Table 29
ISC_AUDIO_MUTE_REQ	Mute audio output	0x000C	Table 30
ISC_AUDIO_MUTE_RESP		0x000D	Table 31

Power Management Messages			**Section 5.6**
ISC_PMAN_CONFIG_REQ	Power manager configuration	0x0062	Table 21
ISC_PMAN_CONFIG_RESP		0x0063	Table 22
ISC_PMAN_STANDBY_ENTRY_REQ	Request entry to standby mode	0x0064	Table 23
ISC_PMAN_STANDBY_ENTRY_RESP		0x0065	Table 24
ISC_PMAN_STANDBY_EXIT_IND	Standby mode exit	0x0066	Table 25
Speech Codec Messages			**Section 5.9**
ISC_SPCODEC_CONFIG_REQ	Configure speech codec	0x0056	Table 42
ISC_SPCODEC_CONFIG_RESP		0x0057	Table 43
ISC_SPCODEC_START_REQ	Start speech codec	0x0058	Table 44
ISC_SPCODEC_START_RESP		0x0059	Table 45
ISC_SPCODEC_PAUSE_REQ	Pause speech codec audio input/output	0x005C	Table 46
ISC_SPCODEC_PAUSE_RESP		0x005D	Table 47
ISC_SPCODEC_STOP_REQ	Stop speech codec immediately	0x005A	Table 48
ISC_SPCODEC_STOP_RESP		0x005B	Table 49
ISC_SPCODEC_READY_IND	Speech codec ready for more input data	0x0060	Table 50
ISC_SPCODEC_FINISHED_IND	Speech codec output finished	0x0061	Table 51
TTS Messages			**Section 5.8**
ISC_TTS_CONFIG_REQ	Configure the TTS	0x0012	Table 32
ISC_TTS_CONFIG_RESP		0x0013	Table 33
ISC_TTS_SPEAK_REQ	Start the TTS (optionally attach data)	0x0014	Table 34
ISC_TTS_SPEAK_RESP		0x0015	Table 35
ISC_TTS_READY_IND	TTS ready for more data	0x0020	Table 36
ISC_TTS_FINISHED_IND	TTS output is finished	0x0021	Table 37
ISC_TTS_PAUSE_REQ	Pause the TTS output	0x0016	Table 38
ISC_TTS_PAUSE_RESP		0x0017	Table 39
ISC_TTS_STOP_REQ	Stop TTS immediately	0x0018	Table 40
ISC_TTS_STOP_RESP		0x0019	Table 41
ISC_TTS_UDICT_DATA_REQ	Send user dictionary data to the device	0x00CE	
ISC_TTS_UDICT_DATA_RESP		0x00D0	
GPIO Messages			
ISC_GPIO_REGISTER_REQ	Register / De-register for control of GPIO interface	0x0045	
ISC_GPIO_REGISTER_RESP		0x0046	
ISC_GPIO_OUTPUT_CONFIG_REQ	Define the GPIO output types	0x004E	
ISC_GPIO_ OUTPUT_CONFIG_RESP		0x004F	
ISC_GPIO_OUTPUT_SET_REQ	Configure GPIO output value	0x0050	
ISC_GPIO_ OUTPUT_SET_RESP		0x0051	

5.4 Error Codes

Error codes generated by the S1V30120 are partitioned as follows:

0x0000 = No Error

0x0001 to 0x3FFF = Reserved (internal) non-fatal error

0x4000 to 0x7FFF = Non-reserved (user) non-fatal error

0x8000 to 0xFFFF = Fatal error

In the normal course of operation, the HOST should not receive a reserved error from S1V30120.

Non-fatal errors can be viewed as warnings, from which the system recovers, and have no adverse effect on system behaviour.

Non-fatal errors are communicated to the HOST processor via the appropriate RESPonse message, or, in an application specific error INDication.

A RESPonse message that includes an error code indicates that the requested action could not be completed.

Fatal errors cannot be recovered by software. The only recovery mechanism is to reset the S1V30120. Fatal errors are communicated to the HOST using the ISC_ERROR_IND message.

The following tables summarise all non-fatal and fatal errors that may be triggered by the HOST:

Table 13 Non-Fatal Error Codes Summary

Error Code (hex)	Associated Messages	Meaning
General error codes:		
0x4001	N/A	Reserved
0x4002	All Speech/Audio Algorithm related messages.	Insufficient system resource to perform request.
0x4003	N/A	Unrecognised message ID
0x4002	N/A	Reserved
0x4005	All application/algorithm related messages	Application is not supported by S1V30120.
0x4006	N/A	Reserved
Audio related error codes:		
0x4020	ISC_AUDIO_CONFIG_REQ	Invalid audio configuration
0x4021	ISC_AUDIO_CONFIG_REQ	Out of range configuration value
0x4022	N/A	Reserved
0x4023	N/A	Reserved
0x4024	N/A	Reserved
0x4025	N/A	Reserved
0x4026	Any audio message	Audio message out of sequence.
0x4027		Reserved
0x4028	ISC_AUDIO_CONFIG_REQ	Incompatible routing configuration.
0x4029	ISC_AUDIO_CONFIG_REQ	Incompatible frequency configuration
0x402A	N/A	Reserved
0x402B	N/A	Reserved
0x402C	N/A	Reserved
0x4034	N/A	Reserved.
0x4039	N/A	Reserved.
Power Manager related error codes:		
0x40C0	ISC_PMAN_CONFIG_REQ	Invalid configuration
0X40C1	ISC_PMAN_STANDBY_ENTRY_REQ	S1V30120 is not ready to enter standby mode.
TTS related error codes:		
0x4040	ISC_TTS_CONFIG_REQ	Invalid language
0x4041	ISC_TTS_CONFIG_REQ	Invalid sample rate
0x4042	ISC_TTS_CONFIG_REQ	Invalid voice selection
0x4043	ISC_TTS_CONFIG_REQ	Invalid data source
0x4044	ISC_TTS_SPEAK_REQ, ISC_TTS_STOP_REQ	TTS not configured
0x4045	ISC_TTS_SPEAK_REQ	TTS not ready
0x4046	ISC_TTS_SPEAK_REQ	Audio output channel open failed
0x4047	ISC_TTS_STOP_REQ	Audio output channel close failed
0x4048	ISC_TTS_STOP_REQ	TTS already stopped
0x4049	ISC_TTS_CONFIG_REQ	Unexpected config request
0x404A	ISC_TTS_SPEAK_REQ	File already open
0x404B	ISC_TTS_SPEAK_REQ	File open failed
0x404C	ISC_TTS_STOP_REQ	File close failed
0x404D	N/A	Reserved

0x404E	N/A	Reserved
0x404F	ISC_TTS_PAUSE_REQ	Cannot pause
0x4050	N/A	Reserved
0x4051	N/A	Reserved
0x4052	ISC_TTS_UDICT_DATA_REQ	Invalid user dictionary file
0x4053	ISC_TTS_SPEAK_REQ	Invalid whilst paused
Speech Codec related error codes:		
0x4101	ISC_SPCODEC_START_REQ, ISC_SPCODEC_PAUSE_REQ	Speech codec not configured before starting
0x4102	ISC_SPCODEC_PAUSE_REQ	Pause attempted when speech codec is inactive
0x4103	ISC_SPCODEC_CONFIG_REQ	Config attempted when speech codec is active
0x4104	ISC_SPCODEC_CONFIG_REQ	Invalid data source
0x4105	N/A	Reserved
0x4106	ISC_SPCODEC_PAUSE_REQ	Speech codec already paused
0x4107	ISC_SPCODEC_PAUSE_REQ	Speech codec already active (not paused)
0x4108	ISC_SPCODEC_CONFIG_REQ	Invalid codec configuration
0x4109	ISC_SPCODEC_START_REQ	Too much input data
0x410A	N/A	Reserved
0x410B	ISC_SPCODEC_START_REQ	Cannot open audio channel
0x410C	ISC_SPCODEC_STOP_REQ	Cannot close audio channel
0x410D	N/A	Reserved
0x410E	N/A	Reserved

Table 14 Fatal Error Codes Summary

Error Code (hex)	Associated Messages	Meaning
Host IF related error codes:		
0x80E0	Any message	Unexpected message
Speech Codec related error codes:		
0x8104	ISC_SPCODEC_START_REQ	Invalid SPCODEC file / CODEC error
0x8105	ISC_SPCODEC_START_REQ	Invalid SPCODEC file / CODEC error
0x8106	ISC_SPCODEC_START_REQ	Invalid SPCODEC file / CODEC error

5.5 System Messages

5.5.1 Overview

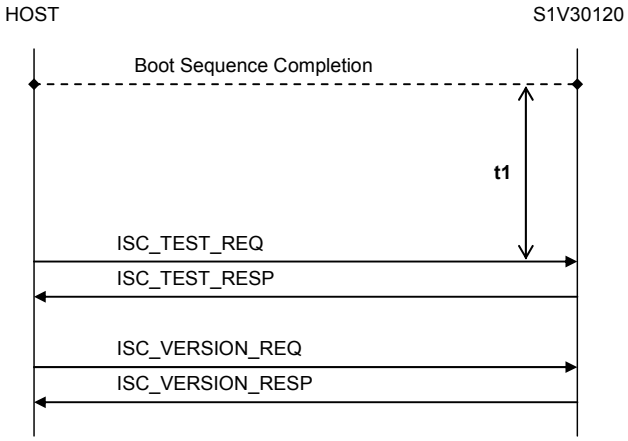
System Initialisation Messages are used for the following functions:

- S1V30120 Registration
- Firmware Version and Features Reporting
- Fatal Error Reporting

5.5.2 Message Flows

5.5.2.1 System INITIALISE

The ISC_TEST_REQ/RESP messages must be used to register the host with the S1V30120. A time t1 must be allowed before sending the ISC_TEST_REQ/RESP message to allow for system start-up. The ISC_VERSION_REQ/RESP message can then be used to request details on the features supported by the S1V30120.



Symbol	Parameter	Min	Max	Unit
t1	Time from Boot Sequence completion to initiating communication in main mode.	N/a	120	ms
Note1	Padding sent during interval t1 corresponding to the ISC_BOOT_RUN_REQ/RESP message may be sent but must be completed within interval t1. Further padding within time t1 is forbidden.			

Figure 7 System INITIALISE Message Flow

5.5.2.2 System MESSAGE BLOCKED

In response to any REQuest message, the S1V30120 may generate an ISC_MESSAGE_BLOCKED_RESP. This message is generated either because the host has not initialized the S1V30120 (see 5.5.2.1), or because there is insufficient resource to carry out the request. This can happen, for example, if a REQuest is sent to the Speech Codec, while TTS synthesis is active or is inactive but has not freed its resources.

Figure 8 System MESSAGE BLOCKED Message Flow

5.5.3 Message Descriptions

5.5.3.1 ISC_TEST_REQ

Table 15 ISC_TEST_REQ

Direction	HOST to S1V30120	
Purpose	This message registers the host with the S1V30120. The host registration sequence is mandatory.	
Byte	**Field**	**Values**
0	length (lsb)	0x000C
1	length (msb)	
2	msg_id (lsb)	ISC_TEST_REQ
3	msg_id (msb)	
4	enable_registration (lsb)	= 0x0000 : reserved = 0x0001 : register S1V30120 (other values are undefined)
5	enable_registration (msb)	
6	padding (lsb)	0x0000
7	padding (msb)	
8	reserved (lsb)	Reserved. Set to 0x00 00 00 00.
9	reserved	
10	reserved	
11	reserved (msb)	

5.5.3.2 ISC_TEST_RESP

Table 16 ISC_TEST_RESP

Direction	S1V30120 to HOST	
Purpose	Response to ISC_TEST_REQ. If received, indicates S1V30120 device is operating correctly and host registration was successful.	
Byte	**Field**	**Values**
0	length (lsb)	0x0006
1	length (msb)	
2	msg_id (lsb)	ISC_TEST_RESP
3	msg_id (msb)	
4	registration_success (lsb)	0x0000 : No Error >0x0000 : Error Code
5	registration_success (msb)	

5.5.3.3 ISC_VERSION_REQ

Table 17 ISC_VERSION_REQ

Direction	HOST to S1V30120	
Purpose	Used to request information about the hardware version, firmware version and supported firmware functionality	
Byte	**Field**	**Values**
0	length (lsb)	0x0004
1	length (msb)	
2	msg_id (lsb)	ISC_VERSION_REQ
3	msg_id (msb)	

5.5.3.4 ISC_VERSION_RESP

Table 18 ISC_VERSION_RESP

Direction	S1V30120 to HOST	
Purpose	Response to ISC_VERSION_REQ.	
Byte	**Field**	**Values**
0	length (lsb)	0x0014
1	length (msb)	
2	msg_id (lsb)	ISC_VERSION_RESP
3	msg_id (msb)	
4	hw_id_int	Hardware version identifier, integer part *NOTE 1
5	hw_id_frac	Hardware version identifier, fractional part
6	fw_version_x	Firmware version identifier, X part *NOTE 2
7	fw_version_y	Firmware version identifier, Y part
8	fw_features	Firmware Features: Bit Fields defined as follows: 0x00000001: TTS (Text to Speech) 0x00000002: Reserved 0x00000004: Reserved 0x00000008: Reserved 0x00000010: ADPCM 0x00000020: Reserved 0x00000040: Reserved 0x00000080: Reserved 0x00000100: GPIO 0x00000200: Reserved 0x00000400: Reserved 0x00000800: Reserved 0x00001000: Reserved 0x00002000: Reserved 0x10000000: Reserved
9		
10		
11		
12	fw_extended_features	TTS Language Support 0x00000001: US English 0x00000002: German 0x00000004: Castilian Spanish 0x00000008: French 0x00000010: UK English 0x00000020: Latin Spanish
13		
14		
15		
16	fw_version_z	Firmware version identifier, Z part
17	padding	
18	padding	
19	padding	

NOTE 1: For the hardware version, the integer part refers to the hardware platform, and the fractional part refers to the ROM revision number. For S1V30120 hardware, version 4.2 has been chosen.

NOTE 2: The firmware versions are defined as version X.Y.Z. where X is the major release number, Y is the minor release number (typically associated with a functionality enhancement), and Z is an incremental (bug fix) release.

5.5.3.5 ISC_ERROR_IND

Table 19 ISC_ERROR_IND

Direction	S1V30120 to HOST	
Purpose	Used to indicate to the host any S1V30120 errors that cannot be included into a RESPONSE. This error may be fatal or non fatal. See section 5.4 for an explanation of error code usage by S1V30120. In the case of a Fatal error all further messages sent to S1V30120 will cause the ISC_ERROR_IND to be resent.	
Byte	**Field**	**Values**
0	length (lsb)	0x0006
1	length (msb)	
2	msg_id (lsb)	ISC_ERROR_IND
3	msg_id (msb)	
4	error_code (lsb)	Error Code
5	error_code (msb)	

5.5.3.6 ISC_MSG_BLOCKED_RESP

Table 20 ISC_MSG_BLOCKED_RESP

Direction	S1V30120 to HOST	
Purpose	This message is sent in response to a request that has been blocked by the S1V30120 system controller. Request messages are blocked usually when the system controller identifies that there is insufficient system resource to service the request. (see section 5.5.2.2 for more details). All request messages are blocked if the interface has not been registered (using an ISC_TEST_REQ). An ISC_MSG_BLOCKED_RESP replaces the normally expected response message associated with the erroneous request.	
Byte	**Field**	**Values**
0	length (lsb)	0x0008
1	length (msb)	
2	msg_id (lsb)	ISC_MSG_BLOCKED_RESP
3	msg_id (msb)	
4	blocked_msg_id (lsb)	ISC message ID of the request message sent in error.
5	blocked_msg_id (lsb)	
6	error_code (lsb)	Error Code
7	error_code (msb)	

5.6 Power Management Messages

5.6.1 Overview

Most power management on the S1V30120 is performed automatically by the S1V30120 without intervention from the host.

However, to reach the lowest power mode, Standby Mode, the host has to request entry to this mode. During Standby Mode, the S1V30120 can only exit Standby mode by receiving a ISC_PMAN_STANDBY_EXIT_IND message

The ISC_PMAN_CONFIG_REQ message can be used to control the power management of the S1V30120.

5.6.2 Message Flows

5.6.2.1 Power Manager CONFIG

The power manager configuration message ISC_PMAN_CONFIG_REQ/RESP is used to configure the power management of the S1V30120. For the S1V30120, no options relating to power management configuration are available to the Host..

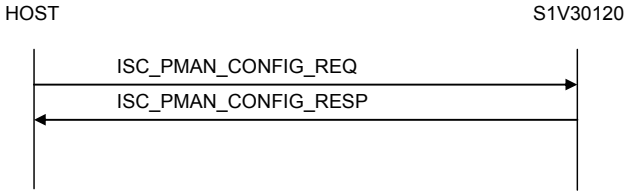
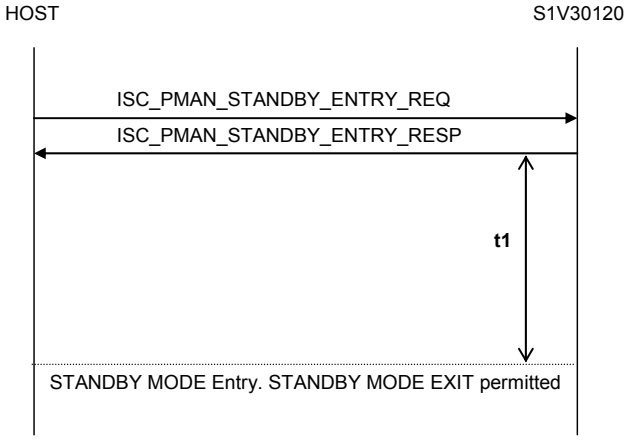


Figure 9 Power Manager CONFIG Message Flow

5.6.2.2 Power Manager STANDBY MODE ENTRY

Standby mode entry is initiated using the
ISC_PMAN_STANDBY_ENTRY_REQ/RESP Message Flow.



Symbol	Parameter	Min	Max	Unit
t1	Time from ISC_PMAN_STANDBY_ENTRY_RESP to Standby-Mode Entry	45	45	ms

Figure 10 Power Manager STANDBY MODE ENTRY Message Flow

5.6.2.3 Power Manager STANDBY MODE EXIT

In standby mode, the S1V30120 is able to correctly receive messages up to 4 bytes in length. To exit from standby mode, the host should send the 4 byte ISC_PMAN_STANDBY_EXIT_IND message to wake the S1V30120. This is the only message that must be used to wake the S1V30120 from standby. ISC_PMAN_EXIT_IND is the only message that should be sent to the host while it is in STANDBY. To ensure correct operation no padding should be sent following the ISC_PMAN_STANDBY_EXIT_IND message from HOST to S1V30120.

Once the S1V30120 has exited STANDBY mode an ISC_PMAN_STANDBY_EXIT_IND message will be received by the host. Padding should be sent as usual following the ISC_PMAN_STANDBY_EXIT_IND received by the HOST from the S1V30120.

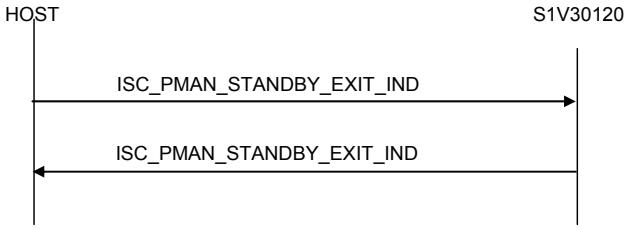


Figure 11 Power Manager STANDBY MODE EXIT Message Flow

5.6.3 Message Descriptions

5.6.3.1 ISC_PMAN_CONFIG_REQ

Table 21 ISC_PMAN_CONFIG_REQ

Direction	HOST to S1V30120	
Purpose	Used to configure the power manager.	
Byte	**Field**	**Values**
0	length (lsb)	0x0008
1	length (msb)	
2	msg_id (lsb)	ISC_PMAN_CONFIG_REQ
3	msg_id (msb)	
4	pman_audio_mode (lsb)	Reserved; set to 0x0001
5	pman_audio_mode (msb)	
6	pman_max_spi_clock_mhz	maximum SPI bit clock frequency used by the host (rounded up to 1MHz) *Set to 0x01.
7	padding	

5.6.3.2 ISC_PMAN_CONFIG_RESP

Table 22 ISC_PMAN_CONFIG_RESP

Direction	S1V30120 to HOST	
Purpose	Reply to ISC_PMAN_CONFIG_REQ. Indicates if request was successful.	
Byte	**Field**	**Values**
0	length (lsb)	0x0006
1	length (msb)	
2	msg_id (lsb)	ISC_PMAN_CONFIG_RESP
3	msg_id (msb)	
4	pman_config_success (lsb)	0x0000 : No Error
5	pman_config_success (msb)	>0x0000 : Error Code

5.6.3.3 ISC_PMAN_STANDBY_ENTRY_REQ

Table 23 ISC_PMAN_STANDBY_ENTRY_REQ

Direction	HOST to S1V30120	
Purpose	Used to request entry into standby mode. If S1V30120 is ready to enter standby, S1V30120 will enter standby mode after sending the ISC_PMAN_STANDBY_RESP message. If S1V30120 is not ready to enter standby mode, an ISC_PMAN_STANDBY_RESP message will be sent with a non-zero success field.	
Byte	**Field**	**Values**
0	length (lsb)	0x0004
1	length (msb)	
2	msg_id (lsb)	ISC_PMAN_STANDBY_ENTRY_REQ
3	msg_id (msb)	

5.6.3.4 ISC_PMAN_STANDBY_ENTRY_RESP

Table 24 ISC_PMAN_STANDBY_ENTRY_RESP

Direction	S1V30120 to HOST	
Purpose	Response to ISC_PMAN_STANDBY_ENTRY_REQ. Indicates if S1V30120 has entered standby mode.	
Byte	**Field**	**Values**
0	length (lsb)	0x0006
1	length (msb)	
2	msg_id (lsb)	ISC_PMAN_STANDBY_ENTRY_RESP
3	msg_id (msb)	
4	pman_standby_entry_success (lsb)	0x0000 : No Error >0x0000 : Error Code
5	pman_standby_entry_success (msb)	

5.6.3.5 ISC_PMAN_STANDBY_EXIT_IND

Table 25 ISC_PMAN_STANDBY_EXIT_IND

Direction	HOST to S1V30120 / S1V30120 to HOST	
Purpose	This message has two uses: Usage 1 (HOST to S1V30120): This message is sent by the Host to indicate that S1V30120 should exit standby mode. (NOTE 1) Usage 2 (S1V30120 to HOST): This message is sent by the S1V30120 when exiting standby mode.	
Byte	**Field**	**Values**
0	length (lsb)	0x0004
1	length (msb)	
2	msg_id (lsb)	ISC_PMAN_STANDBY_EXIT_IND
3	msg_id (msb)	

NOTE 1: The padding requirements after this message are different from those described in Section 3.4. After sending ISC_PMAN_STANDBY_EXIT_IND the host must not send any padding bytes.

5.7 Audio Messages

5.7.1 Overview

The Audio Configuration Messages are used for the following functions:

- Audio Gain, Volume and Sample Rate Configuration

Audio configuration is required before starting speech/audio playback.

Audio output may be started by audio configuration (ISC_AUDIO_CONFIG_REQ) or by the first data bearing message (ISC_SPCODEC_START_REQ or ISC_TTS_SPEAK_REQ). The selection of this is controlled by the ISC_AUDIO_CONFIG_REQ message. The reason for a Host choosing to start the audio with a ISC_AUDIO_CONFIG_REQ message is that it can reduce the delay time between the issuing of an ISC_SPCODEC_START_REQ or ISC_TTS_SPEAK_REQ and the output of audio. However, while audio is permanently on power saving is not possible and so more power is consumed.

5.7.2 Message Flows

5.7.2.1 Audio INITIALISE

Audio Output requires initialisation of audio sample rates and routing using the ISC_AUDIO_CONFIG_REQ/RESP messages.

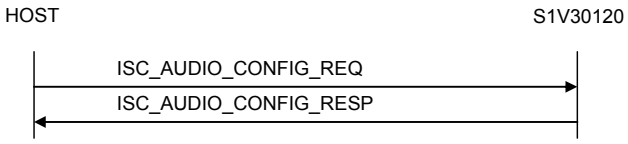


Figure 12 Audio INITIALISE Message Flow

5.7.2.2 Audio MUTE

Audio Output can be muted (set to all zeroes) using the ISC_AUDIO_MUTE_REQ/RESP Message Flow.

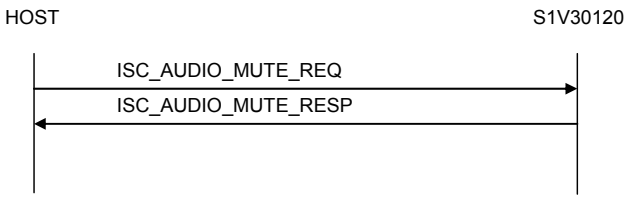


Figure 13 Audio MUTE Message Flow

5.7.2.3 ISC_AUDIO_CONFIG_REQ

Table 26 ISC_AUDIO_CONFIG_REQ

Direction	HOST to S1V30120	
Purpose	Used to configure the audio output.	
Byte	**Field**	**Values**
0	length (lsb)	0x000C
1	length (msb)	
2	msg_id (lsb)	ISC_AUDIO_CONFIG_REQ
3	msg_id (msb)	
4	audio_stereo	0x00 : Mono > 0x00 : Reserved
5	audio_gain NOTE5	0x00 : Mute 0x01 : -48dB 0x02 : -47dB . . 0x43 : +18dB
6	audio_amp	0x00 : Not Selected > 0x00 : Reserved
7	audio_sample_rate *NOTE1 & 2	0x00 : 8KHz 0x01 : 11.025kHz 0x02 : 12kHz (Reserved) 0x03 : 16kHz 0x04 : 22.05kHz (Reserved) 0x05 : 24kHz (Reserved) 0x06 : 32kHz (Reserved) 0x07 : 44.1kHz (Reserved) 0x08 : 48kHz (Reserved) 0x09 : DONT CARE (Set by Application)
8	audio_routing	0x00 : Application to DAC > 0x00 Reserved
9	audio_tone_control	DEPRECIATED, set to 0
10	audio_clock_source	0x00: Internally generated audio clock
11	DAC_permanently_on	0x00: DAC is on only while speech decoder or TTS synthesis is outputting audio. 0x01 : DAC is permanently on (NOTE3& 4)

NOTE 1: If desired, the DON'T CARE setting may be used for ADPCM file playback. In this case the sample rate used is the sample rate that was specified by the encoder and stored in the encoded ADPCM file header.

NOTE 2: 11.025kHz is reserved for use with TTS synthesis. All other sample rates are reserved for use with ADPCM.

NOTE 3: If DAC permanently on is set and minimum delay time is required before audio output after receiving a data bearing message, an initial sample rate must be selected. It must not be set to DON'T CARE. (Set by application)

NOTE 4: To enter standby mode, the DAC must be switched off by sending an ISC_AUDIO_DATA_REQ message with DAC_permanently_on set to 0x00.

NOTE 5: In order to avoid saturation for default configurations of the TTS engine, the audio_gain should not be set to a value greater than 0dB. The TTS engine and audio output hardware will by default utilize the full digital and analogue range. In order to avoid noticeable differences in output levels when synthesizing text and decoding ADPCM, ensure that the audio encoded by the ADPCM encoder utilizes the full scale or that audio level compensation is applied by the audio_gain setting.

5.7.2.4 ISC_AUDIO_CONFIG_RESP

Table 27 ISC_AUDIO_CONFIG_RESP

Direction	S1V30120 to HOST	
Purpose	Response message to ISC_AUDIO_CONFIG_REQ.	
Byte	**Field**	**Values**
0	length (lsb)	0x0006
1	length (msb)	
2	msg_id (lsb)	ISC_AUDIO_CONFIG_RESP
3	msg_id (msb)	
4	audio_config_success (lsb)	0x0000 : No Error >0x0000 : Error Code
5	audio_config_success (msb)	

5.7.2.5 ISC_AUDIO_VOLUME_REQ

Table 28 ISC_AUDIO_VOLUME_REQ

Direction	HOST to S1V30120	
Purpose	Set the output amplifier gain.	
Byte	**Field**	**Values**
0	length (lsb)	0x0006
1	length (msb)	
2	msg_id (lsb)	ISC_AUDIO_VOLUME_REQ
3	msg_id (msb)	
4	audio_gain_inc (lsb)	2's compliment value Gain increment/decrement in dB
5	audio_gain_inc (msb)	

5.7.2.6 ISC_AUDIO_VOLUME_RESP

Table 29 ISC_AUDIO_VOLUME_RESP

Direction	S1V30120 to HOST	
Purpose	Response to ISC_AUDIO_VOLUME_REQ	
Byte	**Field**	**Values**
0	length (lsb)	0x0006
1	length (msb)	
2	msg_id (lsb)	ISC_AUDIO_VOLUME_RESP
3	msg_id (msb)	
4	audio_gain_success (lsb)	0x0000 : No Error >0x0000 : Error Code
5	audio_gain_success (msb)	

5.7.2.7 ISC_AUDIO_MUTE_REQ

Table 30 ISC_AUDIO_MUTE_REQ

Direction	HOST to S1V30120	
Purpose	Used to digitally mute the audio output	
Byte	**Field**	**Values**
0	length (lsb)	0x0006
1	length (msb)	
2	msg_id (lsb)	ISC_AUDIO_MUTE_REQ
3	msg_id (msb)	
4	audio_mute_enable (lsb)	0x0000 : disable mute 0x0001 : enable mute
5	audio_mute_enable (msb)	

5.7.2.8 ISC_AUDIO_MUTE_RESP

Table 31 ISC_AUDIO_MUTE_RESP

Direction	S1V30120 to HOST	
Purpose	Response to ISC_AUDIO_MUTE_REQ	
Byte	**Field**	**Values**
0	length (lsb)	0x006
1	length (msb)	
2	msg_id (lsb)	ISC_AUDIO_MUTE_RESP
3	msg_id (msb)	
4	audio_mute_success (lsb)	0x0000 : No Error >0x0000 : Error Code
5	audio_mute_success (msb)	

5.8 TTS Messages

5.8.1 Overview

The S1V30120 integrates a version of the Fonix DECtalk® 5.1 TTS kernel, from Fonix Corporation that has been optimized for deployment in an embedded system.

5.8.1.1 Primary Pronunciation Dictionaries

The main dictionaries are constructed from a slightly smaller subset of the dictionaries delivered with the standard Fonix DECtalk®. This reduces the overall memory footprint.

5.8.1.2 User Dictionaries

The User Dictionaries are formatted differently from the standard DECtalk® dictionaries (to reduce the memory footprint). User Dictionaries are stored in the S1V30120. internal RAM memory. The host can transfer a user dictionary to the S1V30120 using ISC_TTS_UDICT_DATA_REQ/RESP messages. For the location of the tool that should be used to generate a user dictionary see S1V30120 Evaluation Kit User Guide.

5.8.1.3 Text Format

The TTS accepts text encoded in ISO 8859 Latin1 character set. All character strings passed to the TTS using the ISC_TTS_SPEAK_REQ message must be terminated with the null character (‘\0’) and the text in the buffer should terminate on a clause or sentence boundary as indicated by full stop ‘.’ or comma ‘,’ punctuation.

5.8.1.4 EPSON Parser

Setting the tts_epson_parse field in the ISC_TTS_CONFIG_REQ message enables a parsing function that is run before the TTS engine. The parser adds support for a range of text message style icons & emoticons which can be used to control the TTS.

The set of supported icons is as follows:

Text	Function
\/	Decrease Pitch
/\	Increase Pitch
>>	Increase Speaking Rate
<<	Decrease Speaking Rate
__	Emphasise next word
##	Whisper next word
:-)*X*	Select voice X

Figure 14 EPSON Parser icon support

5.8.1.5 Speaker Definitions

Speaker definition parameters are managed outside of the Fonix DECalk® TTS kernel. For users familiar with the DECtalk® command set, this has an impact on the operation of the [:dv] and [:name] inline commands (see notes in the following section).

The standard Fonix DECtalk® 5.0 voices are mapped on to the tts_voice field in the ISC_TTS_CONFIG_REQ as follows:

Voice Index	Fonix DECtalk® name
0	Paul
1	Harry
2	Reserved
3	Reserved
4	Dennis
5	Reserved
6	Reserved
7	Reserved
8	Wendy
9	Reserved

Figure 15 DECtalk® voice index to name mapping

5.8.1.6 DECtalk® Inline Command Support

The following table summarises the DECtalk® inline commands supported by the S1V30120.

These commands are only available if the EPSON parser is disabled (using the ISC_TTS_CONFIG_REQ message).

Inline Command	Supported	Notes
[:comma]	YES	
[:dv]	YES	save option is not supported.
[:dial]	NO	
[:error]	NO	
[:index mark]	NO	
[:log]	NO	
[:mode]	YES	email parsing is not supported
[:name]	YES	speaker definition parameters are managed outside of DECtalk, syntax now uses a number rather than a name, e.g. [:name 0] instead of e.g. [:name paul]
[:period]	YES	
[:phoneme]	YES	
[:pitch]	YES	
[:play]	NO	
[:pronounce]	YES	
[:punct]	YES	
[:rate]	YES	
[:say]	YES	
[:skip]	YES	
[:sync]	YES	
[:tone]	NO	
[:volume]	NO	volume control is handled outside the DECtalk(R) kernel

Figure 16 DECtalk® inline command summary

5.8.1.7 Audio Output Sample Format

Linear PCM (16bit) at 11.025kHz is supported. A-law and u-law is not supported.

5.8.2 Message Flows

5.8.2.1 TTS CONFIGURE

The ISC_TTS_CONFIGURE _REQ/RESP message is used to configure the TTS.

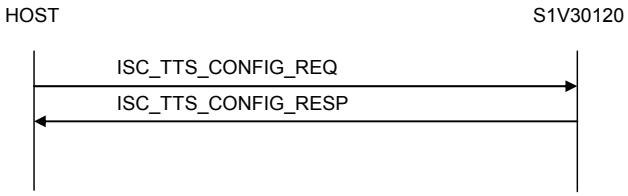


Figure 17 TTS CONFIGURE Message Flow

5.8.2.2 TTS START

The ISC_TTS_SPEAK_REQ/RESP message is used to send text data to the S1V30120. The ISC_TTS_READY_IND is used to indicate that the S1V30120 is ready for more data (the user may ignore it if at the end of the text stream.

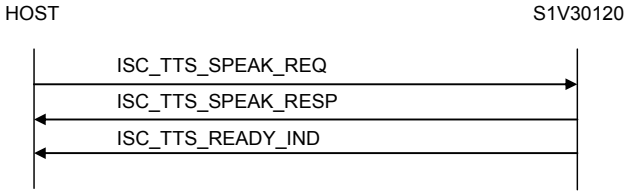


Figure 18 TTS SPEAK Message Flow

Depending on the amount of text in the internal TTS buffers it is possible for the ISC_TTS_READY_IND to be sent before the ISC_TTS_SPEAK_RESP.

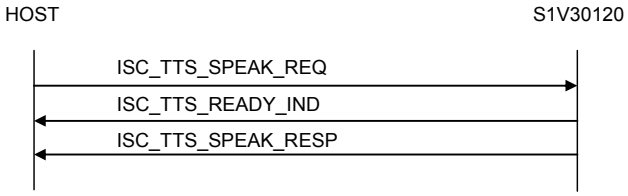


Figure 19 TTS SPEAK Message Flow (alternative)

5.8.2.3 TTS FINALISE IMMEDIATELY

The ISC_TTS_STOP_REQ/RESP message is used to terminate the synthesis of speech immediately. Text already synthesised to audio will be played before the ISC_TTS_STOP_RESP is output.

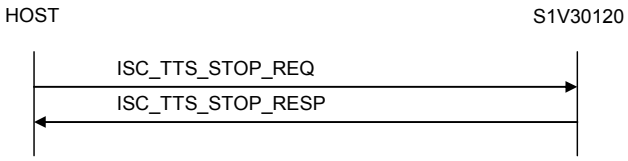


Figure 20 TTS FINALISE IMMEDIATELY Message Flow

5.8.2.4 TTS PAUSE

On issuing the ISC_TTS_PAUSE_REQ/RESP message with the pause flag set, the speech synthesiser will stop processing text. Time must be allowed for the audio output buffers to drain before the pause can take effect.

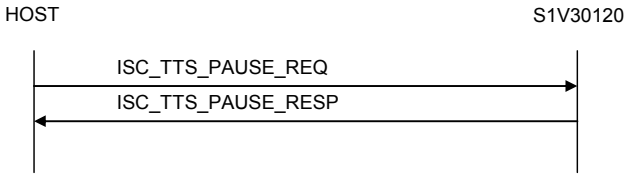


Figure 21 TTS PAUSE Message Flow

5.8.2.5 TTS UNPAUSE

On issuing the ISC_TTS_PAUSE_REQ/RESP message with the pause flag cleared, the tts synthesiser will restart processing text.

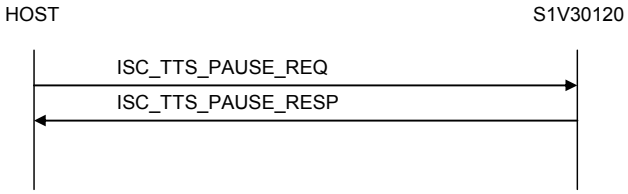


Figure 22 TTS UNPAUSE Message Flow

5.8.2.6 TTS FINALISE AUTOMATICALLY

S1V30120 automatically finalises TTS after completing all input text synthesis and sends an ISC_TTS_FINISHED_IND to indicate that this has been successfully completed. The end of the text stream is identified by the NULL terminator at the end of the submitted text. After this it is necessary to send an ISC_TTS_STOP_REQ in order to free the system resources associated with TTS.

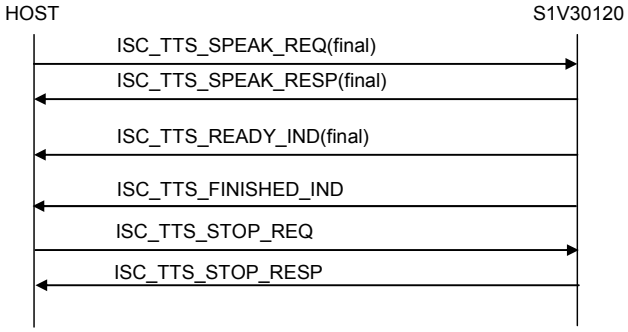


Figure 23 TTS FINALISE AUTOMATICALLY Message Flow

5.8.2.7 TTS UDICT DATA

The ISC_TTS_UDICT_DATA_REQ/RESP message is used to transfer user dictionary data to the S1V30120. The user dictionary data must be small enough to be transferred in a single ISC_TTS_UDICT_DATA_REQ message. Only one user dictionary can be stored in the S1V30120 at any one time. The user dictionary loaded to the S1V30120 can have a maximum size of 2048 bytes. The TTS UDICT DATA message flow should be followed by a TTS CONFIGURE message flow to ensure that the transferred user dictionary data is taken in to use by the TTS engine.

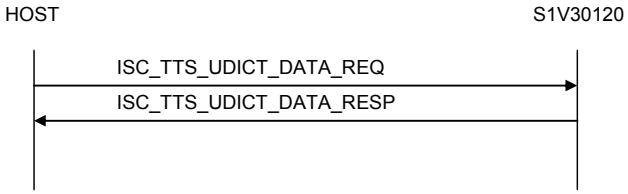


Figure 1 TTS UDICT DATA Message Flow

5.8.2.7 Example: Streaming TTS Synthesis

The sequence diagram below shows an example of speech synthesis. Speech output has terminated when the ISC_TTS_FINISHED_IND is received.

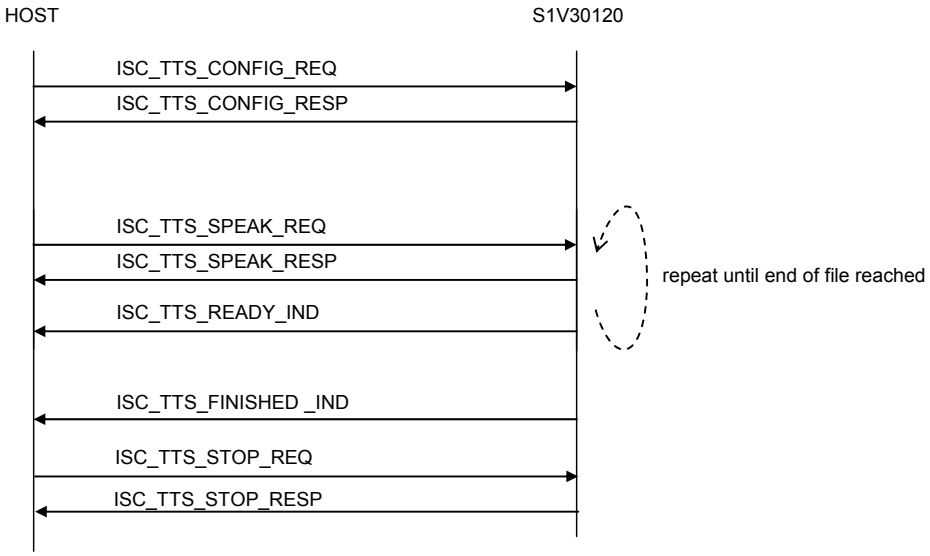


Figure 24 Example Streaming TTS Synthesis Flow

5.8.3 Message Descriptions

5.8.3.1 ISC_TTS_CONFIG_REQ

Table 32 ISC_TTS_CONFIG_REQ

Direction	HOST to S1V30120	
Purpose	Used to request that the TTS system be configured. Can only be used when the TTS system is inactive. A subsequent ISC_TTS_SPEAK_REQ message is required to start outputting speech.	
Byte	**Field**	**Values**
0	length (lsb)	0x000C
1	length (msb)	
2	msg_id (lsb)	ISC_TTS_CONFIG_REQ
3	msg_id (msb)	
4	tts_sample_rate	0x00 : reserved 0x01 : 11.025kHz
5	tts_voice	0x00 : Voice 0 0x01 : Voice 1 . . 0x08 : Voice 9 (custom voice)
6	tts_epson_parse	0x00 : Disable 0x01 : Enable
7	tts_language	0x00 : US English 0x01 : Castilian Spanish 0x02 : reserved 0x03 : reserved 0x04 : Latin Spanish
8	tts_speaking_rate (lsb)	Speaking rate in words per minute The valid range is 0x004B to 0x0258. A suitable default value is 200 words/min (0x00C8)
9	tts_speaking_rate (msb)	
10	tts_datasource	Set to 0x00
11	Reserved	Set to 0x00

5.8.3.2 ISC_TTS_CONFIG_RESP

Table 33 ISC_TTS_CONFIG_RESP

Direction	S1V30120 to HOST	
Purpose	Response to ISC_TTS_CONFIG_REQ, indicates success or failure to configure the TTS	
Byte	**Field**	**Values**
0	length (lsb)	0x0006
1	length (msb)	
2	msg_id (lsb)	ISC_TTS_CONFIG_RESP
3	msg_id (msb)	
4	tts_config_success (lsb)	0x0000 : No Error >0x0000 : Error Code
5	tts_config_success (msb)	

5.8.3.3 ISC_TTS_SPEAK_REQ

Table 34 ISC_TTS_SPEAK_REQ

Direction	HOST to S1V30120	
Purpose	Used to request that the TTS engine start converting text into speech. The text data appended to this message is synthesised to speech.	
Byte	**Field**	**Values**
0	length (lsb)	0x0006 or VARIABLE if text is appended.
1	length (msb)	
2	msg_id (lsb)	ISC_TTS_SPEAK_REQ
3	msg_id (msb)	
4	flush_enable	=0x00 – flush disabled – this string is spoken after any previous speak requests have been finished. =0x01 – flush enabled – this TTS output is flushed and the current string is spoken immediately (other values are undefined)
5	tts_text_buffer	Null terminated character string
...		
variable		0x00

5.8.3.4 ISC_TTS_SPEAK_RESP

Table 35 ISC_TTS_SPEAK_RESP

Direction	S1V30120 to HOST	
Purpose	Response to ISC_TTS_SPEAK_REQ, indicates that the request was successful and that the S1V30120 is now converting text into speech	
Byte	**Field**	**Values**
0	length (lsb)	0x0006
1	length (msb)	
2	msg_id (lsb)	ISC_TTS_SPEAK_RESP
3	msg_id (msb)	
4	tts_speak_success (lsb)	0x0000 : No Error >0x0000 : Error Code
5	tts_speak_success (msb)	

5.8.3.5 ISC_TTS_READY_IND

Table 36 ISC_TTS_READY_IND

Direction	S1V30120 to HOST	
Purpose	This message indicates that the system is ready to receive another ISC_TTS_SPEAK_REQ message.	
Byte	**Field**	**Values**
0	length (lsb)	0x0004
1	length (msb)	
2	msg_id (lsb)	ISC_TTS_READY_IND
3	msg_id (msb)	

5.8.3.6 ISC_TTS_FINISHED_IND

Table 37 ISC_TTS_FINISHED_IND

Direction	S1V30120 to HOST	
Purpose	Indicates that the system has finished speaking all of the null terminated text that has been submitted.	
Byte	**Field**	**Values**
0	length (lsb)	0x0004
1	length (msb)	
2	msg_id (lsb)	ISC_TTS_FINISHED_IND
3	msg_id (msb)	

5.8.3.7 ISC_TTS_PAUSE_REQ

Table 38 ISC_TTS_PAUSE_REQ

Direction	HOST to S1V30120	
Purpose	Used to request that the TTS pause the TTS output.	
Byte	**Field**	**Values**
0	length (lsb)	0x0006
1	length (msb)	
2	msg_id (lsb)	ISC_TTS_PAUSE_REQ
3	msg_id (msb)	
4	tts_pause_enable (lsb)	0x0000 : disable pause 0x0001 : enable pause
5	tts_pause_enable (msb)	

5.8.3.8 ISC_TTS_PAUSE_RESP

Table 39 ISC_TTS_PAUSE_RESP

Direction	S1V30120 to HOST	
Purpose	Response to ISC_TTS_PAUSE_REQ	
Byte	**Field**	**Values**
0	length (lsb)	0x0006
1	length (msb)	
2	msg_id (lsb)	ISC_TTS_PAUSE_RESP
3	msg_id (msb)	
4	tts_pause_success (lsb)	0x0000 : No Error >0x0000 : Error Code
5	tts_pause_success (msb)	

5.8.3.9 ISC_TTS_STOP_REQ

Table 40 ISC_TTS_STOP_REQ

Direction	HOST to S1V30120	
Purpose	Used to request that the current TTS processing is stopped immediately. If the TTS is reset, system resources used by the TTS are freed, and all configuration data is lost (the TTS is finalised).	
Byte	**Field**	**Values**
0	length (lsb)	0x0006
1	length (msb)	
2	msg_id (lsb)	ISC_TTS_STOP_REQ
3	msg_id (msb)	
4	tts_reset_tts (lsb)	0x0000 : do not reset TTS 0x0001 : reset TTS (finalise TTS)
5	tts_reset_tts (msb)	

5.8.3.10 ISC_TTS_STOP_RESP

Table 41 ISC_TTS_STOP_RESP

Direction	S1V30120 to HOST	
Purpose	Response to ISC_TTS_STOP_REQ. Indicates if request was successful.	
Byte	**Field**	**Values**
0	length (lsb)	0x0006
1	length (msb)	
2	msg_id (lsb)	ISC_TTS_STOP_RESP
3	msg_id (msb)	
4	tts_stop_success (lsb)	0x0000 : No Error >0x0000 : Error Code
5	tts_stop_success (msb)	

5.8.3.11 ISC_TTS_UDICT_DATA_REQ

Table 1 ISC_TTS_UDICT_DATA_REQ

Direction	HOST to S1V30120	
Purpose	Used to send user dictionary data. User dictionary data is transferred in a single message. The amount of data that may be appended is limited to 2048 bytes. The ISC_TTS_UDICT_DATA_REQ message may be sent without any data, with the tts_clear_udict field set to 0x0001 to erase any previously transferred user dictionary data.	
Byte	**Field**	**Values**
0	length (lsb)	0x0008 or VARIABLE if data is appended.
1	length (msb)	
2	msg_id (lsb)	ISC_TTS_UDICT_DATA_REQ
3	msg_id (msb)	
4	tts_clear_udict (lsb)	0x0000 : do not erase existing data 0x0001 : erase all existing data
5	tts_clear_udict (msb)	
6	tts_udict_data_length (lsb)	number of bytes of user dictionary data appended to the message
7	tts_udict_data_length (msb)	
…	tts_udict_data	user dictionary data

5.8.3.12 ISC_TTS_UDICT_DATA_RESP

Table 2 ISC_TTS_UDICT_DATA_RESP

Direction	S1V30120 to HOST	
Purpose	Response to ISC_TTS_UDICT_DATA_REQ. Indicates if request was successful.	
Byte	**Field**	**Values**
0	length (lsb)	0x0006
1	length (msb)	
2	msg_id (lsb)	ISC_TTS_UDICT_DATA_RESP
3	msg_id (msb)	
4	tts_udict_data_success (lsb)	0x0000 : No Error >0x0000 : Error Code
5	tts_udict_data_success (msb)	

5.9 Speech Codec Messages (ADPCM)

5.9.1 Overview

The Speech Codec application supports Voice Playback (Decode) using an ADPCM codec.

5.9.1.1 Codec Formats

ADPCM (ITU G.726 compatible), 8kHz sampling at 40kbps, 32kbps and 24kbps.

ADPCM, 16kHz sampling at 64kbps and 48kbps.

5.9.1.2 File Format

An EPSON specific file format is used for storing the compressed speech files. The tool used for the creation of these files is common_alg_file_client.exe which is included with the S1V30120 documentation. Usage instructions for this tool are included within the tool's help option. To access these instructions, invoke common_alg_file_client.exe with no arguments.

5.9.2 Message Flows

5.9.2.1 Speech Codec CONFIGURE

The ISC_SPCODEC_CONFIG_REQ/RESP message is used to configure the codec.

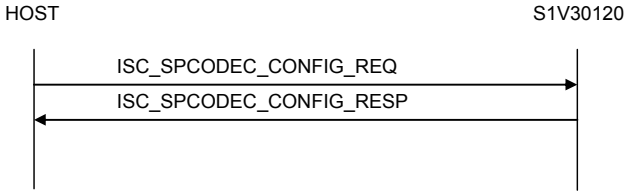


Figure 25 Speech Codec CONFIGURE Message Flow

5.9.2.2 Speech Decode START

The ISC_SPCODEC_START_REQ/RESP message is used to transfer data to the speech decoder. The ISC_SPCODEC_READY_IND is used to indicate whether the S1V30120 can accept further data.

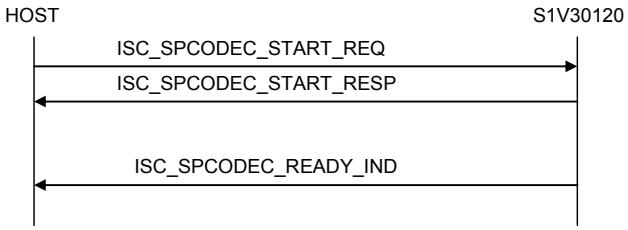


Figure 26 Speech Codec START Message Flow (RESPonse before INDication)

Depending upon the amount of compressed data in the S1V30120 internal buffer, the ISC_SPCODEC_READY_IND may be sent before the ISC_SPCODEC_START_RESP.

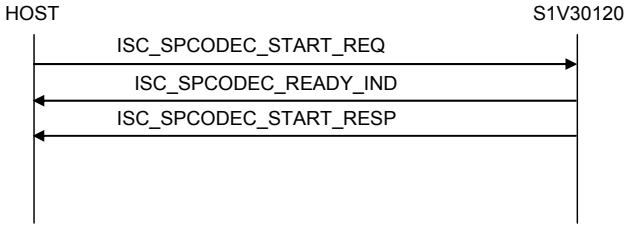


Figure 27 Speech Codec START Message Flow (INDication before RESPonse)

5.9.2.3 Speech Codec FINALISE IMMEDIATELY

The ISC_SPCODEC_STOP_REQ/RESP message is used to terminate the decode of speech data immediately. Audio data already decoded will be played before termination.

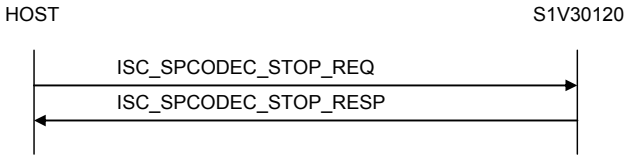


Figure 28 Speech Codec FINALISE IMMEDIATELY Message Flow

5.9.2.4 Speech Codec PAUSE

On issuing the ISC_SPCODEC_PAUSE_REQ/RESP message with the pause flag set, the speech decoders will stop processing speech data.

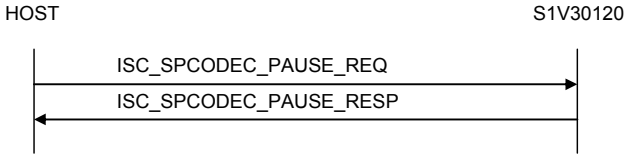


Figure 29 Speech Codec PAUSE Message Flow

5.9.2.5 Speech Codec UNPAUSE

On issuing the ISC_SPCODEC_PAUSE_REQ/RESP message with the pause flag cleared, the speech codec will restart processing of speech data.

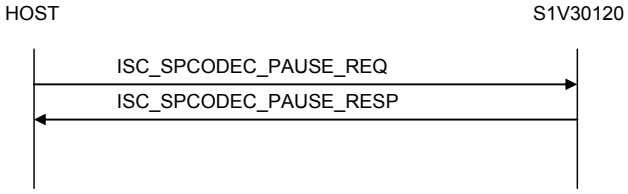


Figure 30 Speech Codec UNPAUSE Message Flow

5.9.2.6 Speech Codec FINALISE AUTOMATICALLY

S1V30120 automatically finalizes Speech decoding after decoding all ADPCM data contained in the source file. After completing finalisation, the S1V30120 will send the ISC_SPCODEC_FINISHED_IND.

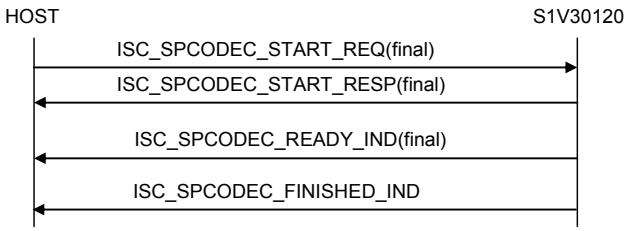


Figure 31

5.9.2.7 Example: Streaming Speech Decode

The sequence diagram below shows an example of speech decode using streaming across the host interface. Speech output has terminated when the ISC_SPCODEC_FINISHED_IND is received.

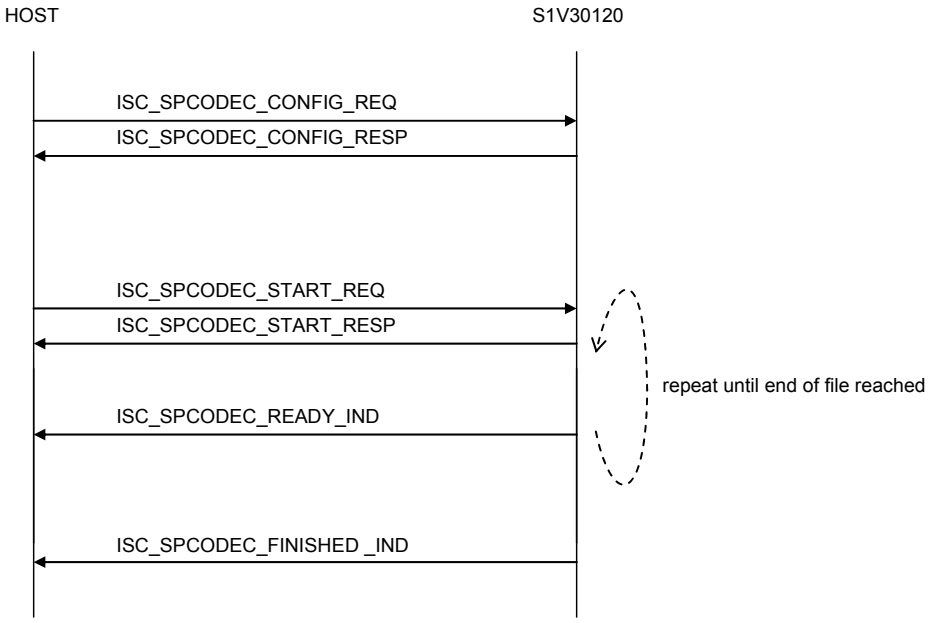


Figure 32 Example Streaming Speech Decode Flow

5.9.3 Message Descriptions

5.9.3.1 ISC_SPCODEC_CONFIG_REQ

Table 42 ISC_SPCODEC_CONFIG _REQ

Direction	HOST to S1V30120	
Purpose	Used to request that the SPCODEC be configured. Can only be used when the SPCODEC is inactive. A subsequent ISC_SPCODEC_START_REQ message is required to start the playback/record process.	
Byte	**Field**	**Values**
0	length (lsb)	0x024
1	length (msb)	
2	msg_id (lsb)	ISC_SPCODEC_CONFIG_REQ
3	msg_id (msb)	
4	datasource	0x00 : Reserved 0x01 : SPI IF
5	codec_config	0x00 : Reserved 0x01 : Decode (Playback)
6	padding	
7	padding	
8 to 27	reserved	Set all fields to 0x00
28	spcodec_type (lsb)	0x00000000 : Reserved 0x00000001 : Reserved 0x00000002 : Unspecified (use for Decode) > 0x00000002 : Reserved
29	spcodec_type	
30	spcodec_type	
31	spcodec_type (msb)	
32	spcodec_rate (lsb)	Reserved, set to 0x00000000
33	spcodec_rate	
34	spcodec_rate	
35	spcodec_rate (msb)	

5.9.3.2 ISC_SPCODEC_CONFIG_RESP

Table 43 ISC_SPCODEC_CONFIG_RESP

Direction	S1V30120 to HOST	
Purpose	Reply to ISC_SPCODEC_CONFIG_REQ. Indicates if request was successful.	
Byte	**Field**	**Values**
0	length (lsb)	0x0006
1	length (msb)	
2	msg_id (lsb)	ISC_SPCODEC_CONFIG_RESP
3	msg_id (msb)	
4	common_config_success (lsb)	0x0000 : No Error >0x0000 : Error Code
5	common_config_success (msb)	

5.9.3.3 ISC_SPCODEC_START _REQ

Table 44 ISC_SPCODEC_START_REQ

Direction	HOST to S1V30120	
Purpose	Used to request that the SPCODEC start decoding audio data. Data appended to this message is decoded.	
Byte	**Field**	**Values**
0	length (lsb)	VARIABLE
1	length (msb)	
2	msg_id (lsb)	ISC_SPCODEC_START_REQ
3	msg_id (msb)	
…	data	Compressed audio data

NOTE 1: The amount of compressed data appended to ISC_SPCODEC_START_REQ should be either 512, 1024 or 2048 bytes. This corresponds to valid message lengths of 516, 1028 and 2052 bytes. The last SPCODEC_START_REQ for the file should be the remainder. The only constraint on this remainder value is that it be less than 2052 bytes.

5.9.3.4 ISC_SPCODEC_START_RESP

Table 45 ISC_SPCODEC_START_RESP

Direction	S1V30120 to HOST	
Purpose	Reply to ISC_SPCODEC_START_REQ. Indicates if request was successful.	
Byte	**Field**	**Values**
0	length (lsb)	0x000A
1	length (msb)	
2	msg_id (lsb)	ISC_SPCODEC_START_RESP
3	msg_id (msb)	
4	common_decode_success (lsb)	0x0000 : No Error >0x0000 : Error Code
5	common_decode_success (msb)	
6	length (lsb)	Reserved
7	length (msb)	
8	data	Reserved
9	data	Reserved

5.9.3.5 ISC_SPCODEC_PAUSE_REQ

Table 46 ISC_SPCODEC_PAUSE_REQ

Direction	HOST to S1V30120	
Purpose	Used to request a pause in the SPCODEC output.	
Byte	**Field**	**Values**
0	length (lsb)	0x0006
1	length (msb)	
2	msg_id (lsb)	ISC_SPCODEC_PAUSE_REQ
3	msg_id (msb)	
4	common_pause_enable (lsb)	0x0000 : disable pause 0x0001 : enable pause
5	common_pause_disable (msb)	

5.9.3.6 ISC_SPCODEC_PAUSE_RESP

Table 47 ISC_SPCODEC_PAUSE_RESP

Direction	S1V30120 to HOST	
Purpose	Reply to ISC_SPCODEC_PAUSE_REQ. Indicates if request was successful.	
Byte	**Field**	**Values**
0	length (lsb)	0x0006
1	length (msb)	
2	msg_id (lsb)	ISC_SPCODEC_PAUSE_RESP
3	msg_id (msb)	
4	common_pause_success (lsb)	0x0000 : No Error >0x0000 : Error Code
5	common_pause_success (msb)	

5.9.3.7 ISC_SPCODEC_STOP_REQ

Table 48 ISC_SPCODEC_STOP_REQ

Direction	HOST to S1V30120	
Purpose	Used to request that the current SPCODEC processing is stopped immediately. If the SPCODEC is reset, system resources used by the SPCODEC are freed, and all configuration data is lost (the SPCODEC is finalised).	
Byte	**Field**	**Values**
0	length (lsb)	0x0006
1	length (msb)	
2	msg_id (lsb)	ISC_SPCODEC_STOP_REQ
3	msg_id (msb)	
4	reset_algorithm (lsb)	0x0000 : do not reset SPCODEC 0x0001 : reset SPCODEC
5	reset_algorithm (msb)	

5.9.3.8 ISC_SPCODEC_STOP_RESP

Table 49 ISC_SPCODEC_STOP_RESP

Direction	S1V30120 to HOST	
Purpose	Reply to ISC_SPCODEC_STOP_REQ. Indicates if request was successful.	
Byte	**Field**	**Values**
0	length (lsb)	0x0006
1	length (msb)	
2	msg_id (lsb)	ISC_SPCODEC_STOP_RESP
3	msg_id (msb)	
4	common_stop_success (lsb)	0x0000 : No Error >0x0000 : Error Code
5	common_stop_success (msb)	

5.9.3.9 ISC_SPCODEC_READY_IND

Table 50 ISC_SPCODEC_READY_IND

Direction	S1V30120 to HOST	
Purpose	If the SPCODEC is configured to accept data over the Active I/F, this message indicates that the system is ready to receive another ISC_SPCODEC_START_REQ message.	
Byte	**Field**	**Values**
0	length (lsb)	0x0004
1	length (msb)	
2	msg_id (lsb)	ISC_SPCODEC_READY_IND
3	msg_id (msb)	

5.9.3.10 ISC_SPCODEC_FINISHED_IND

Table 51 ISC_SPCODEC_FINISHED_IND

Direction	S1V30120 to HOST	
Purpose	Indicates that the system has finished decoding all of the submitted data.	
Byte	**Field**	**Values**
0	length (lsb)	0x0004
1	length (msb)	
2	msg_id (lsb)	ISC_SPCODEC_FINISHED_IND
3	msg_id (msb)	

5.10 GPIO Messages

5.10.1 Overview

The S1V30120 includes messages to support GPIO Output control. GPIOA10-5 can be controlled by this message.

5.10.1.1 Registration and Control Messages

To use the GPIO Messages, the host must first be registered for control of the GPIO pins using the ISC_GPIO_REGISTER_REQ message.

5.10.1.2 GPIO Output Configuration and Control

After GPIO registration the pins that are to be used must be configured as outputs by the ISC_GPIO_OUTPUT_CONFIG_REQ message. Once configured, outputs can be switched between high and low levels using the ISC_GPIO_OUTPUT_SET_REQ message.

5.10.2 Message Flows

The sequence diagram below shows an example to set a GPIOA output value.

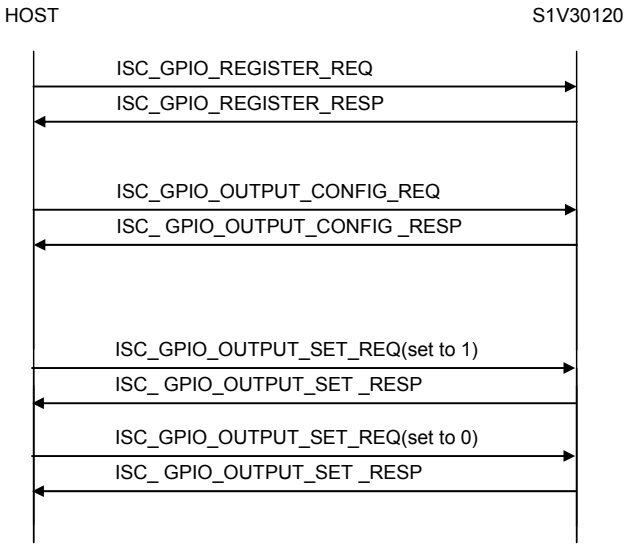


Figure 33 Example GPIO Set Output Flow

5.10.3 GPIO Transition Latency

The time from the sending of the last byte of a GPIO_OUTPUT_SET_REQ message to the corresponding change in signal level on the GPIO pin is guaranteed to be less than 1ms.

5.10.3 Message Descriptions

5.10.3.1 ISC_GPIO_REGISTER_REQ

Table 52 ISC_GPIO_REGISTER_REQ

Direction	HOST to S1V30120	
Purpose	Used by host to register or de-register for control of GPIO.	
Byte	**Field**	**Values**
0	length (lsb)	0x0006
1	length (msb)	
2	msg_id (lsb)	ISC_GPIO_REGISTER_REQ
3	msg_id (msb)	
4	enable_registration (lsb)	0x0000 : de-register for control of GPIO interface 0x0001 : register for control of GPIO interface
5	enable_registration (msb)	

5.10.3.2 ISC_GPIO_REGISTER_RESP

Table 53 ISC_GPIO_REGISTER_RESP

Direction	S1V30120 to HOST	
Purpose	Response to ISC_GPIO_REGISTER_RESP.	
Byte	**Field**	**Values**
0	length (lsb)	0x0006
1	length (msb)	
2	msg_id (lsb)	ISC_GPIO_REGISTER_RESP
3	msg_id (msb)	
4	registration_success (lsb)	0x0000 : No Error >0x0000 : Error Code
5	registration_success (msb)	

5.10.3.3 ISC_GPIO_OUTPUT_CONFIG_REQ

Table 54 ISC_GPIO_OUTPUT_CONFIG_REQ

Direction	HOST to S1V30120	
Purpose	Configures GPIO setting.	
Byte	**Field**	**Values**
0	length (lsb)	0x0010
1	length (msb)	
2	msg_id (lsb)	ISC_GPIO_OUTPUT_CONFIG_REQ
3	msg_id (msb)	
4	Reserved	Set to 0x00
5	Reserved	Set to 0x00
6	Reserved	Set to 0x00
7	Reserved	Set to 0x00
8	Reserved	Set to 0x00
9	output_config[5] (GPIO 5)	Set to 0x03
10	output_config[6] (GPIO 6)	Set to 0x03
11	output_config[7] (GPIO 7)	Set to 0x03
12	output_config[8] (GPIO 8)	Set to 0x03
13	output_config[9] (GPIO 9)	Set to 0x03
14	output_config[10] (GPIO 10)	Set to 0x03
15	Reserved	Set to 0x00

5.10.3.4 ISC_GPIO_OUTPUT_CONFIG_RESP

Table 55 ISC_GPIO_OUTPUT_CONFIG_RESP

Direction	S1V30120 to HOST	
Purpose	Reply to ISC_GPIO_OUTPUT_CONFIG_RESP.	
Byte	**Field**	**Values**
0	length (lsb)	0x0004
1	length (msb)	
2	msg_id (lsb)	ISC_GPIO_OUTPUT_CONFIG_RESP
3	msg_id (msb)	

5.10.3.5 ISC_GPIO_OUTPUT_SET_REQ

Table 56 ISC_GPIO_OUTPUT_SET_REQ

Direction	HOST to S1V30120	
Purpose	Sets the value of GPIO output.	
Byte	**Field**	**Values**
0	length (lsb)	0x000C
1	length (msb)	
2	msg_id (lsb)	ISC_GPIO_OUTPUT_SET_REQ
3	msg_id (msb)	
4	output_value (lsb)	Bit[0] : set to 0 Bit[1] : set to 0 Bit[2] : set to 0 Bit[3] : set to 0 Bit[4] : set to 0 Bit[5] : set GPIOA5:Low=0x00 High=0x01 Bit[6] : set GPIOA6 Low=0x00 High=0x01 Bit[7] : set GPIOA7 Low=0x00 High=0x01 Bit[8] : set GPIOA8 Low=0x00 High=0x01 Bit[9] : set GPIOA9 Low=0x00 High=0x01 Bit[10] : set GPIOA10 Low=0x00 High=0x01 Bit[11] : set to 0
5	output_value	
6	output_value	
7	output_value (msb)	
8	output_disable (lsb)	Set to 0x0000_0000
9	output_disable	
10	output_disable	
11	output_disable (msb)	

5.10.3.6 ISC_GPIO_OUTPUT_SET_RESP

Table 57 ISC_GPIO_OUTPUT_SET_RESP

Direction	S1V30120 to HOST	
Purpose	Reply to GPIO_OUTPUT_SET_RESP.	
Byte	**Field**	**Values**
0	length (lsb)	0x0004
1	length (msb)	
2	msg_id (lsb)	ISC_GPIO_OUTPUT_SET_RESP
3	msg_id (msb)	

Revision History

Date	Revision details			
	Rev.	Page	Type	Details
06/22/2007	1.10			Added GPIOA10-5 control function. Modified to finalise automatically for both ADPCM and TTS. ISC_SPCODEC_CLOSE/ISC_TTS_CLOSE are deleted. Modified to exit from standby without using GPIOA4. Modified the description to exit from standby by command. Deleted SPDEFS function. Modified MSG_READY explanation. Added DAC permanently on function. Added explanation to go to standby, DAC must be off, if DAC permanently on is set. Modified ISC_VERSION_RESP field. Changed GPIOA4 setting value to “Low”(from High).
07/12/2007	1.11			Deleted SPDEFS in Voice Index TTS INPUT buffer maximum size
07/18/2007	1.12			Added SRAM_vectors Deleted Related C source code
10/26/2007	1.13			Major revision of document. Restructured and rewritten to present information as the customer should see it rather than from the developers point of view.
11/30/2007	1.14			Added notes on GPIO response times and audio gain settings. Tidied up GPIO description. Tidied up ISC message summary table. Re-instated description of user dictionary messages
01/15/2008	1.15			Deleted duplicate description of 5.8.1.2 User Dictionary.
03/04/2008	1.16			Changed tts_datasource to 0x00 in 5.8.3.1 ISC_TTS_CONFIG_REQ
24/06/2008	1.17			Changed ISC_TTS_UDIC_RESP message ID 0x00CF to 0x00D0

EPSON

International Sales Operations

AMERICA

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

SALES OFFICES
Northeast
301 Edgewater Place, Suite 210
Wakefield, MA 01880, U.S.A.
Phone: +1-800-922-7667 FAX: +1-781-246-5443

EUROPE

EPSON EUROPE ELECTRONICS GmbH
HEADQUARTERS
Riesstrasse 15 Muenchen Bayern, 80992 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

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 (CHINA) CO., LTD.
SHENZHEN BRANCH
12/F, Dawning Mansion, Keji South 12th Road,
Hi- Tech Park, Shenzhen
Phone: +86-755-2699-3828 FAX: +86-755-2699-3838

EPSON TAIWAN TECHNOLOGY & TRADING LTD.
14F, No. 7, Song Ren Road,
Taipei 110
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 CORPORATION
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

GUMI OFFICE
2F, Grand B/D, 457-4 Songjeong-dong,
Gumi-City, KOREA
Phone: +82-54-454-6027 FAX: +82-54-454-6093

SEIKO EPSON CORPORATION
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: 411304401
First Issue October 2007
Printed June 2008 in JAPAN