EPSON
EXCEED YOUR VISION

S1R72U01
Technical Manual

SEIKO EPSON CORPORATION

Rev. 1.00

NOTICE

No part of this material may be reproduced or duplicated in any form or by any means without the written permission of Seiko Epson. Seiko Epson reserves the right to make changes to this material without notice. Seiko Epson does not assume any liability of any kind arising out of any inaccuracies contained in this material or due to its application or use in any product or circuit and, further, there is no representation that this material is applicable to products requiring high level reliability, such as, medical products. Moreover, no license to any intellectual property rights is granted by implication or otherwise, and there is no representation or warranty that anything made in accordance with this material will be free from any patent or copyright infringement of a third party. This material or portions thereof may contain technology or the subject relating to strategic products under the control of the Foreign Exchange and Foreign Trade Law of Japan and may require an export license from the Ministry of Economy, Trade and Industry or other approval from another government agency.

All other product names mentioned herein are trademarks and/or registered trademarks of their respective companies.

Scope of Application

This document applies to the S1R72U01 serial (UART) – USB Host/Device bridge LSI, which supports USB 2.0 FS/LS.

Table of Contents

1. Overview

This document is the Technical Manual for the S1R72U01 serial (UART) – USB Host/Device bridge LSI, which supports USB 2.0 FS/LS (hereinafter referred to as the "LSI"). The document supplements hardware information for the LSI and describes technical specifics such as control and operating procedures for building a system (i.e., a product incorporating the LSI).

For LSI hardware information, refer to the *S1R72U01 Data Sheet*. For other support information, refer to the *S1R72U01 Development Support Manual*.

2. Normative Standard

The LSI complies with the following USB standard:

- Universal Serial Bus Specification Revision 2.0

  Speed mode: FS and LS supported (HS not supported)

- Universal Serial Bus (USB) Device Class Definition for Human Interface Devices (HID) Version 1.11
- Universal Serial Bus (USB) Language Identifiers (LANGIDs) Version 1.0

3. Terminology

SIO:	General term for the serial communications interface.
UART:	Asynchronous serial communications.
USB:	General term for products that can be controlled according to the USB standard.
USB Host:	Product with a Host function that complies with the USB standard.
USB Device:	Product with a Device function that complies with the USB standard.
Class:	Definition relating to control methods and other matters specified in the USB standard.
HID:	Human interface device that complies with the USB standard.
HID Class:	Human interface device class as defined under the USB standard.
HUB:	HUB device specified in the USB standard.
BUS:	Transmission route used to transmit and receive USB data.
Enumeration:	Recognition process between USB Host/Device.
BUS reset:	USB BUS reset that complies with the USB standard.
Hardware reset:	Reset using the XRESET pin on the LSI.
Main CPU:	LSI mounted in the system controlling the LSI.
Write:	An action that sends data from the Main CPU to the LSI.
Read:	An action that sends data from the LSI to the Main CPU.
Transmission:	Sending data from the LSI to the USB.
Reception:	Sending data from the USB to the LSI.
EI request:	Command for controlling the LSI using the SIO. (EPSON Interface)
Complete:	Main CPU or USB processing has ended.
End:	The LSI has ended internal processing. Also includes End as defined in standards.

4. System Configuration

The LSI is provided with a UART user interface (hereafter referred to as the “SIO”) and two bridge interfaces (hereafter referred to as the “USB”), Host and Device.

Refer to Appendix-F, “Other Configuration Examples” for configurations other than the one shown below.

4.1 USB Host Configuration

The configuration shown below controls the LSI as a USB Host. The LSI supports USB Device connections for the HID Class.

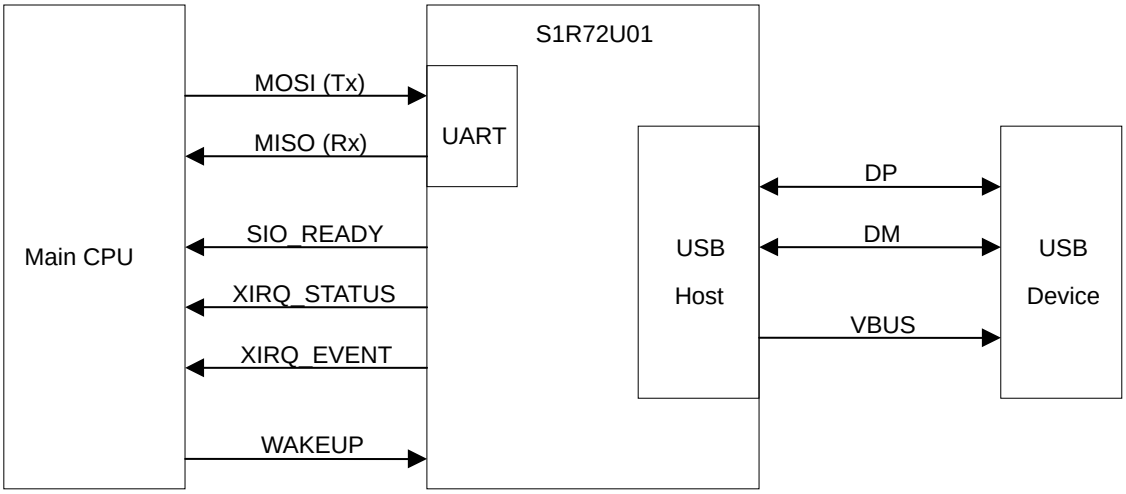


Fig. 4-1 USB Host configuration

4.2 USB Device Configuration

The configuration below controls the LSI as a USB Device. The LSI operates as a HID Class USB Device.

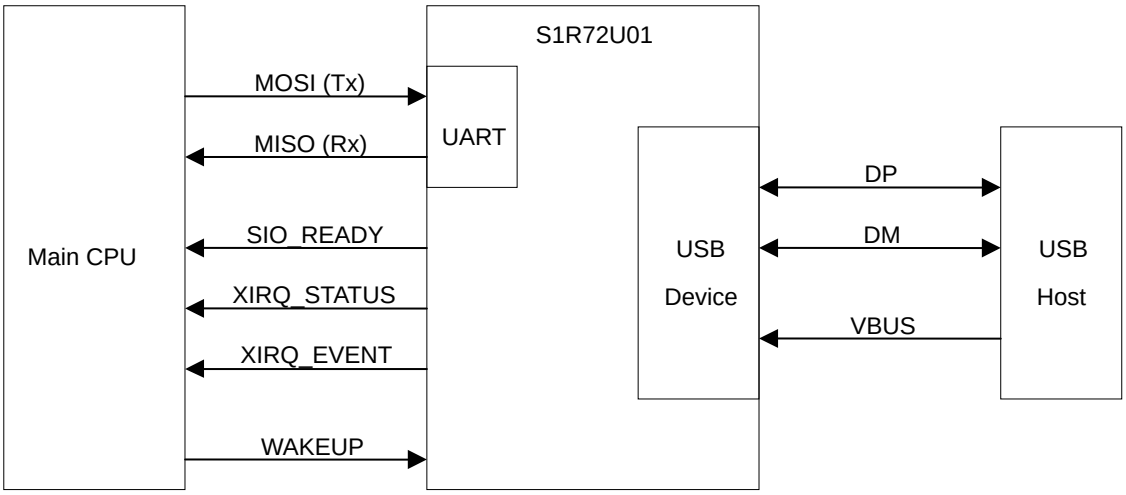


Fig. 4-2 USB Device configuration

5. Functions

The LSI enables USB (Host/Device) control using SIO (serial interface). The LSI does USB standard-compliant operations.

The LSI offers various functions for settings that affect LSI operations and for notifying the Main CPU of the LSI status. The LSI also includes functions that support system development.

Text enclosed within square brackets in this document represents EI requests and EI request codes. For example, [“HID START” (10h)] refers to the EI request “HID START” and its command code, 10h.

For more information, refer to the relevant sections.

5.1 UART Function

The LSI is provided with a UART function (hereafter referred to as the “UART”) as an SIO. Specifics are described below.

5.1.1 UART Overview

When an event is generated in the USB, the LSI uses the UART to automatically transfer information (such as data) to the Main CPU.

When the XIRQ_EVENT pin is set to “enable”, if the Main CPU cannot obtain information (data) correctly from the LSI for some reason (e.g., Busy status), the information can be reacquired with an EI request. For additional details, refer to Section “5.1.5 Event control”.

The initialization flow, sequence, event information, status information, etc. required by the UART function are explained in the following. For error information, refer to Section “5.2.5 Host error information” or Section “5.3.5 Device error information”.

Table 5-1 Outline specifications

Item	Specification
Transfer buffer	Max. 2,048 bytes (Buffer for the LSI internal write data)
Transfer rate	Initial stage: 300 bps or 9,600 bps Normal operation: 300 bps to 3,000,000 bps (3 Mbps)
Command	EI request
Transfer error detection	Supported
Command error detection	Supported
USB error detection	Supported
Flow control	Not supported

5.1.2 UART setting

5.1.2.1 Initial settings

Using the LSI's setting function, set the pins shown in Table 5-2 to the initial settings. For details, refer to Section "5.4 Setting Function".

Table 5-2 Initial settings

Setting item	Pin	Setting
UART selection	SPIxUART	Low
Initial baud rate	INIT_BAUD	Low (300 bps) or High (9,600 bps)
Other	SCK	Low
	SS	*

* The MISO pin status can be controlled by setting this pin.
For more information, refer to the *S1R72U01 Data Sheet*.

5.1.2.2 Communication settings

Using an EI request, set the UART communication conditions. Note that the LSI does not support flow control. Table 5-3 shows default values. For details, refer to Section "7.2.10 F8h_SERIAL PORT".

Table 5-3 Communication settings

Item	Default value
Data bit	8 (fixed)
Stop bit	1
Parity	None
Flow control	None (fixed)

5.1.3 UART initialization flow

Initialize the UART according to the flow shown in Fig. 5-1.

Make sure that the SIO_READY pin is set to High. If the pin does not change to High during initialization, refer to Section “5.6 Development Support Functions”.

When a “SERIAL PORT (F8h) setting” EI request is transmitted, the SIO_READY pin changes to Low. When the setting becomes active, it changes to High.

“LSI SETTING (03h) setting” is not necessary when using the default setting.

The “EVENT INT CONTROL (FFh) setting” is not required when the XIRQ_EVENT pin is set to “disable”.

Regarding the sequence related to the Main CPU control method and other methods, refer to Section “5.1.4 UART sequence”.

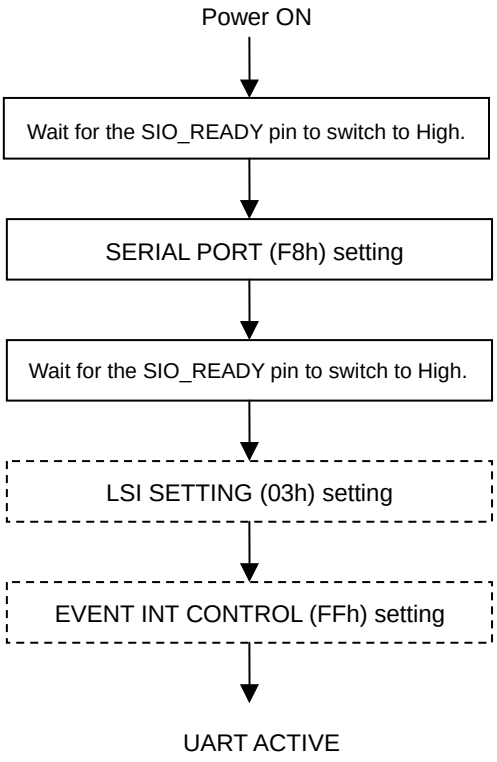


Fig. 5-1 Initialization flow

5.1.4 UART sequence

This section describes the UART sequence between the LSI and the Main CPU.

5.1.4.1 EI header

An EI header is added before the data or notification information transferred from the LSI to the Main CPU. Fig. 5-2 illustrates examples of an EI request and EI header (GET STATUS). The EI header generally contains the same details as the EI request. For more information on commands for which the EI request and EI header differ, refer to the command descriptions in Section “7 Command Specifications”.

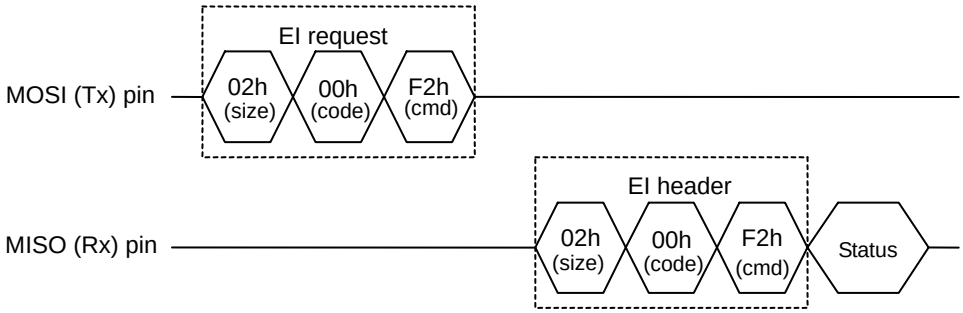


Fig. 5-2 EI header example

5.1.4.2 EI request transfer

Fig. 5-3 shows an EI request transfer written from the Main CPU to the LSI.

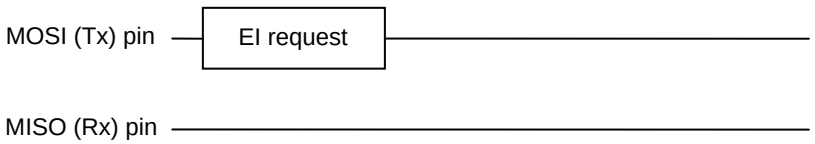


Fig. 5-3 EI request transfer

5.1.4.3 Data Transfer

This section describes the data transfer sequence between the LSI and the Main CPU.

Fig. 5-4 shows the sequence for writing data following an EI request.

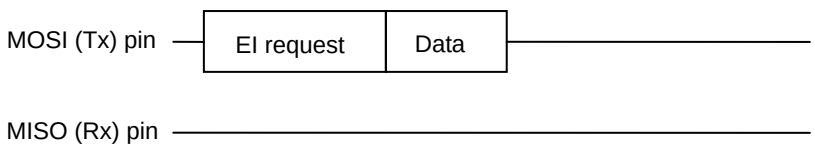


Fig. 5-4 Data writing

Fig. 5-5 shows the data read sequence after an EI request transfer.

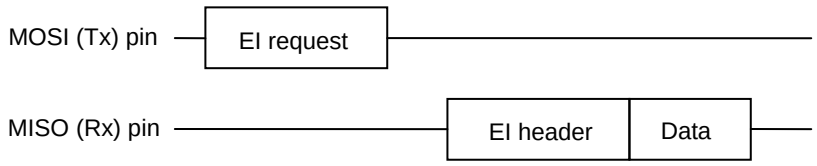


Fig. 5-5 Data reading (1)

Fig. 5-6 shows the data read sequence following the EI header.

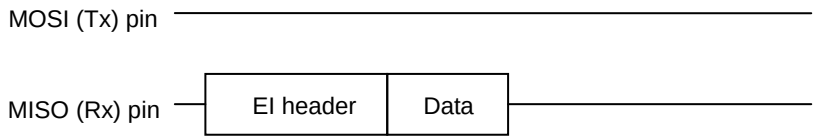


Fig. 5-6 Data reading (2)

Fig. 5-7 shows the data read sequence when the XIRQ_EVENT pin is set to “disable”. Event information is notified from the LSI according to sequence 1. The Main CPU should read the data after the EI request is transferred according to sequence 2.

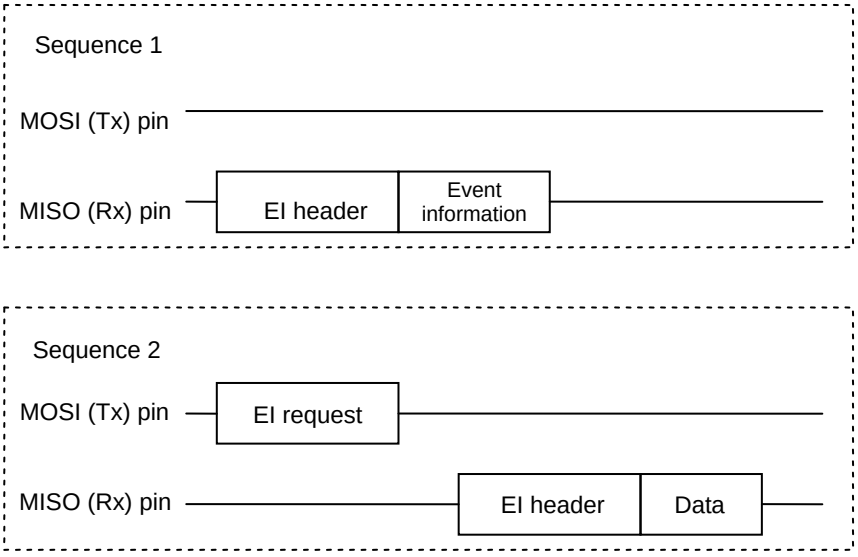


Fig. 5-7 Data reading (3)

Fig. 5-8 shows the data read sequence with the XIRQ_EVENT pin enabled. Event information is notified from the LSI according to sequence 1. The Main CPU should read the event information after the EI request is transferred according to sequence 2. The Main CPU should read the data after the EI request is transferred according to sequence 3.

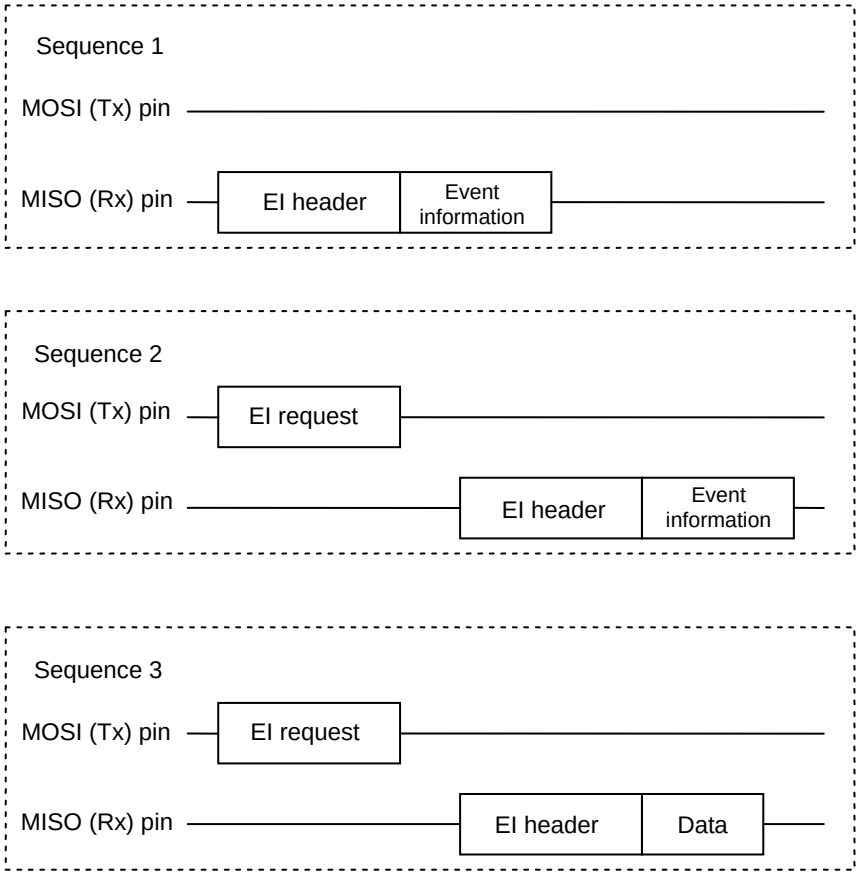


Fig. 5-8 Data reading (4)

5.1.4.4 Notification transfer

This section describes the sequence for notification transfer between the LSI and the Main CPU. Notification information is event information, status information, and error information.

Fig. 5-9 shows the sequence for transferring notification information following the EI header from the LSI.

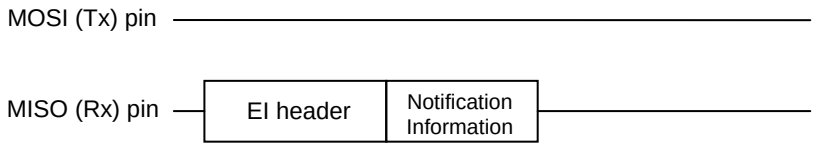


Fig. 5-9 Notification transfer (1)

Fig. 5-10 shows the sequence for transferring notification information from the LSI following the EI request transfer.

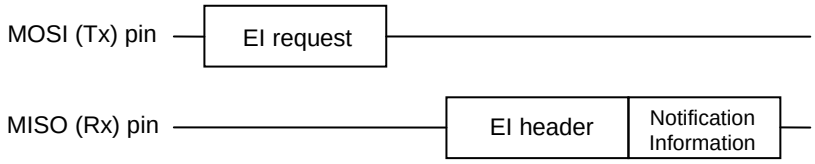


Fig. 5-10 Notification transfer (2)

Fig. 5-11 shows the sequence for reacquiring notification information by the Main CPU after notification transfer from the LSI.

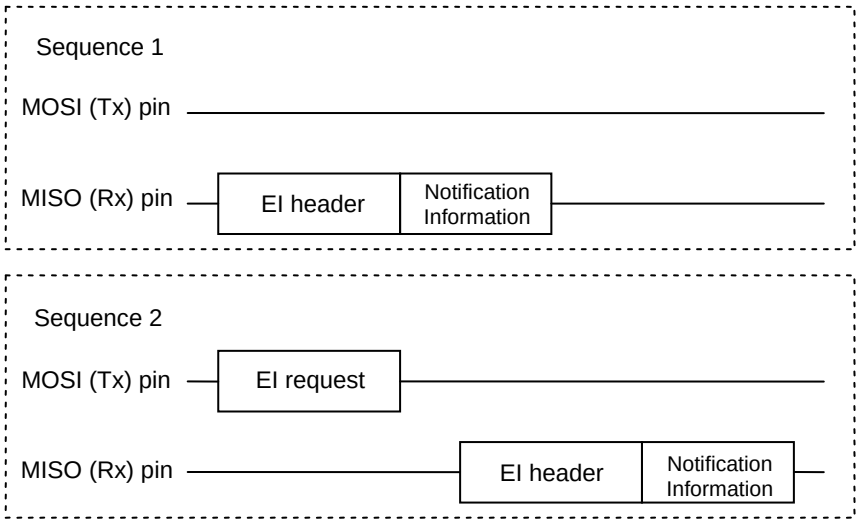


Fig. 5-11 Notification transfer (3)

5.1.5 Event control

Event information is obtained in two ways: directly or reacquired following notification by the XIRQ_EVENT pin. The method for controlling event information varies with pin settings. For more information, refer to Section “7.2.11 FFh_EVENT INT CONTROL”.

The XIRQ_EVENT pin is set to “disable” in the UART default condition. This means event information is acquired directly. Refer to Fig. 5-12.

When the pin is set to “enable”, event information is transferred concurrently with notification from the XIRQ_EVENT pin. However, event information must be reacquired, since control assumes reacquisition. Refer to Fig. 5-13.

For detailed information on events, refer to Section “5.2.4 Host event information” and Section “5.3.4 Device event information”.

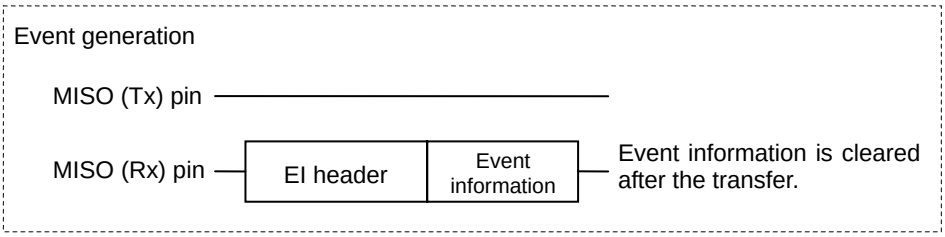


Fig. 5-12 XIRQ_EVENT pin set to disable (default)

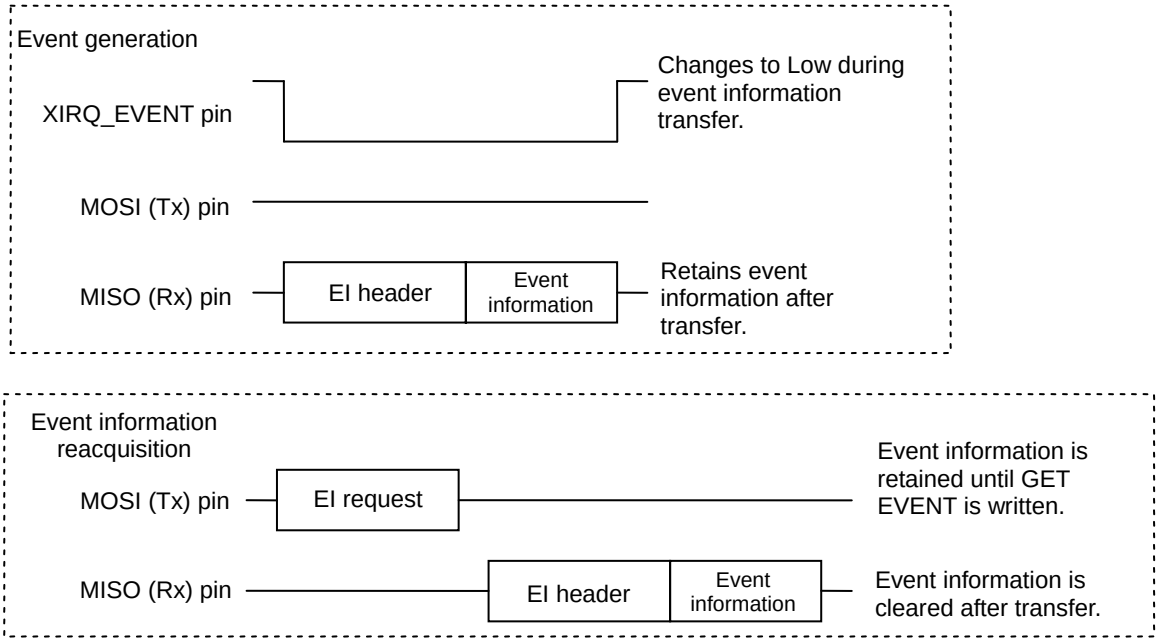


Fig. 5-13 XIRQ_EVENT pin set to Enable

5.1.6 Status information

Table 5-4 lists the status information. The Main CPU is notified of the status of Bits 7 to 4 on detection, as described in Section “5.1.4.4 Notification transfer”. Note that information concerning Bits 3 and 1 to 0 is not sent to the Main CPU if they change and must be obtained using “GET STATUS” (F2h). An error status persists until the next EI request is written.

The status information for the LSI is the same for both the USB Host and Device.

Table 5-4 Status information

Bit	Content	Description
7	Buffer Overflow Error	0b: Normal 1b: Error
6	Parity Error	0b: Normal 1b: Error
5	Framing Error	0b: Normal 1b: Error
4	Noise Detection	0b: Normal 1b: Noise Detection
3	Protocol Error	0b: Normal 1b: Error
2	reserved	
1-0	Condition	00b: Idle 01b: Busy 10b, 11b: reserved

5.1.6.1 Buffer Overflow Error

This is generated when an overflow occurs in the UART reception buffer of the LSI. Refer to Section “5.6 Development Support Functions”.

5.1.6.2 Parity Error

This indicates the detection of a parity error during the reception of UART data by the LSI. Refer to Section “5.6 Development Support Functions”.

5.1.6.3 Framing Error

This indicates the detection of a framing error during the reception of UART data by the LSI. Refer to Section “5.6 Development Support Functions”.

5.1.6.4 Noise Detection

This indicates the detection of noise during the reception of UART data by the LSI. The LSI performs multiple sampling operations for one set of data; if a mismatch results in sampling results, it is determined to be noise. Refer to Section “5.6 Development Support Functions”.

5.1.6.5 Protocol Error

This indicates a protocol error. For more information on errors, refer to Sections “5.2.5 Host error information” or Section “5.3.5 Device error information”.

5.1.6.6 Condition

This indicates the internal state of the LSI. The XIRQ_STATUS pin status changes if the internal state changes. Refer to Section “5.5.2 XIRQ_STATUS” for details.

(1) Idle Indicates a standby state.

(2) Busy Indicates a command is being processed.

5.1.7 Precautions

(1) Be careful with data to be transferred from the LSI, including transfer rate, to prevent overflows on the Main CPU side.

(2) The LSI does not support the reacquisition of transfer data. Design the product (including circuit board wire length and noise susceptibility of circuit patterns) to ensure reliable data acquisition.

(3) If an error occurs during the initialization flow shown in Section “5.1.3 UART initialization flow”, causing the LSI to fail during initialization, the LSI may send no error reply and remain non-responsive. Regarding the analysis of operating status, refer to Section “5.6 Development Support Functions”.

(4) The LSI is designed for one-to-one connections with the Main CPU. It does not support multi-stage connections.

5.2 USB Host Function

The LSI incorporates a USB Host function. This function is described in detail below.

5.2.1 Host function overview

The Host function of the LSI supports LS and FS (HS not supported) for the HID Class. One USB Device at a time can be connected to the LSI.

The LSI handles control processing for USB Devices complying with the USB standard. By performing initial setting and event processing, the Main CPU can easily provide control processing for USB Devices.

The following describes the event information required by the USB Host function as well as error information, VBUS control, connection/disconnection, power management, NSF, TPL and other topics. For more information on HID Class, refer to Section “6 HID Class Overview”.

5.2.2 Host settings

Using the LSI’s setting function, set the pin shown in Table 5-5 to the initial setting. For detailed information, refer to Section “5.4 Setting Function”.

Table 5-5 Initial setting

Setting item	Pin	Setting
HOST selection	HOSTxDEVICE	High

5.2.3 Host initialization flow

Initialize the Host according to the flow shown in Fig. 5-14.

For information on initializing the SIO, refer to Section “5.1.3 UART initialization flow”.

“TPL DOWNLOAD (02h) setting” is not required with default settings. If TPL is downloaded, it must be downloaded each time power is turned on. The TPL downloaded is retained in the LSI RAM.

If a Device is connected when the LSI is in the USB ACTIVE status, connection processing starts automatically. For detailed information, refer to Section “5.2.6 Device connection”. For information on disconnection processing, refer to Section “5.2.7 Device disconnection”.

Refer to Appendix-G, “Initialization Flow” for more information on the procedural flow from initialization to Device connection.

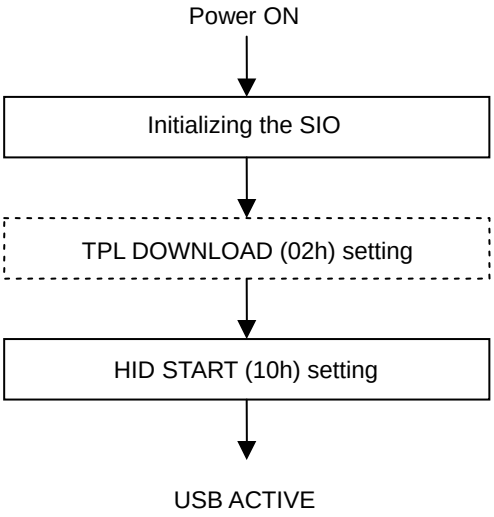


Fig. 5-14 Initialization flow

5.2.4 Host event information

When an event occurs during Host operations, the Main CPU is notified as described in Section “7.2.6 F0h_GET EVENT”. Table 5-6 lists the event information issued by the LSI. Control for event information varies depending on the XIRQ_EVENT pin setting. For more information, refer to Section “5.1.5 Event control”.

Table 5-6 Host event information

Bit	Content	Description
7-4	reserved	
3	Remote Wakeup	0b: Normal 1b: Detect (Event)
2	Rcv Input Report	0b: Normal 1b: Receive (Event)
1	CD Change	0b: Normal 1b: Change (Event)
0	CD (Connection Detect)	0b: Disconnect 1b: Connect

5.2.4.1 Remote Wakeup

This event is generated when a Remote Wakeup occurs in the Device.

5.2.4.2 Rcv Input Report

When the XIRQ_EVENT pin is set to “enable”, an event occurs if data is received via Input Report transfer from a Device. Note that this event does not occur if the XIRQ_EVENT pin is set to “disable”, since data is transferred to the Main CPU when data reception from the Device is completed.

5.2.4.3 CD Change

This event is generated when a change in CD (refer to Section 5.2.4.4) occurs.

5.2.4.4 CD

The status changes when a Device is connected to or disconnected from the LSI and USB processing is completed. For more information on connection and disconnection, refer to Section “5.2.6 Device connection” and Section “5.2.7 Device disconnection”.

5.2.4.5 Event Clear

When the XIRQ_EVENT pin is set to “enable”, the event information is cleared when the “Event information acquisition” (F0h) EI request from the Main CPU is written. When the pin is set to “disable”, the event information is cleared when it is transferred to the Main CPU.

For information on the transition of the XIRQ_EVENT pin status, refer to command descriptions in Section “7 Command Specifications”.

5.2.5 Host error information

When an error occurs during Host operations, the Main CPU is notified as described in Section “7.2.8 F3h_ERROR”. Table 5-7 lists the error information issued by the LSI. The error status is retained until the next EI request is written. Note that the status information in Section “5.1.6.5 Protocol Error” will be “Error” if this error occurs.

Table 5-7 Host error information

Bit	Content	Description
7-3	reserved	
2	EI Req Aborted	0b: Normal 1b: Error
1	Invalid Parameter	0b: Normal 1b: Error
0	Req Unsupported	0b: Normal 1b: Error

5.2.5.1 EI Req Aborted

This error is generated when EI request processing is aborted. This error is caused by the USB.

5.2.5.2 Invalid Parameter

This error occurs if an error is detected in the EI request written from the Main CPU. This error may indicate invalid data in an EI request parameter.

5.2.5.3 Req Unsupported

This error occurs when an EI request cannot be executed, as in the following situations.

(1) An unsupported EI request was written from the Main CPU.

(2) An HID Class EI request was written before setting Start for “HID START” (10h).

5.2.6 Device connection

If “HID START” (10h) is set to Start, the LSI automatically executes enumeration when a Device is connected. When the HID Class control is possible, the LSI issues notification of this state to the Main CPU with an event.

For more information on transfer rates when connected, refer to Section “6.1 Transfer System”.

If “HID START” (10h) is set to Stop, the VBUS output is stopped, in which case Device connections cannot be detected.

5.2.6.1 Connection flow

Set the Device operation in the Main CPU by referring to the flow shown in Fig. 5-15. The process shown in this flowchart must be executed each time a Device is connected.

The “Main CPU: REPORT ID REGISTRATION (11h) setting” is required when using Report Protocol. Obtain the Report Descriptor from the Device connected and set the Report ID registration information. For information on setting examples, refer to Appendix-E, “Report ID Registration Information Setting Examples”. No settings are required when using Boot Protocol.

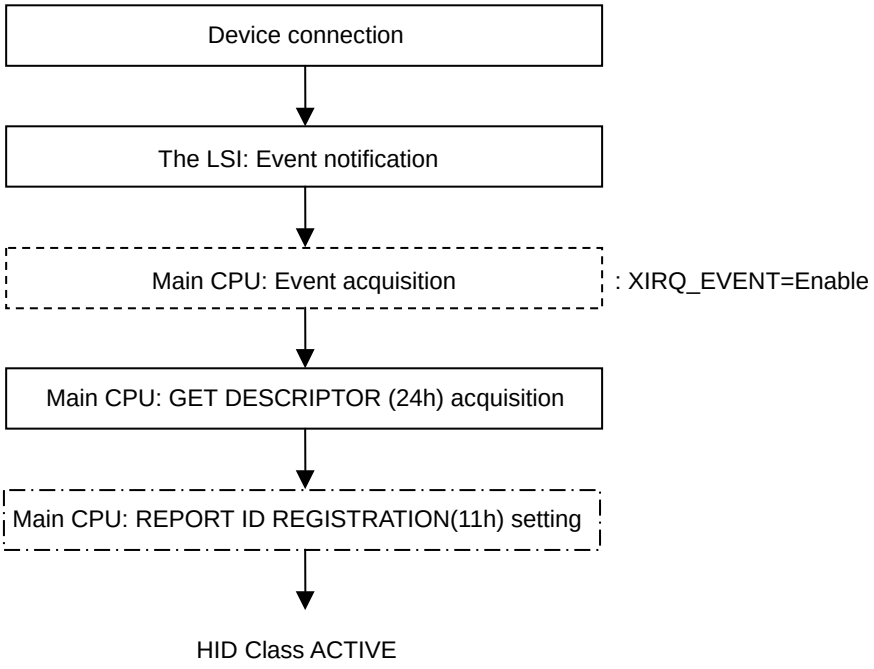


Fig. 5-15 Connection flow

5.2.6.2 Recognition process

When multiple Devices are connected simultaneously, the LSI will recognize the Devices or Interface Descriptors in the order in which they were connected. It will recognize “supported Devices” or “unsupported Devices” from recognized Device information. Examples of the LSI’s recognition process (in the case of two Interface Descriptors) are shown in Table 5-8.

The LSI can retain recognition information on up to three units. It will not recognize four or more units.

The LSI retains information on up to five Interface Descriptors. It will not recognize six or more Interface Descriptors.

Table 5-8 Recognition process

Interface 0	Interface 1	Recognition condition
HID Class Device	Video Class Device	HID Class Devices are recognized as supported Devices. Video Class Devices are recognized as unsupported Devices.
Audio Class Device	HID Class Device	Audio Class Devices are recognized as unsupported Devices. The HID Class is recognized as a supported Device.
Video Class Device	Audio Class Device	Both Classes are recognized as unsupported Devices.
HID Class Device	HID Class Device	HID Class Devices having Interface 0 descriptor are recognized as supported Devices. Interface 1 descriptor is not recognized.

5.2.7 Device disconnection

The disconnect processing performed when a Device is disconnected from the LSI depends on the immediately preceding state. Refer to the following sections in this manual. Note that the operations performed are the same as for disconnection, even when “HID START” (10h) is set to Stop.

5.2.7.1 Disconnect flow

The LSI performs USB disconnect processing if a Device is disconnected from the LSI. The Main CPU is notified of the USB disconnect status by an event (Bit 1 CD Change).

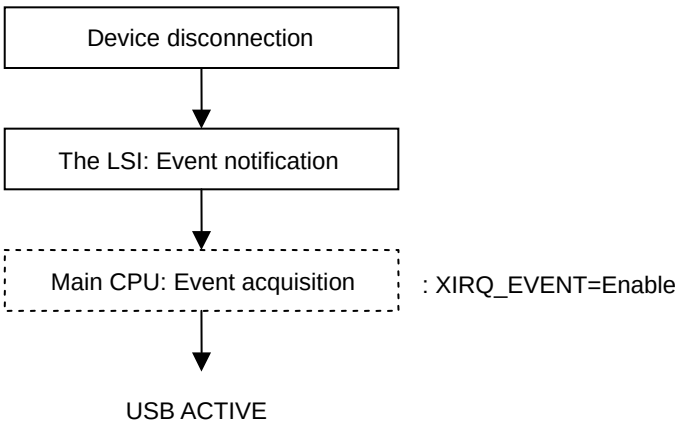


Fig. 5-16 Disconnection flow

5.2.7.2 Disconnect flow during data reads

The LSI performs USB disconnect processing if a Device is disconnected while Report data is being read from the LSI by the Main CPU. When “Bit 2 Rcv Input Report” of event information is 01b, the Main CPU should read the remaining Report data in the LSI using “RECV REPORT (23h) acquisition”.

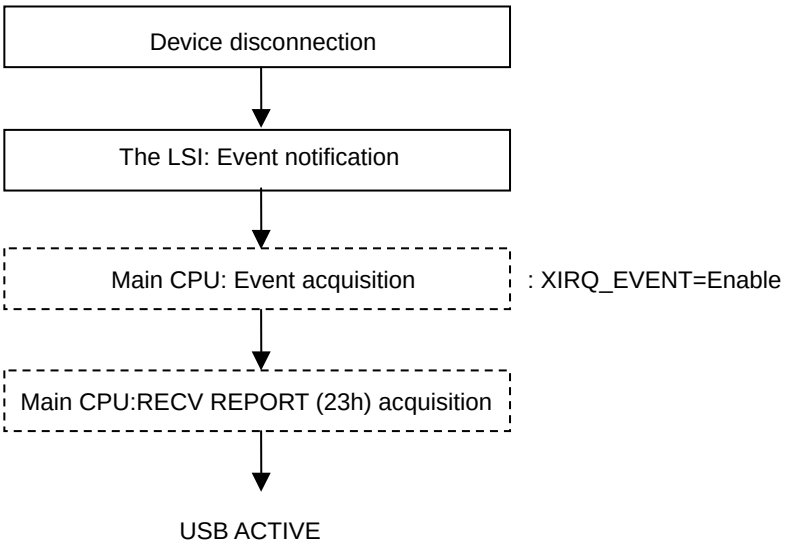


Fig. 5-17 Disconnect flow during data reads

5.2.7.3 Disconnect flow during Sleep

If a Device is disconnected while the LSI is in the Sleep state, notification is provided via the XIRQ_EVENT pin. The Main CPU should set the WAKEUP pin to High. The LSI performs USB disconnect processing and notifies the Main CPU that the USB has been disconnected with an event (Bit 1 CD Change). For the WAKEUP pin, refer to Section “5.4.3 WAKEUP”.

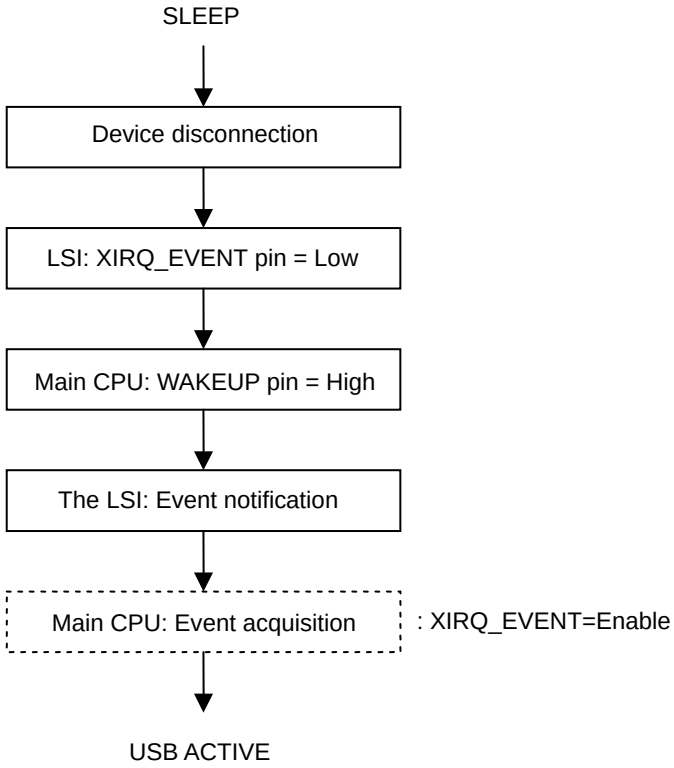


Fig. 5-18 Disconnect flow in Sleep

5.2.8 VBUS control

The LSI incorporates a VBUS supply circuit. It can control a BUS-powered Device as long as the current consumed is lower than the LSI’s supply capacity.

The LSI immediately cuts off the Device connection if it detects a VBUS overcurrent. Approximately one second after the disconnection, the LSI automatically initiates the Device connection process. If the VBUS overcurrent remains, the LSI will repeat the disconnection and connection. Note that the VBUS overcurrent detection setting can be changed. Refer to Section “7.2.5 03h_LSI SETTING”.

The LSI outputs VBUS when the VBUS control is set to Start with an “HID START” (10h) EI request. When VBUS control is set to Stop, the LSI will not output VBUS.

5.2.9 Host power management

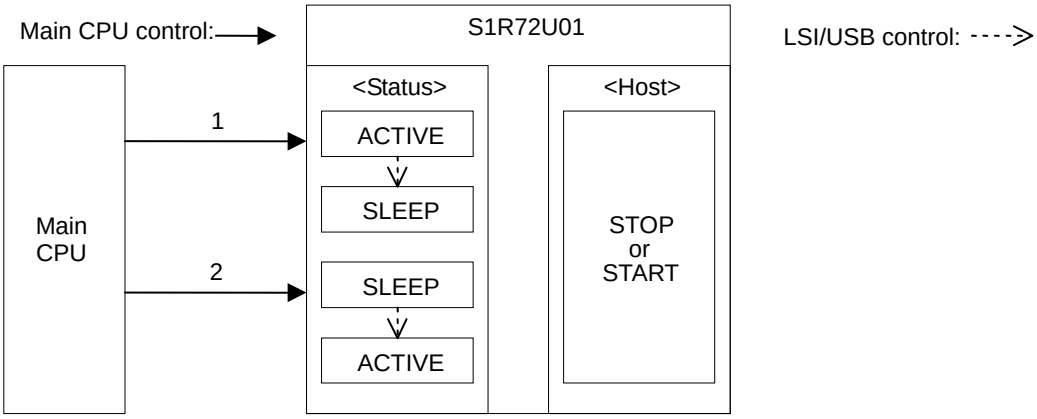
The LSI is equipped with two types of power management functions. One manages the LSI using SLEEP control, while the other manages the USB using the Suspend control. Note that commands cannot be written if the LSI is in SLEEP state, and so it should be switched to ACTIVE state using the WAKEUP pin. Each control method is described below.

5.2.9.1 Control when Device is not connected

Table 5-9 shows the LSI power management method when no Device is connected. A lower power consumption state can be set with this control even if the Host is set to Stop or Start by "HID START" (10h).

Table 5-9 Status control when no Device is connected

Step	Control target	Control method
1	EIreq	Execute "SLEEP" (01h). When the command is recognized, the status changes from ACTIVE to SLEEP. The SIO_READY pin changes from High to Low.
2	WAKEUP pin	Set the WAKEUP pin from Low to High. The status changes from SLEEP to ACTIVE. The SIO_READY pin changes from Low to High.

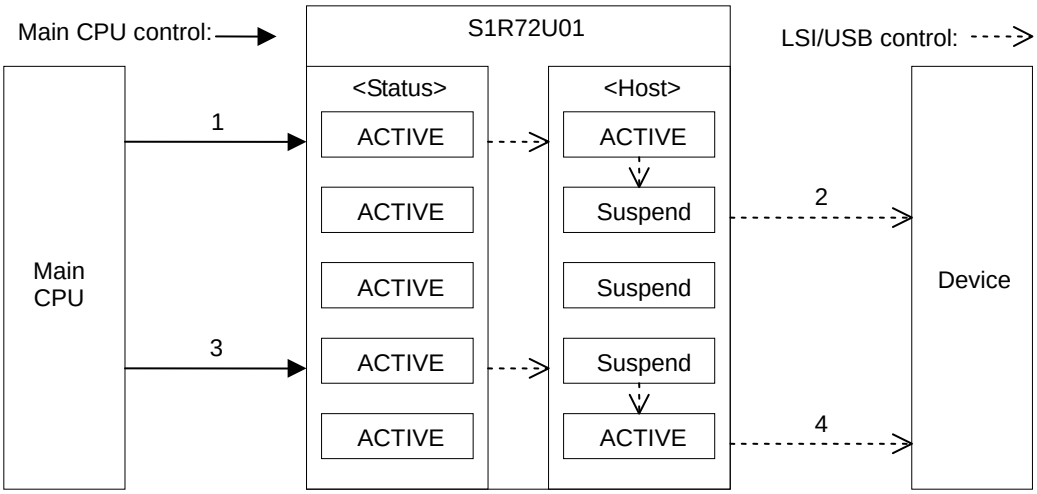


5.2.9.2 Suspend control with Device connected

Table 5-10 gives the Device power management method when a Device is connected. The LSI controls the Device with the Remote Wakeup "Prohibit" setting.

Table 5-10 Suspend status control

Step	Control target	Control method
1	EIreq	Execute "DEVICE POWER MANAGEMENT" (12h). "01h: Suspend & Remote Wakeup Prohibit" setting
2	Suspend	The Host (Device) status changes from ACTIVE to Suspend.
3	EIreq	Execute "DEVICE POWER MANAGEMENT" (12h). "00h: Resume" setting
4	Resume	The Host status changes from Suspend to ACTIVE. The Device is set to Resume.



5.2.9.3 SLEEP control with Device connected

Table 5-11 describes LSI power management when a Device is connected. This control method applies when the Device does not support Remote Wakeup. The LSI controls the Device with the Remote Wakeup "Prohibit" setting.

The Main CPU should determine the Device state from the following event information if a Device is disconnected or reconnected while the LSI is in the Sleep state (between steps 2 and 3).

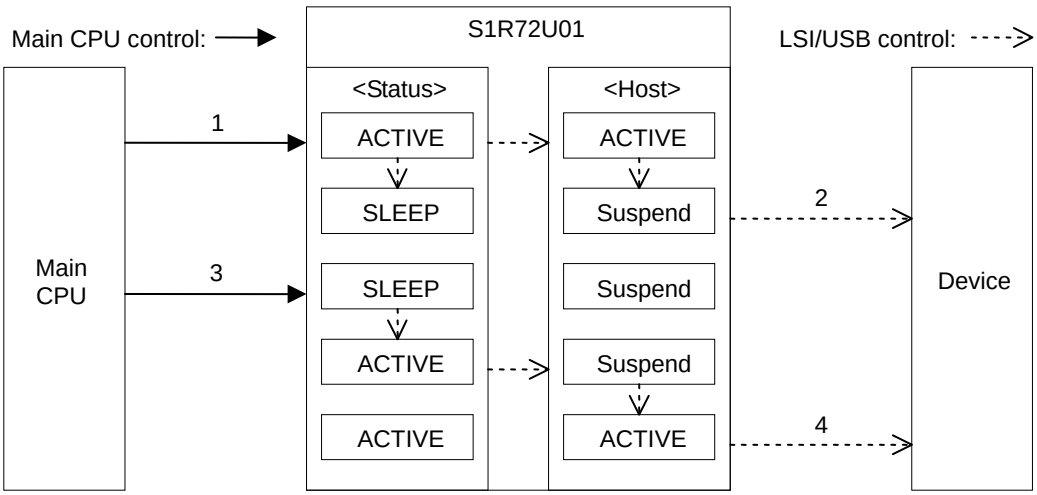
Disconnection: Event information notification is transferred when the XIRQ_EVENT pin changes and the LSI switches to the ACTIVE state. Disconnection can be confirmed using CD in the event information.

Connection: Event information notification is transferred when the XIRQ_EVENT pin changes and the LSI switches to the ACTIVE state. Connection can be confirmed using CD in the event information. Device connection processing should follow "Main CPU: GET DESCRIPTOR (24h) acquisition" in Section "5.2.6.1 Connection flow". If error notification is given for "GET DESCRIPTOR (24h)", this means that the Device has been disconnected and reconnected while in Sleep. Execute "DEVICE RESET" (13h) and perform the Device connection processing described in Section "5.2.6.1 Connection flow".

Note that if the Device connection (or disconnection) state is the same for the LSI before entering Sleep state and after switching from Sleep to ACTIVE; no notification is issued, even if the LSI switches to ACTIVE.

Table 5-11 SLEEP status control

Step	Control target	Control method
1	Elreq	Execute "SLEEP" (01h).
2	Suspend	The Host (Device) status changes from ACTIVE to Suspend. The LSI changes from ACTIVE to SLEEP. The SIO_READY pin changes from High to Low.
3	WAKEUP pin	Set the WAKEUP pin from Low to High.
4	Resume	The LSI changes from SLEEP to ACTIVE. The Host status changes from Suspend to ACTIVE. The Device is set to Resume. The SIO_READY pin changes from Low to High.

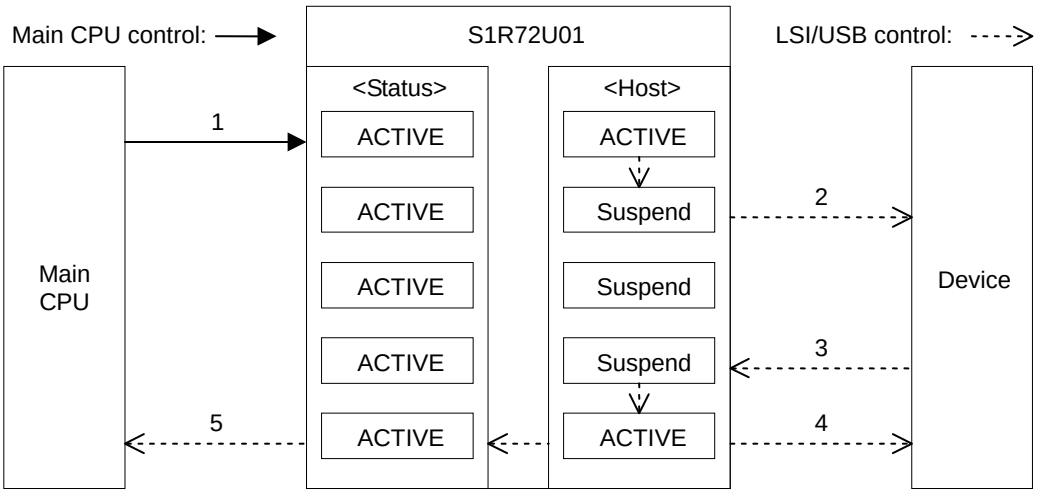


5.2.9.4 Suspend (Remote Wakeup) control with Device connected

Table 5-12 shows Device power management when a Device is connected. The LSI controls the Device with the Remote Wakeup "Permit" setting.

Table 5-12 Remote Wakeup control in Suspend status

Step	Control target	Control method
1	EIreq	Execute "DEVICE POWER MANAGEMENT" (12h). "02h: Suspend & Remote Wakeup Permit" setting
2	Suspend	The Host (Device) status changes from ACTIVE to Suspend.
3	Remote Wakeup	"Remote Wakeup" from the Device is detected.
4	Resume	The Host status changes from Suspend to ACTIVE. The Device is set to Resume.
5	Event information	Event generation is notified to the Main CPU.



5.2.9.5 SLEEP (Remote Wakeup) control with Device connected

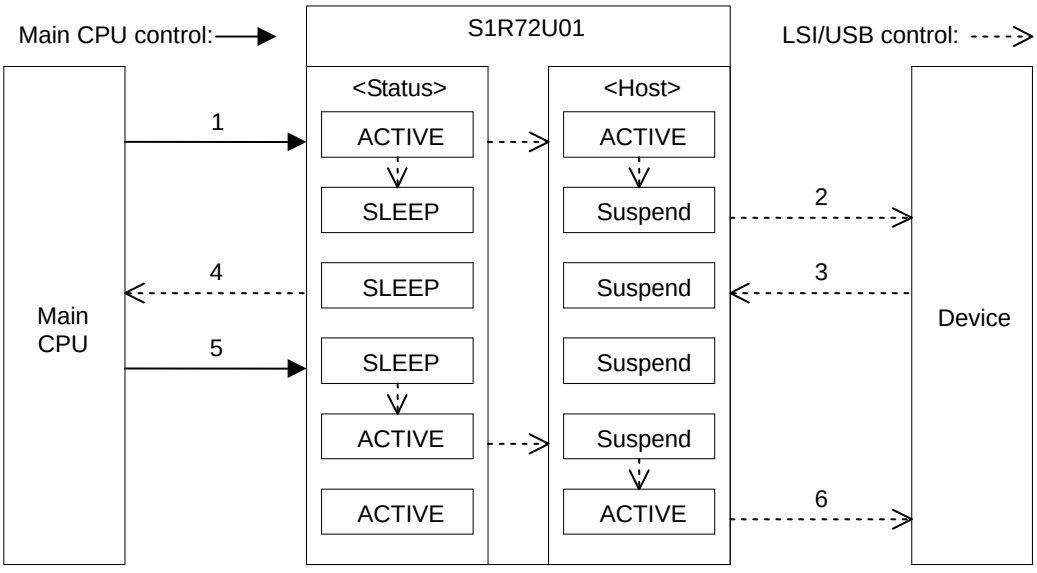
Table 5-13 shows LSI power management method when a Device is connected. This control method applies when the Device supports Remote Wakeup. The LSI controls the Device with the Remote Wakeup "Permit" setting.

If a Device is disconnected, then reconnected while the LSI is in the SLEEP state (Step 2 and thereafter), the LSI will not be able to detect the disconnection/reconnection. To determine if the status is resulted from Remote Wakeup or from disconnection/reconnection occurred during the SLEEP status, have the Main CPU perform the following process after the SIO_READY pin changes to High in Step 6.

Write "GET DESCRIPTOR" (24h). If this is read properly, the status is determined to be Remote Wakeup. If error notification is sent, this means that the Device has been disconnected and reconnected while in Sleep. Execute "DEVICE RESET" (13h) and perform the Device connection processing described in Section "5.2.6.1 Connection flow".

Table 5-13 Remote Wakeup control in SLEEP state

Step	Control target	Control method
1	EIreq	Execute "SLEEP" (01h).
2	Suspend	The Host (Device) status changes from ACTIVE to Suspend. The LSI changes from ACTIVE to SLEEP. The SIO_READY pin changes from High to Low.
3	Remote Wakeup	"Remote Wakeup" is detected from the Device.
4	XIRQ_EVENT pin	The XIRQ_EVENT pin changes from High to Low.
5	WAKEUP pin	Set the WAKEUP pin from Low to High.
6	Resume	The LSI changes from SLEEP to ACTIVE. The Host status changes from Suspend to ACTIVE. The Device is set to Resume. The SIO_READY pin changes from Low to High.



5.2.10 NSF

NSF (No Silent Failures) is a function that issues notification of errors detected by the Host to the Main CPU. For more information, refer to Section “5.5 Notification Function”.

5.2.11 TPL

TPL (Target Peripheral List) is a list of supported Devices that can be recognized by the Host. Setting TPL makes all Devices other than those explicitly mentioned in the list unsupported Devices. By default, the LSI is set to recognize all HID Class Devices as supported Devices.

If the user’s system requires individual settings, a TPL must be produced. For detailed information, refer to the *S1R72U01 Development Support Manual*.

5.2.12 Precautions

(1) When a Device is disconnected, the LSI clears the Report ID registration information set in the process described in Section “5.2.6 Device connection”. It must be set each time a Device is connected.

(2) Reset the LSI using hardware reset if the Device connected causes an error and fails to respond (the Device continually gives NAK or STALL responses and the LSI does not change from Busy status).

(3) A Device is present that does not comply with USB standard or that has an incorrect descriptor description. The LSI may not recognize such Devices.

(4) Any single Device having six Interface Descriptors (e.g., 1 to 5: Audio Class, 6: HID Class) will be regarded as an unsupported device, since the LSI cannot recognize the 6th Interface Descriptor.

(5) The Device will be treated as unsupported if the total size of the following descriptors exceeds 512 bytes.

- Configuration Descriptor
- Interface Descriptor
- Endpoint Descriptor
- HID Descriptor

(6) Connecting a Device may generate a VBUS overcurrent. Particularly for BUS-powered Devices, the description in the descriptor may differ from the actual current consumed by the Device.

(7) If the connected Device is not recognized, the Device may be defective or its battery (for battery-operated Devices) may be dead.

(8) Various precautions apply to the pins described in Section “5.4 Setting Function”. Refer to the corresponding section of this manual.

5.3 USB Device Function

The LSI incorporates a USB Device function. The details of this function are described below.

5.3.1 Device function overview

The Device function of the LSI supports LS and FS (HS not supported) for HID Class Devices.

The LSI performs control processing for USB Devices complying with the USB standard. By performing initial setting and event processing, the Main CPU can easily provide control processing for USB Devices.

The event information required by the USB Device function and error information, operation settings, connection/disconnection, power management, and other topics are discussed below. For information on the HID Class, refer to Section “6 HID Class Overview”.

5.3.2 Device settings

Using the LSI’s setting function, set the pin shown in Table 5-14 to the initial setting. For more information, refer to Section “5.4 Setting Function”.

Table 5-14 Initial setting

Setting item	Pin	Setting
DEVICE selection	HOSTxDEVICE	Low

5.3.3 Device initialization flow

Initialize the Device according to the flow shown in Fig. 5-19.

For more information on initializing the SIO, refer to Section “5.1.3 UART initialization flow”.

The “Descriptor DOWNLOAD (02h) setting” must be downloaded each time power is turned on. The Descriptor downloaded is retained in the LSI RAM.

The “INITIAL FEATURE REPORT (24h) setting” is not required if Feature Report is not used.

If the LSI is connected to a Host while it is in USB ACTIVE status, connection processing will start automatically. For detailed information, refer to Section “5.3.7 Host connection”. For information on disconnect processing, refer to Section “5.3.8 Host disconnection”.

If the system is configured as a “BUS power device”, the time elapsed from “Power ON” to “HID START (10h) setting” must be set to within 70 ms to comply with the USB standard. Similarly, the VBUS current consumption must not exceed 100 mA from “Power ON” until “CD Change” or “CD (Connect)” events occur.

Refer to Appendix-G, “Initialization Flow” for more information on the procedural flow from initialization to Host connection.

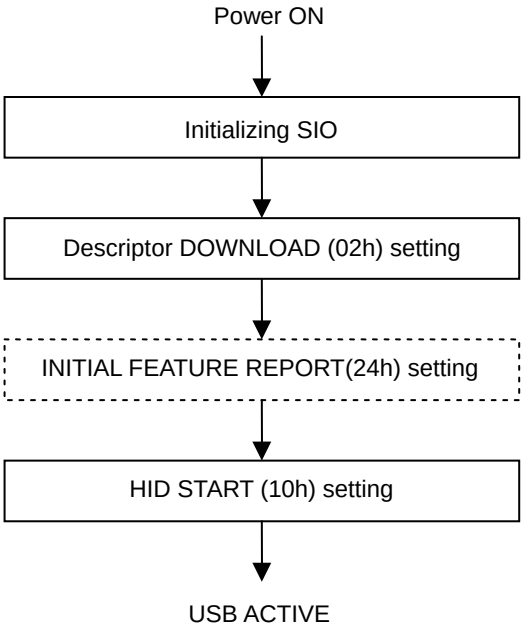


Fig. 5-19 Initialization flow

5.3.4 Device event information

When an event occurs during Device operations, the Main CPU is notified as described in Section “7.2.6 F0h_GET EVENT”. Table 5-15 lists the event information issued by the LSI. Control for event information varies with the XIRQ_EVENT setting. For more information, refer to Section “5.1.5 Event control”.

Table 5-15 Device event information

Bit	Content	Description
7	USB Com Status	0b: Inactive 1b: Active
6	USB Suspend Status	0b: Resume (Event) 1b: Suspend (Event)
5	Detect Reset	0b: Normal 1b: Detect (Event)
4	Protocol Mode Change	0b: Normal 1b: Receive (Event)
3	Rcv Feature Report	0b: Normal 1b: Receive (Event)
2	Rcv Output Report	0b: Normal 1b: Receive (Event)
1	CD Change	0b: Normal 1b: Change (Event)
0	CD (Connection Detect)	0b: Disconnect 1b: Connect

5.3.4.1 USB Com Status

This indicates that the LSI is connected to or disconnected from a Host. “Active” for this status indicates a physical connection.

5.3.4.2 USB Suspend Status

An event occurs when the USB state changes from Resume to Suspend or from Suspend to Resume. This event is not cleared as described in Section “5.3.4.9 Event Clear”, and the USB BUS state for the time at which the event information was read is indicated. Note that “Resume” will indicate an Active state after the event occurs.

5.3.4.3 Detect Reset

This event is generated when a USB BUS reset is detected when CD (refer to Section “5.3.4.8 CD”) is in the Connect state.

5.3.4.4 Protocol Mode Change

This event is generated when a protocol mode setting is received from the Host. The Main CPU should check the protocol using the “GET PROTOCOL MODE” (25h) EI request.

5.3.4.5 Rcv Feature Report

This event is generated when data is received from the Host via Feature Report transfer. When the XIRQ_EVENT pin is set to “disable”, data is sent to the Main CPU as soon as reception of data from the Host is completed. If so, this event is not generated.

5.3.4.6 Rcv Output Report

This event is generated when data is received from the Host via Output Report transfer. As described in Section “5.3.4.5 Rcv Feature Report”, when the XIRQ_EVENT pin is set to “disable”, this event is not generated.

5.3.4.7 CD Change

This event is generated when the CD status (refer to Section 5.3.4.8) changes.

5.3.4.8 CD

The status changes when the LSI is connected to or disconnected from the Host and USB processing is completed. For information on connections/disconnections, refer to Section “5.3.7 Host connection” and Section “5.3.8 Host disconnection”.

5.3.4.9 Event Clear

When the XIRQ_EVENT pin is set to “enable”, the event information is cleared when the “Event information acquisition” (F0h) EI request from the Main CPU is written. When the pin is set to “disable”, the event information is cleared when it is transferred to the Main CPU.

For information on the status transition of the XIRQ_EVENT pin, refer to the command descriptions in Section “7 Command Specifications”.

5.3.5 Device error information

When an error occurs during Device operations, the Main CPU is notified as described in Section “7.2.8 F3h_ERROR”. Table 5-16 lists the error information issued by the LSI. The error status is retained until the next EI request is written. Note that the status information described in Section “5.1.6.5 Protocol Error” will be “Error” if this error occurs.

Table 5-16 Device error information

Bit	Content	Description
7	HID Start Failed	0b: Normal 1b: Error
6	HID Req Failed	0b: Normal 1b: Error
5-3	reserved	
2	EI Req Aborted	0b: Normal 1b: Error
1	Invalid Parameter	0b: Normal 1b: Error
0	Req Unsupported	0b: Normal 1b: Error

5.3.5.1 HID Start Failed

This error is generated when Start processing for “HID START” (10h) cannot be executed. This error may indicate that the descriptor setting or Report information initial setting is invalid.

5.3.5.2 HID Req Failed

This error is generated when data transfer processing with the Host cannot be executed. This error may indicate that the USB is in a Suspend state.

5.3.5.3 EI Req Aborted

This error is generated when EI request processing is aborted. This error is caused by the USB.

5.3.5.4 Invalid Parameter

This error occurs if an error is detected in the EI request written from the Main CPU. This error may indicate invalid data in an EI request parameter.

5.3.5.5 Req Unsupported

This error occurs when an EI request cannot be executed, as in the following situations.

(1) An unsupported EI request was written from the Main CPU.

(2) An HID Class EI request was written before setting Start for “HID START” (10h).

5.3.6 Device operation setting

USB Device operations and functions are defined by Device data called descriptors. Descriptor formats and content are specified by the USB standard. Certain descriptors are common to all Devices, while others are specific to a Class. The HID Class of the LSI uses HID Descriptors and Report Descriptors. Fig. 5-20 shows the descriptors. Note that Physical Descriptors are not supported.

Set the Device operating conditions and other parameters in the descriptors. Values indicated as “fixed” in the following pages are values specified under the USB standard or other regulations. For examples of descriptor settings, refer to Appendix-A, “Descriptor Setting Examples”.

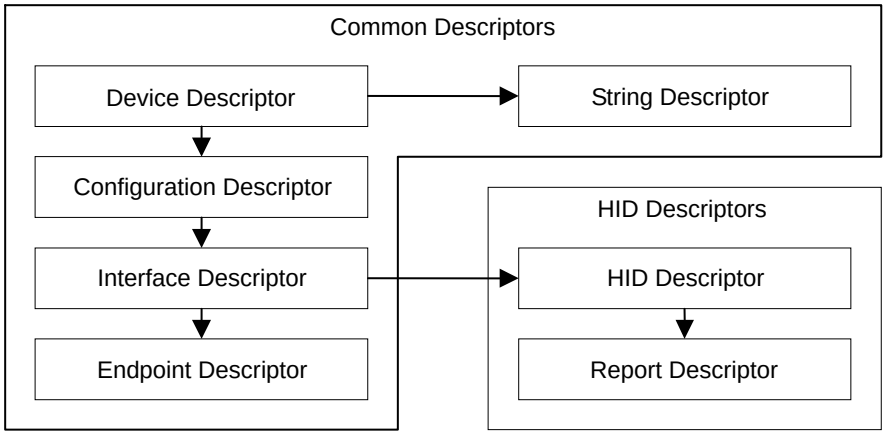


Fig. 5-20 Descriptors

The LSI requires Descriptor Header and Report ID Registration Information as special descriptors. Fig. 5-21 shows the configuration of all descriptors used by the LSI.

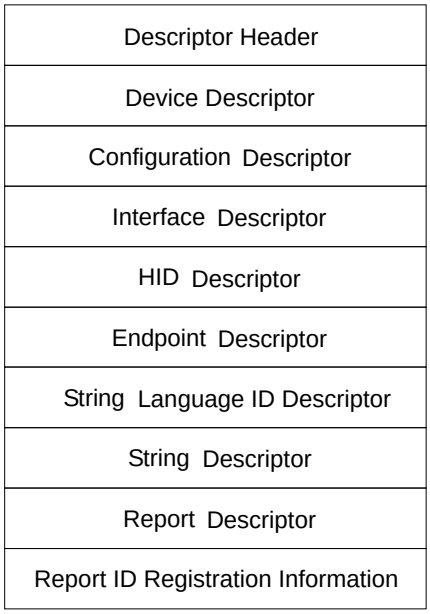


Fig. 5-21 Descriptor configuration

5.3.6.1 Descriptor Header

Set the information shown in Table 5-17 in descriptors used with the LSI. Set this Descriptor Header at the beginning of the descriptors. Data (indicated by "xxxxh" in the "Value" column) that need to be set individually must be entered manually.

wTotalSize: Keep descriptor size below 1,012 bytes.

wOffsetForDescriptorInfo: Set the value of the offset from the beginning.

wOffsetForClassPeculiarInfo1: Same as above.

wOffsetForClassPeculiarInfo2: Same as above.

Note that an error will occur if the settings contain an error when "DOWNLOAD" (02h) is executed.

Table 5-17 Descriptor Header

Content	Size (Byte)	Value	Description
wTotalSize	2	xxxxh	Total size of descriptors including wTotalSize
wDescriptorInfo	2	Fixed	Device Descriptor data
wOffsetForDescriptorInfo	2	xxxxh	Offset value of Device Descriptor (from wTotalSize to beginning of Device Descriptor)
wClassPeculiarInfo1	2	Fixed	Report Descriptor data
wOffsetForClassPeculiarInfo1	2	xxxxh	Offset value of Report Descriptor (from wTotalSize to beginning of Report Descriptor)
wClassPeculiarInfo2	2	Fixed	Report ID Registration Information data
wOffsetForClassPeculiarInfo2	2	xxxxh	Offset value of Report ID Registration Information (from wTotalSize to beginning of Report ID Registration Information)

5.3.6.2 Device Descriptor

Set the basic information specific to the Device. Since the USB standard specifies fixed values, data (indicated by "xxxxh" or "xxh" in the "Value" column) that need to be set individually must be entered manually.

bMaxPacketSize0: Set the maximum packet size for endpoint 0 to suit the transfer speed.

idVendor: Set the value assigned by the USB-IF.

idProduct: Set a desired number used for product identification.

bcdDevice: Set a desired number (product version).

iManufacturer: Set a desired number (normally "1").

iProduct: Same as above (normally "2").

iSerialNumber: Same as above (normally "3").

Table 5-18 Device Descriptor

Content	Size (Byte)	Value	Description
bLength	1	Fixed	Size of this descriptor
bDescriptorType	1	Fixed	Type for this descriptor
bcdUSB	2	Fixed	Release number in accordance with USB specifications
bDeviceClass	1	Fixed	Class code
bDeviceSubClass	1	Fixed	Sub-class code
bDeviceProtocol	1	Fixed	Protocol code
bMaxPacketSize0	1	xxh	Maximum packet size for Endpoint 0 LS : 08h FS : 40h
idVendor	2	xxxxh	Vender ID
idProduct	2	xxxxh	Product ID
bcdDevice	2	xxxxh	Device release number
iManufacturer	1	xxh	Index number of string descriptor indicating manufacturer
iProduct	1	xxh	Index number of string descriptor indicating product
iSerialNumber	1	xxh	Index number of string descriptor indicating device serial number
bNumConfigurations	1	Fixed	Configurable number

5.3.6.3 Configuration Descriptor

Set the Device configuration information. Since the USB standard specifies fixed values, data (indicated by "xxxxh" or "xxh" in the "Value" column) that need to be set individually must be entered manually.

wTotalLength: Set the total if more than one descriptor such as Endpoint Descriptor is involved.

bmAttributes: Set the operation of the Device.

bMaxPower: Set the VBUS consumed current for self-powered Devices.
Set a value of less than 500 mA (FAh) for BUS-powered Devices.

Note that an error will occur if the wTotalLength setting contains an error when "DOWNLOAD" (02h) is executed.

Table 5-19 Configuration Descriptor

Content	Size (Byte)	Value	Description
bLength	1	Fixed	Size of this descriptor
bDescriptorType	1	Fixed	Type for this descriptor
wTotalLength	2	xxxxh	Total size of the following descriptors Configuration Descriptor Interface Descriptor Endpoint Descriptor HID Descriptor
bNumInterfaces	1	Fixed	Number of interfaces
bConfigurationValue	1	Fixed	Value for Set configuration request
iConfiguration	1	Fixed	Offset value of string descriptor indicating configuration
bmAttributes	1	xxh	Set the following information. Bit 7: reserved (fixed at "1") Bit 6: Power setting 0b BUS power 1b Self power Bit 5: Remote Wakeup setting 0b OFF 1b ON Bits 4 to 0: reserved (fixed at "0")
bMaxPower	1	xxh	Set maximum BUS consumed current in 2mA steps (max. 500 mA). 2 to 500 mA (01h to FAh)

5.3.6.4 Interface Descriptor

Set information related to the interface. Since the USB standard specifies fixed values, data (indicated by “xxh” in the “Value” column) that need to be set individually must be entered manually.

bNumEndpoints: Set the total number of endpoints used, excluding Endpoint 0.

bInterfaceSubClass: Set the operation of the Device.

bInterfaceProtocol: Specify keyboard or mouse at the time of the Boot Protocol.

Table 5-20 Interface Descriptor

Content	Size (Byte)	Value	Description
bLength	1	Fixed	Size of this descriptor
bDescriptorType	1	Fixed	Type for this descriptor
bInterfaceNumber	1	Fixed	Interface number
bAlternateSetting	1	Fixed	Value applicable when alternative setting is used
bNumEndpoints	1	xxh	Total number of endpoints, excluding Endpoint 0 (Specify “1” or “2”.)
bInterfaceClass	1	Fixed	Designation of HID Class
bInterfaceSubClass	1	xxh	Designation of sub-class 00h: No subclass 01h: Boot Protocol
bInterfaceProtocol	1	xxh	Designation of protocol 00h: None 01h: Keyboard 02h: Mouse
iInterface	1	Fixed	Offset value of string descriptor indicating this descriptor

5.3.6.5 HID Descriptor

Set HID Class information. Since the USB standard specifies fixed values, data (indicated by “xxxxh” or “xxh” in the “Value” column) that need to be set individually must be entered manually.

bCountryCode: Set a code identifying the country as necessary. For more information on country codes, refer to Appendix-C, “Country Code”.

wDescriptorLength: Set the Report Descriptor size.

Note that an error will occur if the wDescriptorLength setting differs from the Report Descriptor size when “DOWNLOAD” (02h) is executed.

Table 5-21 HID Descriptor

Content	Size (Byte)	Value	Description
bLength	1	Fixed	Size of this descriptor
bDescriptorType	1	Fixed	Type for this descriptor
bcdHID	2	Fixed	Release No. depending on HID Class specifications
bCountryCode	1	xxh	Country code
bNumDescriptors	1	Fixed	Number of Report descriptors
bDescriptorType	1	Fixed	Report descriptor type
wDescriptorLength	2	xxxxh	Report descriptor size

5.3.6.6 Endpoint Descriptor

Set the information related to endpoints. Since the USB standard specifies fixed values, data (indicated by "xxxxh" or "xxh" in the "Value" column) that need to be set individually must be entered manually.

bEndpointAddress: Set the endpoint.

wMaxPacketSize: Set the maximum packet size to suit the transfer speed.

bInterval: Set the polling interval to suit the transfer speed.

Table 5-22 Endpoint Descriptor

Content	Size (Byte)	Value	Description
bLength	1	Fixed	Size of this descriptor
bDescriptorType	1	Fixed	Type for this descriptor
bEndpointAddress	1	xxh	Endpoint setting Bit 7: Direction 0b OUTPUT 1b INPUT Bits 6 to 4: reversed Bits 3 to 0: Endpoint number
bmAttributes	1	Fixed	Designation of Interrupt transfer
wMaxPacketSize	2	xxxxh	Maximum packet size LS: 1 to 8 bytes (0001h to 0008h) FS: 1 to 64 bytes (0001h to 0040h)
bInterval	1	xxh	Setting of polling interval in 1-ms increments LS: 8 to 255 ms (08h to FFh) FS: 1 to 255 ms (01h to FFh)

5.3.6.7 String Language ID Descriptor

Set the language code to be used for String Descriptors. This setting applies to all String Descriptors. Since the USB standard specifies fixed values, data (indicated by “xxxxh” in the “Value” column) that need to be set individually must be entered manually.

For more information on UNICODE language codes, refer to Section “2 Normative Standard”.

Table 5-23 String Language ID Descriptor

Content	Size (Byte)	Value	Description
bLength	1	Fixed	Size of this descriptor
bDescriptorType	1	Fixed	Type for this descriptor
wLangID	2	xxxxh	UNICODE language code

5.3.6.8 String Descriptor

Set the string. Since the USB standard specifies fixed values, data (indicated by “xxh” in the “Value” column) that need to be set individually must be entered manually.

For examples of UNICODE text strings, refer to Appendix-B, “UNICODE”.

This descriptor is the String Descriptor independently specified by “iManufacturer”, “iProduct”, and “iSerialNumber” described in Section “5.3.6.2 Device Descriptor”.

Table 5-24 String Descriptor

Content	Size (Byte)	Value	Description
bLength	1	xxh	Size of this descriptor
bDescriptorType	1	Fixed	Type for this descriptor
bString[0]	1	xxh	UNICODE character string
...			
bString[n]	1	xxh	UNICODE character string

5.3.6.9 Report Descriptor

Set information related to the HID Class Report. The following Item Tags must be set; other Item Tags are optional. For examples of settings, refer to Appendix-D, “Report Descriptor Setting Examples”.

USAGE, USAGE_PAGE: USAGE and USAGE_PAGE are combined, and USAGE is used to access a Report for use in controlling the HID Class and acquisition of information.

REPORT_COUNT: Set the number of Reports.

REPORT_SIZE: Set the size of the Reports in bits.

LOGICAL_MINIMUM: Set the minimum value of the Report.

LOGICAL_MAXIMUM: Set the maximum value of the Report.

INPUT (OUTPUT or FEATURE): The Item Tag defined before the INPUT determines the INPUT property.

Table 5-25 Report Descriptor

Item Tag	Value (Bit)*	Description
USAGE_PAGE	0000 01nn	Specify the Usage page.
USAGE	0000 10nn	Specify the Usage.
COLLECTION	1010 00nn	Grouping of Item Tags defined during END_COLLECTION. COLLECTION may become nested.
USAGE	0000 10nn	Specify the Usage. This Usage is correlated with INPUT (OUTPUT, FEATURE) defined thereafter. If the UASGE is a sequential number, it can be defined using USAGE_MINIMUM and USAGE_MAXIMUM.
USAGE_PAGE	0000 01nn	Specify the Usage page.
USAGE_MINIMUM	0001 10nn	Minimum value of Usage
USAGE_MAXIMUM	0010 10nn	Maximum value of Usage
REPORT_ID	1000 01nn	Define when using multiple Reports of the same type. For example, when an 8-byte INPUT Report (keyboard) and 4-byte INPUT Report (mouse) are transferred using the same endpoint, this is used to distinguish the data.
REPORT_COUNT	1001 01nn	Number of Reports
REPORT_SIZE	0111 01nn	Report size
LOGICAL_MINIMUM	0001 01nn	Logical minimum value of Report
LOGICAL_MAXIMUM	0010 01nn	Logical maximum value of Report
INPUT	1000 00nn	Definition of INPUT Report INPUT, OUTPUT, or FEATURE Report can be defined continuously.
OUTPUT	1001 00nn	Definition of OUTPUT Report
FEATURE	1011 00nn	Definition of FEATURE Report
END_COLLECTION	1100 00nn	End of COLLECTION

*Specify the data size of each Item Tag in “nn”.

5.3.6.10 Report ID Registration Information

Set the Report ID registration information. Data (indicated by “xxxxh” or “xxh” in the “Value” column) that need to be set individually must be entered manually.

When using multiple Reports, set “bReportType” through “wReportLen” for each Report. The total number of Reports must not exceed 32 (“bNumInitReports” value). The total size of all Report data (“wReportLen” total) must not exceed 544 bytes.

bNumInitReports: Set the total number of Reports used.

bReportType: Set the Report type.

bReportID: Set the Report ID number (01h to FFh).
Set “00h” when this is not used. “00h” cannot be used for Report numbers.

wReportLen: Set the Report data size.
For multiple Report IDs, the size must include 1-byte Report IDs. Refer to Section “6.3 Report ID” for more information.

Table 5-26 Report ID Registration Information

Content	Size (Byte)	Value	Description
bNumHID_Interfaces	1	Fixed	Number of interfaces
bInterfaceNo	1	Fixed	Interface number
bNumInitReports	1	xxh	Total number of Reports (max. 32)
bReserve	1	Fixed	reserved
bReportType	1	xxh	Report type 00h: reserved 01h: Input Report 02h: Output Report 03h: Feature Report 04h to FFh: reserved
bReportID	1	xxh	Report ID number 00h: Not used 01h to FFh: ID number
wReportLen	2	xxxxh	Size of Report data (max. 257 bytes) 1 to 257 bytes (0001h to 0101h)

5.3.7 Host connection

If “HID START” (10h) is set to Start, the LSI automatically executes enumeration as soon as it is connected to the Host. Once the HID Class control is possible, the LSI issues notification of this state to the Main CPU with an event. If “HID START” (10h) is set to Stop, the LSI cannot connect to the Host.

Make the Main CPU execute Host connection processing, referring to the flow shown in Fig. 5-22.

“The LSI: Event notification” depends on the Host, and “USB Suspend Status” event may be generated several times, followed by the issuance of notification of “CD Change” event.

An event is generated when the LSI receives protocol mode settings from the Host. Change the protocol mode by making the Main CPU process the “Main CPU: GET PROTOCOL MODE (25h) acquisition”.

For information on transfer rates at the time of connection, refer to Section “6.1 Transfer System”.

If the system is configured as a BUS-powered device, the processing described in Section “5.3.3 Device initialization flow” must be executed within the specified timeframe during the connection of the Host.

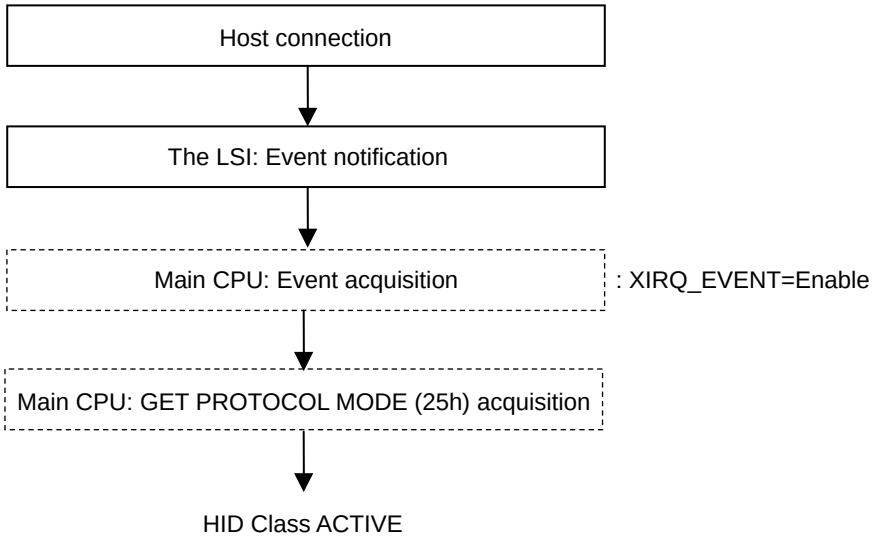


Fig. 5-22 Connection flow

5.3.8 Host disconnection

The disconnect processing performed when the LSI is disconnected from the Host depends on the immediately preceding state. Refer to the following sections in this manual. Note that the operations performed are the same as for disconnection, even when “HID START” (10h) is set to Stop.

5.3.8.1 Disconnect flow

The LSI performs USB disconnect processing when disconnected from the Host. The Main CPU is notified of the USB disconnect status by an event (Bit 1 CD Change).

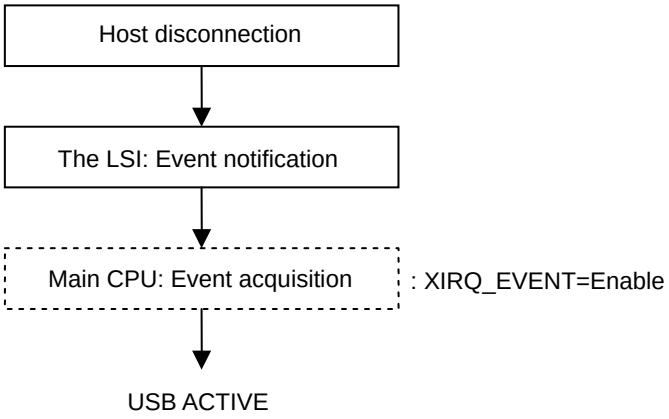


Fig. 5-23 Disconnect flow

5.3.8.2 Disconnect flow during data reads

The LSI performs USB disconnect processing when disconnected from the Host while Report data is being read from the LSI by the Main CPU. When “Bit 3 Rcv Feature Report” or “Bit 2 Rcv Output Report” of event information is 01b, the Main CPU should read the remaining Report data in the LSI using “RECV FEATURE REPORT (21h) or RECV REPORT (23h) acquisition”.

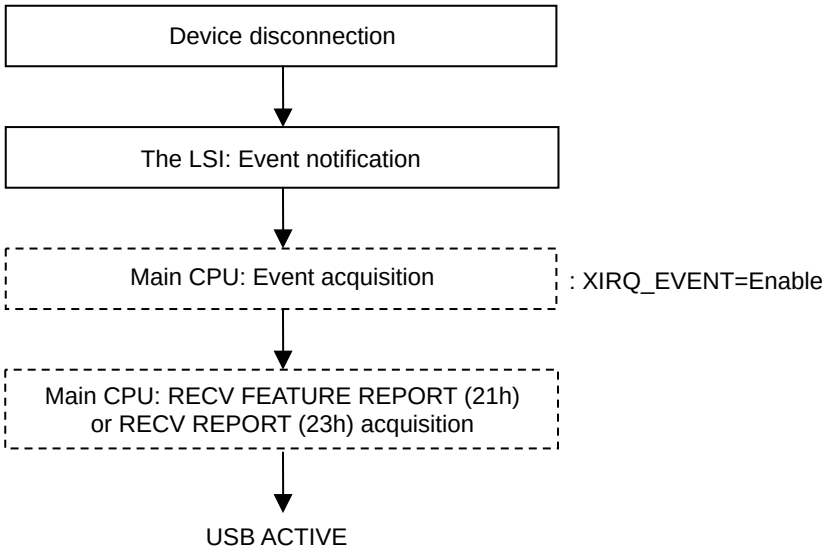


Fig. 5-24 Disconnect flow during data reads

5.3.8.3 Disconnect flow during Sleep

If the LSI is disconnected from the Host during the Sleep state, notification is provided via the XIRQ_EVENT pin. The Main CPU should set the WAKEUP pin to High. The LSI performs USB disconnect processing and notifies the Main CPU that the USB has been disconnected with an event (Bit 1 CD Change). For information on the WAKEUP pin, refer to Section “5.4.3 WAKEUP”.

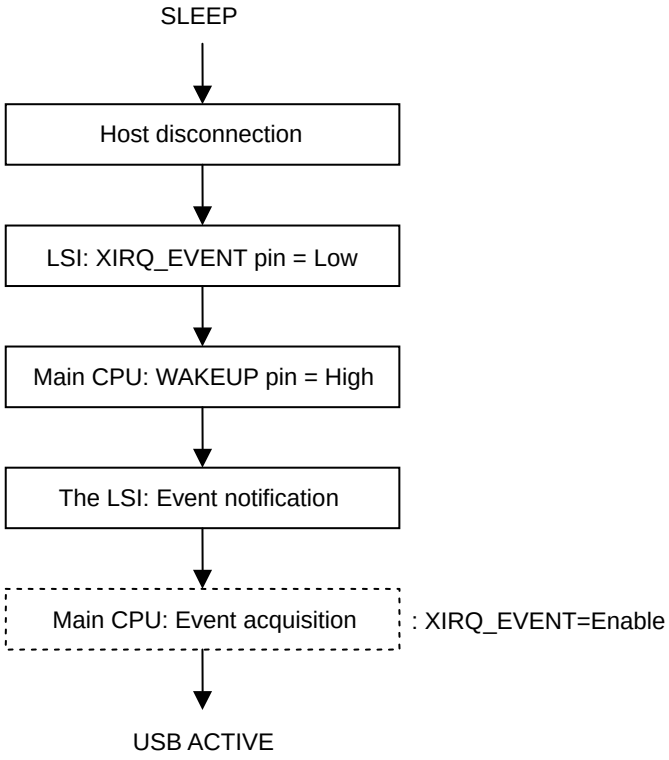


Fig. 5-25 Disconnect flow in Sleep

5.3.9 Device power management

The LSI is equipped with two types of power management functions. One manages the LSI using the SLEEP control, while the other manages the USB using the Suspend control. Note that commands cannot be written when the LSI is in the SLEEP state. The LSI must be switched to Active using the WAKEUP pin. Each control method is described below.

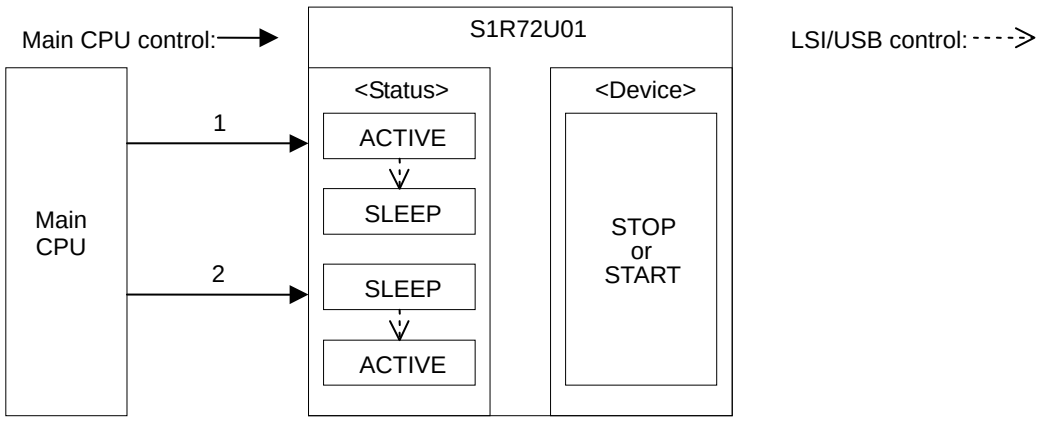
If the Main CPU requires the LSI to be in the SLEEP state while the LSI remains connected to the Host, "HID START" (10h) must be set to Stop.

5.3.9.1 Control when Host is not connected

Table 5-27 shows the LSI power management method with no Host connected. A lower power consumption state can be set with this control even if the Device is set to Stop or Start by "HID START" (10h).

Table 5-27 Control of LSI with no Host connected

Step	Control target	Control method
1	EIreq	Execute "SLEEP" (01h). When the command is recognized, the status changes from ACTIVE to SLEEP. The SIO_READY pin changes from High to Low.
2	WAKEUP pin	Set the WAKEUP pin from Low to High. The status changes from SLEEP to ACTIVE. The SIO_READY pin changes from Low to High.



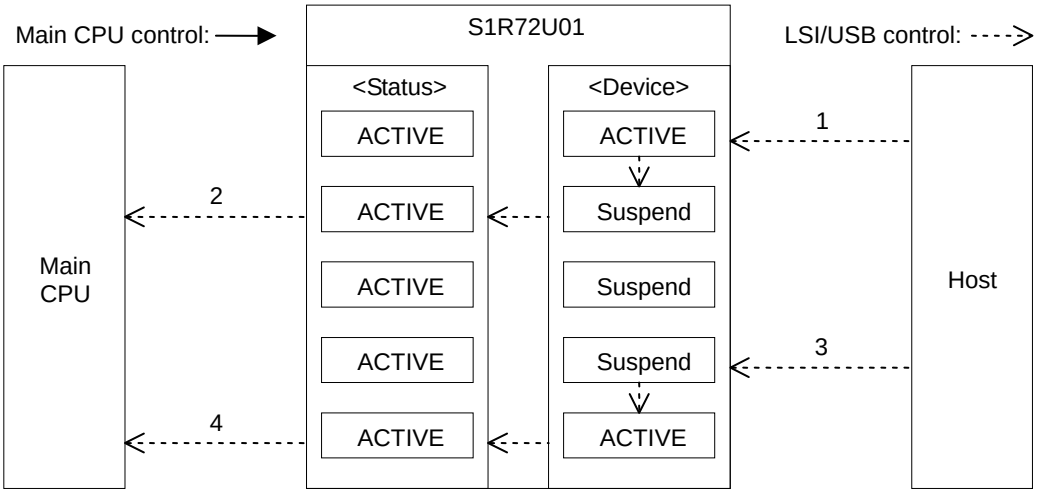
5.3.9.2 Suspend control when connected to Host

Table 5-28 shows the Device power management method with the Host connected. This method provides control with Remote Wakeup set to Prohibit by the Host. If the Main CPU executes a data write during the Suspend state, the LSI discards the data and issues error notification to the Main CPU. Data transfers are not possible until the Host is set to Resume.

If the system is configured as a BUS-powered device, the VBUS consumed current must be set to 2.5 mA or less within 10 ms of the Suspend state in Step 1. Since setting the LSI to the SLEEP state to reduce the current consumed requires the processing described in Section "5.3.9.3 SLEEP control when connected to Host", the power management described in this section cannot be used.

Table 5-28 Control of Suspend status

Step	Control target	Control method
1	Suspend	"Suspend" from the Host is detected. The Device status changes from ACTIVE to Suspend.
2	Event information	Notification of event generation is issued to the Main CPU.
3	Resume	"Resume" from the Host is detected.
4	Event information	The Device status changes from Suspend to ACTIVE. Notification of event generation is issued to the Main CPU.



5.3.9.3 SLEEP control when connected to Host

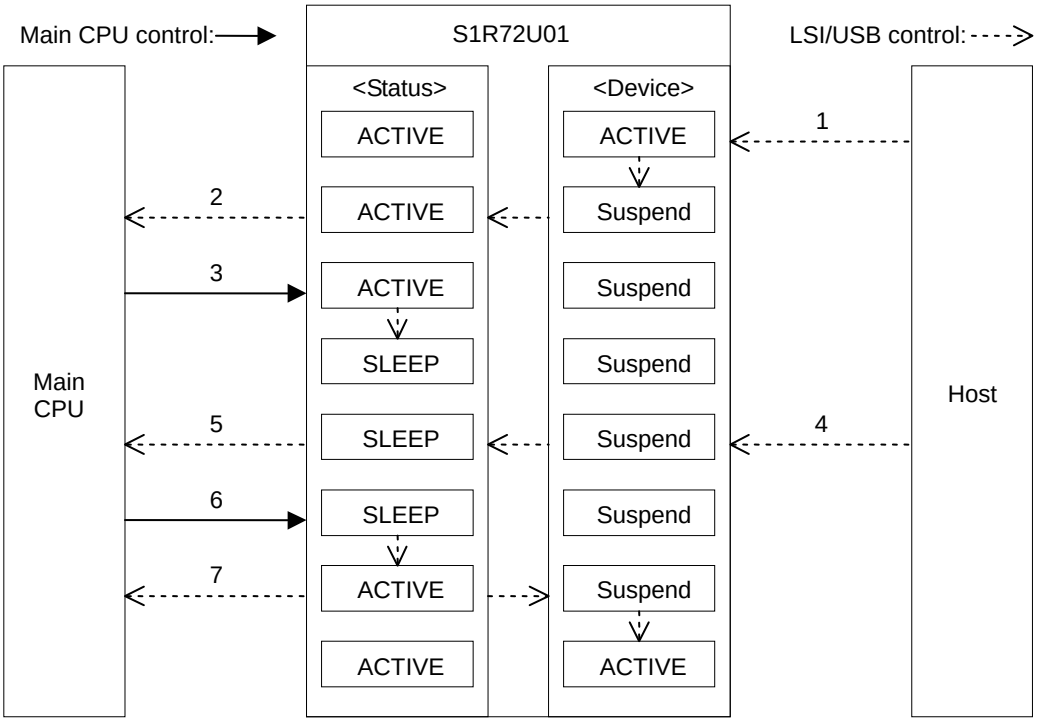
Table 5-29 shows the LSI power management method with the Host connected. This method provides control with Remote Wakeup set to Prohibit by the Host. If a data write is executed by the Main CPU while in the Suspend state, the LSI discards the data and issues an error notification to the Main CPU. Data transfers are not possible until the Host is set to Resume.

If the system is configured as a BUS-powered device, execute "SLEEP" (01h) as described in Step 3 within 8 ms of the event notification in Step 2.

Set the WAKEUP pin to High as described in Step 6 within 5 ms after the XIRQ_EVENT pin is asserted in Step 5.

Table 5-29 Control of SLEEP state

Step	Control target	Control method
1	Suspend	"Suspend" from the Host is detected. The Device status changes from ACTIVE to Suspend.
2	Event information	Notification of event generation is issued to the Main CPU.
3	EIreq	Execute "SLEEP" (01h). When the command is recognized, the status changes from ACTIVE to SLEEP. The SIO_READY pin changes from High to Low.
4	Resume	"Resume" from the Host is detected.
5	XIRQ_EVENT pin	The XIRQ_EVENT pin changes from High to Low.
6	WAKEUP pin	Set the WAKEUP pin from Low to High. The status changes from SLEEP to ACTIVE. The SIO_READY pin changes from Low to High.
7	Event information	The Device status changes from Suspend to ACTIVE. Notification of event generation is issued to the Main CPU.



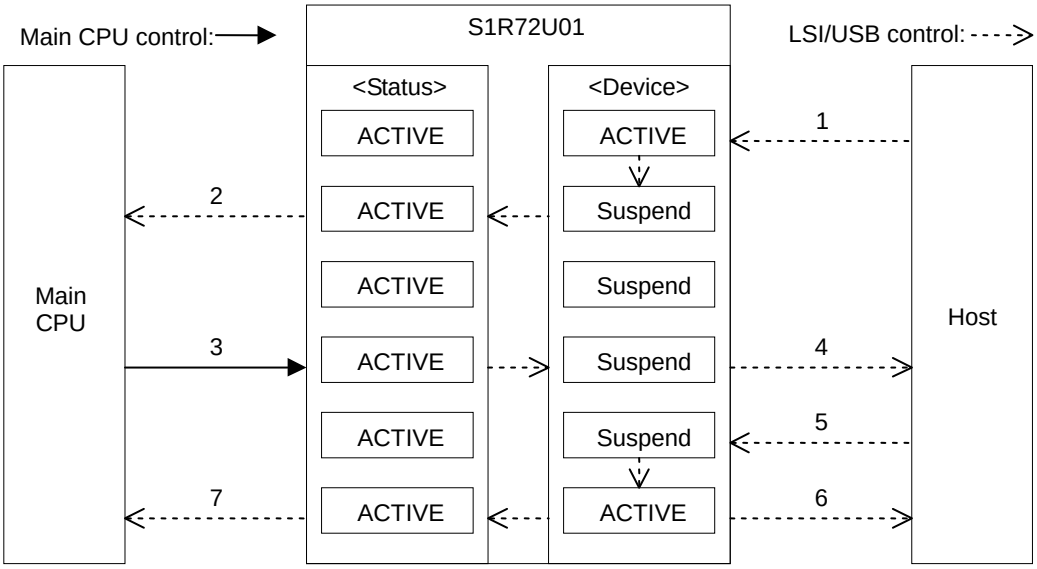
5.3.9.4 Suspend (Remote Wakeup) control when connected to Host

Table 5-30 shows the Device power management method with the Host connected. This method provides control with Remote Wakeup set to Permit by the Host. Data transfers are not possible until the Host is set to Resume.

If the system is configured as a BUS-powered device, the VBUS consumed current must be reduced to 2.5 mA or less within 10 ms of the Suspend state in Step 1. Since setting the LSI to the SLEEP state to reduce the consumed current requires the processing described in Section "5.3.9.5 SLEEP (Remote Wakeup) control when connected to Host", the power management described in this section cannot be used.

Table 5-30 Control of Remote Wakeup in Suspend state

Step	Control target	Control method
1	Suspend	"Suspend" from the Host is detected. The Device status changes from ACTIVE to Suspend.
2	Event information	Notification of event generation is issued to the Main CPU.
3	EIreq	Execute "SEND REPORT" (22h).
4	Remote Wakeup	Notification of "Remote Wakeup" is issued to the Host.
5	Resume	"Resume" from the Host is detected.
6	Data transmission	The Device status changes from Suspend to ACTIVE. Data is transmitted to the Host.
7	Event information	Notification of event generation is issued to the Main CPU.



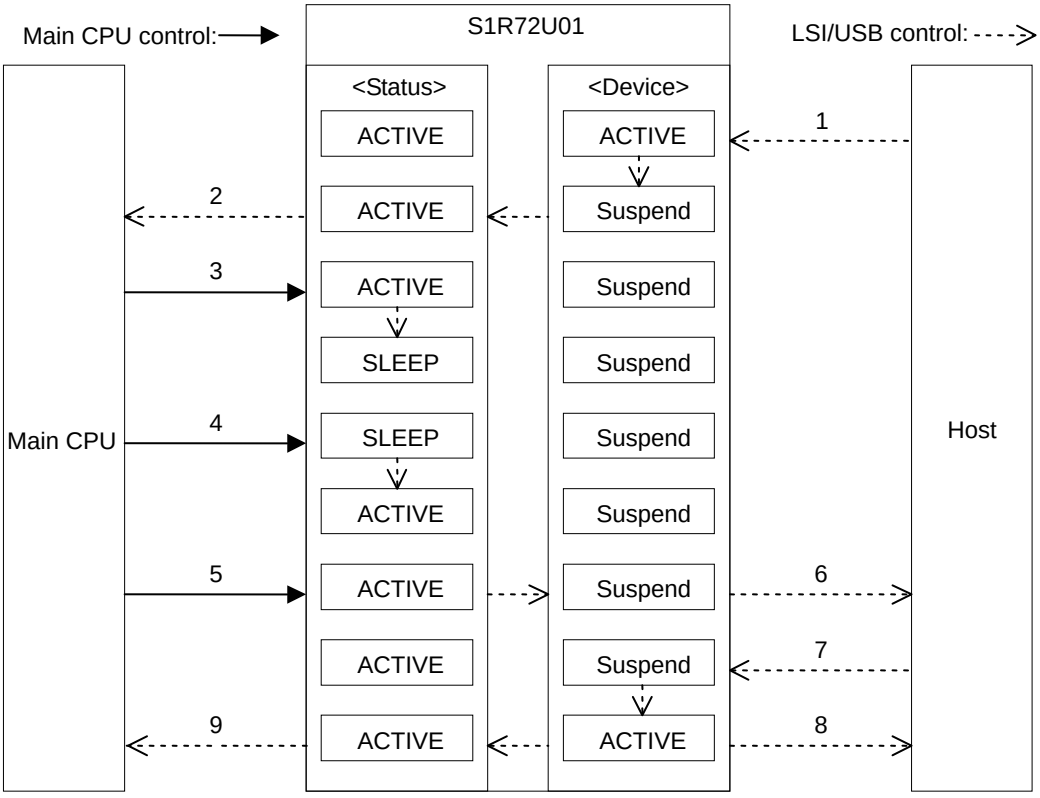
5.3.9.5 SLEEP (Remote Wakeup) control when connected to Host

Table 5-31 shows the LSI power management method with the Host connected. This method provides control with Remote Wakeup set to Permit by the Host. Data transfers are not possible until the Host is set to Resume.

If the system is configured as a BUS-powered device, execute "SLEEP" (01h) described in Step 3 within 8 ms of the event notification in Step 2.

Table 5-31 Control of Remote Wakeup in SLEEP state

Step	Control target	Control method
1	Suspend	"Suspend" from the Host is detected. The Device status changes from ACTIVE to Suspend.
2	Event information	Notification of event generation is issued to the Main CPU.
3	EIreq	Execute "SLEEP" (01h). When the command is recognized, the status changes from ACTIVE to SLEEP. The SIO_READY pin changes from High to Low.
4	WAKEUP pin	Set the WAKEUP pin from Low to High. The status changes from SLEEP to ACTIVE. The SIO_READY pin changes from Low to High.
5	EIreq	Execute "SEND REPORT" (22h).
6	Remote Wakeup	Notification of "Remote Wakeup" is issued to the Host.
7	Resume	"Resume" from the Host is detected.
8	Data transmission	The Device status changes from Suspend to ACTIVE. Data is transmitted to the Host.
9	Event information	Notification of event generation is issued to the Main CPU.



5.3.10 Precautions

(1) If the LSI is not recognized when connected to the Host or is recognized as an “unknown device”, there may be an error in the description of the descriptors, or the descriptions in the descriptors may not match the control setting (e.g., FS set by the descriptor but LS is specified by the “HID START”).

(2) The Host may trigger a Suspend state in Device control complying with the USB standard. If an event is generated due to a change of status to Suspend, the Main CPU must perform the processing described in Section “6.2 Time-related Regulations”.

(3) The USB standard stipulates use of Resume processing for resetting from Suspend state, but some Hosts may use BUS resetting. This involves reconnection using disconnection and then connection processing.

(4) Various cautions apply to the pins in Section “5.4 Setup Function”. Refer to the corresponding section of this manual.

5.4 Setting Function

The LSI uses the setting function to enter the initial settings and controlling operations. Refer to the following table for detailed information.

Table 5-32 Setting function

Pin	Content	Description
SPIxUART	Low : UART	Fix to Low.
HOSTxDEVICE	Low : DEVICE High : HOST	A hardware reset occurs if the setting changes while power is being supplied.
WAKEUP	Low → High:WAKEUP	Wakeup occurs at the rising edge of the change from Low to High.
INIT_BAUD	Low : 300 bps High : 9,600 bps	Changes in this setting are not possible while power is being supplied.

5.4.1 SPIxUART

Set to Low.

5.4.2 HOSTxDEVICE

Set the Host or Device of the USB function. If this setting changes while power is being supplied, a forced hardware reset is activated. Be careful to minimize the effects of external noise and other factors. For detailed information on the USB function, refer to Section “5.2 USB Host Function” or Section “5.3 USB Device Function”.

5.4.3 WAKEUP

This is used to change the status of the LSI from SLEEP to ACTIVE. Since this pin is activated by a rising edge, set to Low → High → Low, and keep at Low during normal use. If this pin is set to Low → High → Low while in the ACTIVE state and the LSI is set to SLEEP, the rising edge is retained; thus, the LSI status will not change to SLEEP. Be careful to minimize the effects of external noise and other factors.

For information on using this pin, refer to Section “5.2.9 Host power management” or Section “5.3.9 Device power management”.

5.4.4 INIT_BAUD

Set the initial baud rate for the UART to 9,600 or 300 bps.

5.5 Notification Function

The notification function notifies the Main CPU of the LSI status and the USB NSF (notification function enabled for USB Host only). Details are given below.

Table 5-33 Notification function

Pin	Content	Description
SIO_READY	Low : Not Ready High : Ready	Indicates the UART status.
XIRQ_STATUS	Low : Busy High : Idle	Indicates the LSI status.
XIRQ_EVENT	Low: EVENT High: Normal	Indicates the event status.
TPL	Low : Normal High : Error	NSF: Unsupported Device
ManyDev	Low : Normal High : Error	NSF: Too Many Devices
ManyHub	Low : Normal High : Error	NSF: Too Many Hubs
VBUS_Cur	Low : Normal High : Error	NSF: VBUS Over Current

5.5.1 SIO_READY

This switches to High when UART transfer is possible. It will be Low in states in which UART transfer is not possible, for example before a hardware reset or during the Sleep state.

5.5.2 XIRQ_STATUS

This changes to Low when the LSI status is Busy and changes to High when the status is Idle. For more information, refer to Section “5.1.6.6 Condition”.

5.5.3 XIRQ_EVENT

The LSI includes two types of events: protocol events and Sleep state events.

5.5.3.1 PROTOCOL EVENTS

PROTOCOL EVENTS are the events described in Section “7 Command Specifications”.

The output setting for this pin can be set to “enable” or “disable”. Setting “enable” causes a change to Low when an event occurs. A change to High results when the “GET EVENT” (F0h) EI request is written from the Main CPU. Refer to Section “5.1.5 Event control” for more information.

5.5.3.2 SLEEP EVENTS

Sleep state events result in a change to Low under the following conditions. A change to High results when the LSI receives a Wakeup.

USB Host

- When a Device is connected
- When a Device is disconnected
- When a Device remote wakeup has been detected

USB Device

- When connected to a Host
- When disconnected from a Host or when USB BUS reset or resume has been detected

5.5.4 TPL

If an unsupported Device is detected, notification of an “Error” is issued. If an unsupported Device is disconnected, “Normal” status is restored.

5.5.5 ManyDev

If two or more Devices are detected, notification of an “Error” is issued. If the number of connected Devices becomes one or less, “Normal” status is restored.

5.5.6 ManyHub

If two or more Devices with a built-in HUB are detected, notification of an “Error” is issued. When the number of connected Devices becomes one or less, “Normal” status is restored.

5.5.7 VBUS_Cur

If a VBUS overcurrent condition is detected, notification of an “Error” is issued. Refer to Section “5.2.8 VBUS control” for detailed information.

5.6 Development Support Functions

The LSI incorporates functions to confirm operating conditions and to support USB authentication and a download function. For detailed information, refer to the *S1R72U01 Development Support Manual*.

6. HID Class Overview

The LSI incorporates an HID Class Host and HID Class Device that complies with the HID Class (Human Interface Device Class) USB standard described in Section “2. Normative Standard”. The HID Class for the LSI is capable of controlling control devices such as keyboards and pointer devices. The specifications common to the Host and Device are outlined below.

6.1 Transfer System

The LSI’s HID Class supports the transfer modes and transfer rates shown in Table 6-1.

If the LSI serves as a Device, “LS” or “FS” can be selected.

If the LSI serves as a Host, it operates at the transfer rates of the connected Device. Note that an HS Device will be recognized (controlled) as FS when connected.

If the LSI serves as a Device, Report data size can be set below 257 bytes. In the HID Class, it transfers packet data at certain cycles based on polling interval settings. Thus, Report data is divided based on the packet size for transfer. For large Report data sizes, the Busy status persists until transmission/reception is completed. Set appropriate Report data size, packet size, and polling intervals.

Table 6-1 Transfer system

Transfer mode			Control transfer	Interrupt transfer
Application			Feature Report data	Report data
Transfer rate	FS	Speed	12 Mbps	
		Max. Packet Size	64 bytes (EP0) *1	64 bytes (IN, OUT) *2
	LS	Speed	1.5 Mbps	
		Max. Packet Size	8 bytes	

*1 EP0 refers to endpoint 0. EP0 is used for Control transfer.
*2 IN and OUT refer to the INPUT and OUTPUT endpoints used for Interrupt transfer.

6.2 Time-related Regulations

If the LSI serves as a Device, the Main CPU must complete processing within the following time limits.

(1) The WAKEUP pin must be set to High within 5 ms after the XIRQ_EVENT pin is asserted while the LSI is in the SLEEP state.

(2) If the system is configured as a BUS-powered device, the VBUS consumed current must be reduced to 2.5 mA or less when the Host sets it in the Suspend state. If the LSI’s consumed current is reduced, execute “SLEEP” (01h) within 8 ms of the “USB Suspend Status” event notification. For information on events, refer to Section “5.3.4 Device event information”.

6.3 Report ID

ID can be set to distinguish the Feature Report/Report data used in the HID Class. ID allows control of a keyboard and mouse by a single endpoint.

6.3.1 Data using Report ID

When Report ID is used, a 1-byte Report ID is added to the beginning of the data. The 1 byte for the Report ID must be included in the data size. For more information on data size, refer to Section “5.3.6.10 Report ID Registration Information”.

The Main CPU should add the 1-byte Report ID to the data to be written.

Data read by the Main CPU has a 1-byte Report ID added by the USB.

Fig. 6-1 shows data using Report ID.

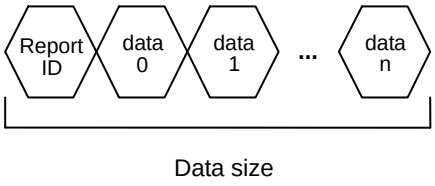


Fig. 6-1 Data using Report ID

6.3.2 Data not using Report ID

Only the data is used when no Report ID is used.

Fig. 6-2 shows data lacking a Report ID.

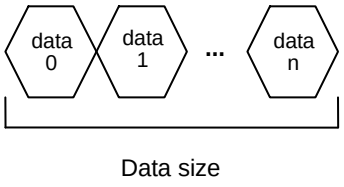


Fig. 6-2 Data not using Report ID

6.4 Protocol

6.4.1 Boot Protocol

Boot Protocol is a protocol specified by the USB standard. When Boot Protocol is used, “SEND REPORT” (22h) must be written or “RECV REPORT” (23h) must be read in the data format shown below.

Fig. 6-3 and Fig. 6-4 show the keyboard data format.

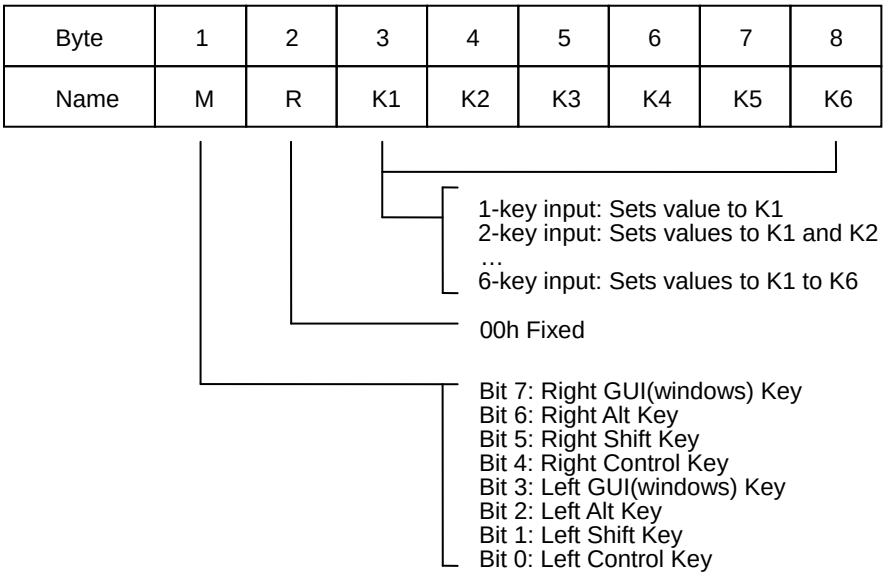


Fig. 6-3 Keyboard Input

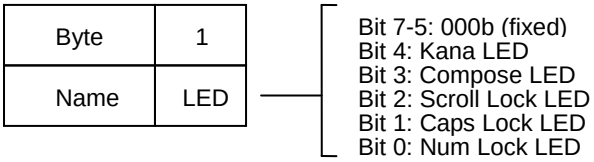


Fig. 6-4 Keyboard Output

Fig. 6-5 shows the mouse data format.

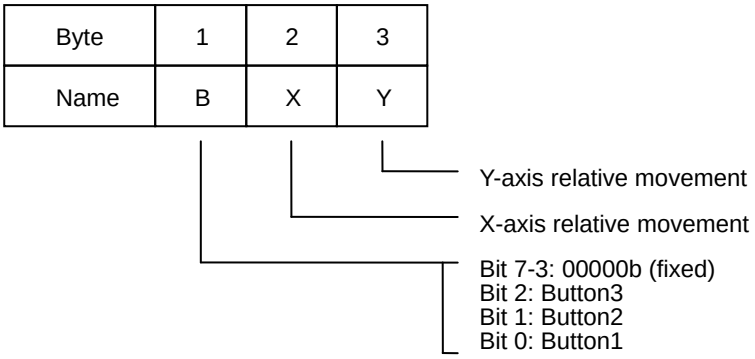


Fig. 6-5 Mouse Input

6.4.2 Protocol comparison

6.4.2.1 HID Cass Host

Table 6-2 illustrates the differences between Boot Protocol and Report Protocol for an HID Class Host.

Table 6-2 Protocol comparison

Details	Boot Protocol	Report Protocol
GET DESCRIPTOR (24h) 00h : HID Descriptor 01h : Report Descriptor	Not required	Required
Report Descriptor analysis *1	Not required	Required
REPORT ID REGISTRATION (11h)	Not required	Required
Data format	"6.4.1 Boot Protocol" format	Format specified by Report Descriptor
Supported HID Class Devices	Keyboard or Mouse *2	All HID Class Devices

*1 Analyze Report Descriptor and generate data to be registered by "REPORT ID REGISTRATION" (11h).

*2 Boot Protocol supports the following.
Keyboard: 84, 101, 104 keyboards
Mouse: 3-button mouse

6.4.2.2 HID Class Device

The HID Class Device protocol is the default Report Protocol setting. Boot Protocol is used when specified by protocol mode settings from the Host.

The Report Descriptor is required even when using Boot Protocol alone. The Report Descriptor can use Appendix-D.1, "Mouse Setting Examples" or Appendix-D.2, "Keyboard Setting Examples".

6.5 Other

(1) The USB standard requires a certification test to verify compliance with the standard. The USB logo mark cannot be placed on a product until the product passes this test. The LSI supports Hosts and Devices bearing the USB logo.

(2) The USB connector must comply with the USB standard. If the connector does not comply with the USB standard, signal quality cannot be guaranteed.

(3) Use a USB cable certified for compliance with the USB standard. If a cable without the USB logo is used, signal quality cannot be guaranteed.

(4) The LSI assumes that requirements such as USB cable length and circuit board wiring are met.

(5) Devices not supporting Boot Protocol are present.

7. Command Specifications

The LSI uses EI requests to implement control functions. Refer to Section “7.1.2 Abbreviations”, for the meaning of the codes used in the following sections.

7.1 EI requests

EI request (hereinafter referred to as “EIreq”) is defined as a control EI request and a USB (Host or Device) HID Class EI request. Commands should therefore be used to suit a particular function. EIreq is a command composed of three bytes—block size, Control code, and EIreq code—together with Information data (not present for certain EIreq commands).

The EIreq values (refer to individual EIreq for more information) issued by the Main CPU are added as an EI header (hereinafter referred to as “EIhead”) before the data and notification information (status information, event information, and error information) transferred by the LSI to the Main CPU. The Main CPU must confirm that the EIhead is a response to the EIreq.

7.1.1 Operating conditions

(1) EIreq cannot receive other EIreqs during processing.
If an error is generated during EIreq processing, the LSI transfers the error information after the processing ends.

(2) In some EIreqs, the protocol depends on the setting of the XIRQ_EVENT pin. For more information, refer to the explanation given for each EIreq.

(3) Unexpected data may be transferred if the settings are altered (e.g., from “enable” to “disable”) while using the XIRQ_EVENT pin. Settings should be changed when “HID START” (10h) is set to Stop.

(4) If a command other than the EIreqs specified below is written in error, an error or malfunction may occur. If an error occurs, check that the EIreq values have been correctly set before writing again. A hardware reset is required if an abnormal operation occurs.

(5) The LSI will issue an error if the Main CPU issues a read command (such as “GET DATA” (F5h) or “RECV REPORT” (23h)) when there is no data that can be transferred from the LSI.

(6) “Reserved” in command settings should be set to “0”.

(7) Events occurring must be processed first by the Main CPU. Executing other processes (e.g., writing EIreq not related to the event) after events have occurred may lead to unexpected results (e.g., data errors or malfunctions).

7.1.2 Abbreviations

The abbreviations in Table 7-1 are used in the discussion of EIreqs.

Table 7-1 List of abbreviations

Abbreviation	Full name
Tx	MOSI pin
Rx	MISO pin
xSTATUS	XIRQ_STATUS pin
xEVENT	XIRQ_EVENT pin
Stat	Status information
Event	Event information
Error	Error information

Abbreviation	Full name
size	Block size
code	Control code
cmd	EIreq code
Info	Information data *
D	Data *

* Write multiple Information data (example 1) or data (example 2) as shown below.

Example 1: If Information data is present from Block 3 to Block 5:
Block 3: Info0, Block 4: Info1, Block 5: Info2

Example 2: If multiple data is present:
D0 to Dn

7.2 Control EI requests

Table 7-2 shows Control EI requests. The following sections discuss command parameters and protocols.

Table 7-2 Control EI requests

Control code	EIreq code	EIreq name	Description
00h	01h	SLEEP	Sleep setting
	02h	DOWNLOAD	Download execution
	03h	LSI SETTING	LSI-specific setting
	F0h	GET EVENT	Event information acquisition
	F2h	GET STATUS	Status information acquisition
	F3h	ERROR	Error notification from the LSI
	F5h	GET DATA	Data read start request
	F8h	SERIAL PORT	Serial port setting
	FFh	EVENT INT CONTROL	Event interrupt setting

7.2.1 Operating conditions

(1) When the LSI detects an error, it issues an “ERROR” (F3h) to the Main CPU to indicate the error. This EIreq is read-only and cannot reacquire error information.

(2) The XIRQ_EVENT pin can be set to “enable” or “disable” (default) using “EVENT INT CONTROL” (FFh).

7.2.2 Precautions

(1) “GET DATA” (F5h) is regarded as an unused command by the Main CPU if the XIRQ_EVENT pin are set to “disable”.

(2) In the event management of the LSI, the occurrence of the same events may be detected, depending on the baud rate or read timing of the Main CPU.

7.2.3 01h_SLEEP

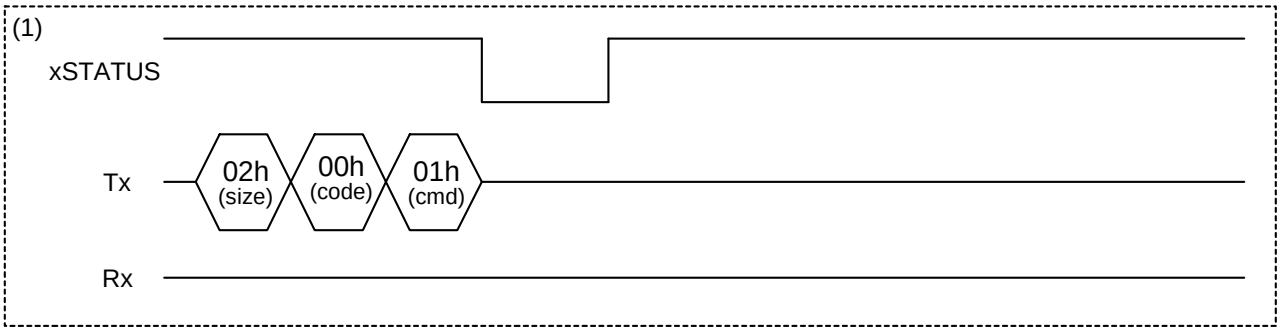
This EIreq sets the LSI to Sleep status. For more information on how to recover from the Sleep state, refer to "5.4.3 WAKEUP".

Table 7-3 SLEEP

Content		Block	Value	Description
EIreq	Block size	0	02h	Sum of Control code + EIreq code
	Control code	1	00h	Fixed value
	EIreq code	2	01h	Fixed value

Achieve access using protocol procedure (1).

The SIO_READY pin changes to Low when the LSI changes to the Sleep state.



7.2.4 02h_DOWNLOAD

This EIreq downloads a “TPL” or “descriptor” to the LSI. Write the download data after the EIreq. For more information on how to create download data, refer to Section “5.6 Development Support Functions”. For information on TPLs and descriptors, refer to Section “5.2.11 TPL” and Section “5.3.6 Device operation setting”.

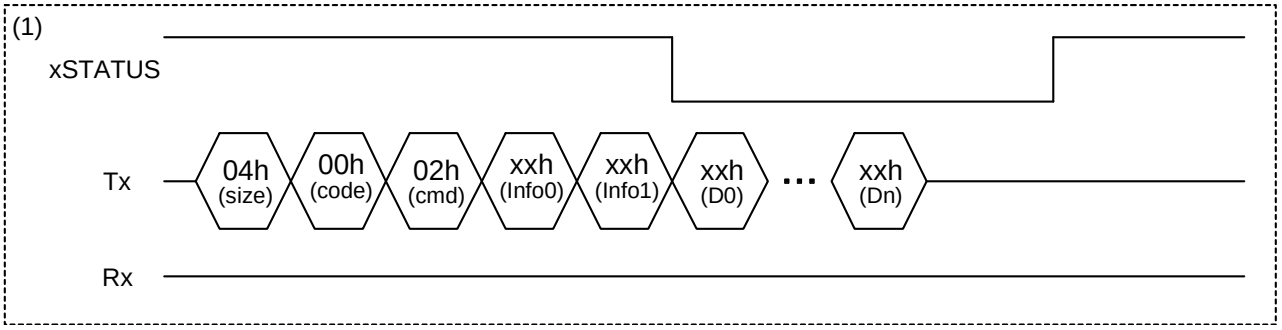
This EIreq is executable when “HID START” (10h) is set to Stop.

Table 7-4 DOWNLOAD

Content		Block	Value	Description
EIreq	Block size	0	04h	Sum of Control code + EIreq code + Information data
	Control code	1	00h	Fixed value
	EIreq code	2	02h	Fixed value
	Information data	3	xxxxh	Size of download data 0001h to FFFFh (Block 4: MSB, Block 3: LSB)
		4		
Data		-	xxh	Download data

Achieve access using protocol procedure (1).

The LSI asserts xSTATUS when it recognizes a command. When the data write from the Main CPU is completed and internal processing ends, the xSTATUS is negated.



7.2.5 03h_LSI SETTING

This EIreq sets the CLKOUT pin of the LSI and the VBUS detection in Host operation. For more information on the CLKOUT pin, refer to *S1R72U01 Data Sheet*. For more information on VBUS detection, refer to Section "5.2.8 VBUS control".

Table 7-5 LSI SETTING

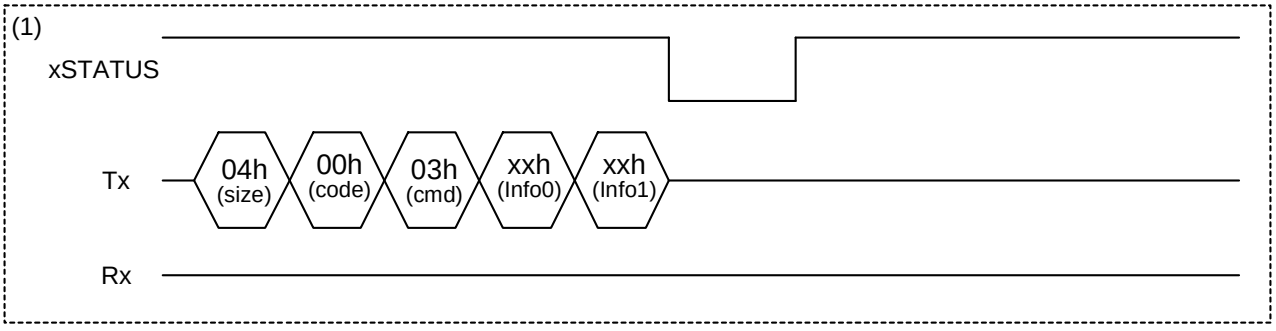
Content		Block	Value	Description
EIreq	Block size	0	04h	Sum of Control code + EIreq code + Information data
	Control code	1	00h	Fixed value
	EIreq code	2	03h	Fixed value
	Information data	3	xxh	Clock output setting

Bit	Content	Setting value
7	Output setting	0b: Output prohibited 1b: Output permitted (default)
6-4	reserved	
3-0	Frequency setting (valid when Bit 7 = 1b)	0000b: 48MHz (default) 0001b: 24MHz 0010b: 12MHz 0100b: 6MHz 1000b: 3MHz

		4	xxh	VBUS setting

Bit	Content	Setting value
7	Over Current detection setting	0b: Detection prohibited 1b: Detection permitted (default)
6-0	reserved	

Achieve access using protocol procedure (1).



7.2.6 F0h_GET EVENT

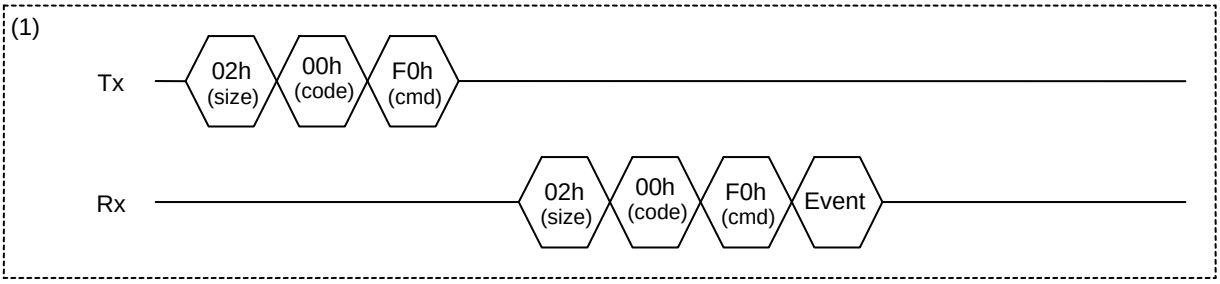
Used to acquire event information from the LSI or to send notification of an event occurrence to the Main CPU. “Event information” is added when transferred from the LSI.

Table 7-6 GET EVENT

Content		Block	Value	Description
EIreq	Block size	0	02h	Sum of Control code + EIreq code
	Control code	1	00h	Fixed value
	EIreq code	2	F0h	Fixed value
Event information		-	xxh	For more information, refer to Section “5.2.4 Host event information” or Section “5.3.4 Device event information”.

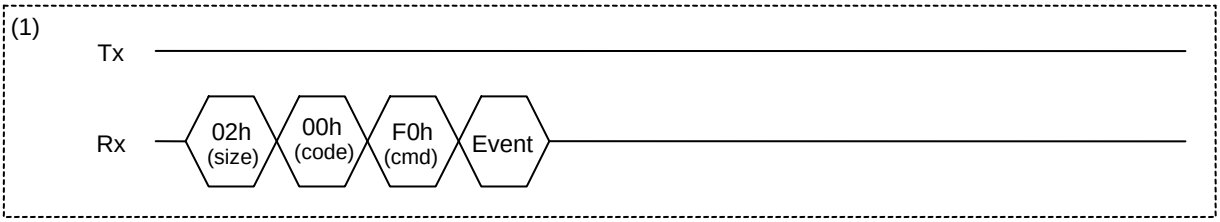
The protocol for acquiring event information from the LSI is shown below.

Access using the protocol procedure in (1).



The protocol for notifying the Main CPU of an event occurrence is shown below.

Transfer to the Main CPU uses the protocol procedure in (1). Note that the XIRQ_EVENT pin operates if the XIRQ_EVENT pin is set to “enable”. For more information, refer to the each EI request sections.



7.2.7 F2h_GET STATUS

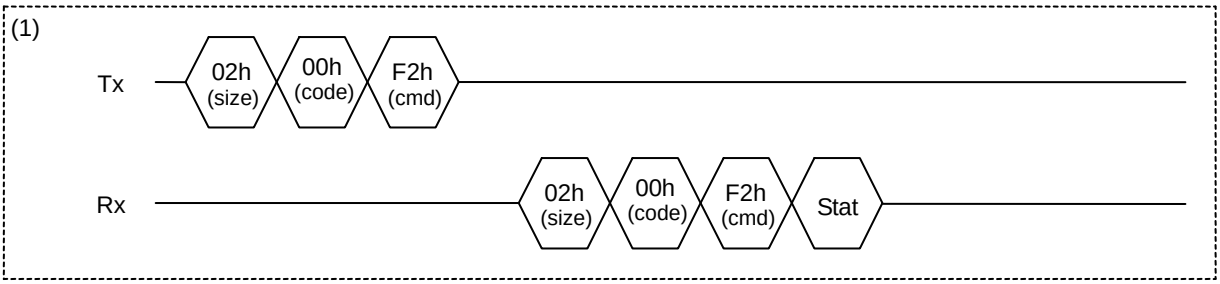
Used to acquire status information from the LSI or to send notification of a status change to the Main CPU. "Status information" is added when transferred from the LSI.

Table 7-7 GET STATUS

Content		Block	Value	Description
EIreq	Block size	0	02h	Sum of Control code + EIreq code
	Control code	1	00h	Fixed value
	EIreq code	2	F2h	Fixed value
Status information		-	xxh	For more information, refer to Section "5.1.6 Status information".

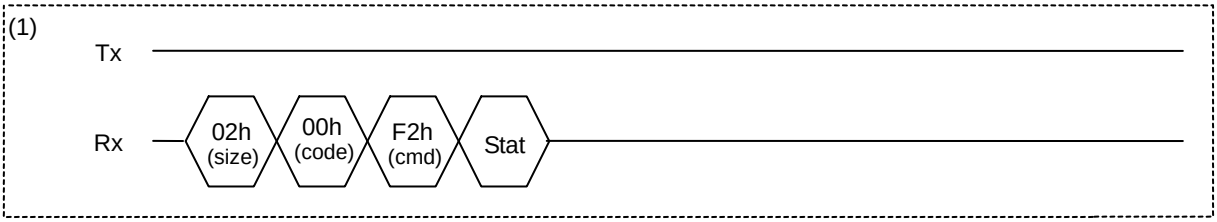
Shown below is the protocol for acquiring status information by the Main CPU.

Access using the protocol procedure in (1).



The protocol for notifying the Main CPU of a status change is shown below.

Transfer to the Main CPU uses the protocol procedure in (1).



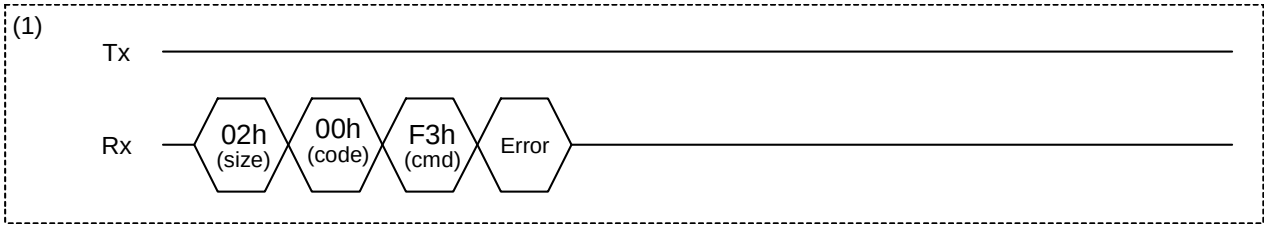
7.2.8 F3h_ERROR

This EIreq issues notification of errors to the Main CPU. Since this EI request is read-only, it cannot be used by the Main CPU. For more information, refer to Section “7.2.1 Operating conditions”.

Table 7-8 ERRORS

Content		Block	Value	Description
EIhead	Block size	0	02h	Sum of Control code + EIreq code
	Control code	1	00h	Fixed value
	EIreq code	2	F3h	Fixed value
Error information		-	xxh	For more information, refer to Section “5.2.5 Host error information” or Section “5.3.5 Device error information”.

Transfer data to the Main CPU using protocol procedure (1).



7.2.9 F5h_GET DATA

This EIreq is used when the XIRQ_EVENT pin is set to “enable”.

It reads data from the LSI. This EI request can be used instead of “RECV FEATURE REPORT” (21h) or “RECV REPORT” (23h). For more information on protocol procedures, refer to Section “7.3.8 21h_RECV FEATURE REPORT” or Section “7.3.10 23h_RECV REPORT”.

Table 7-9 GET DATA

Content		Block	Value	Description
EIreq	Block size	0	02h	Sum of Control code + EIreq code
	Control code	1	00h	Fixed value
	EIreq code	2	F5h	Fixed value

7.2.10 F8h_SERIAL PORT

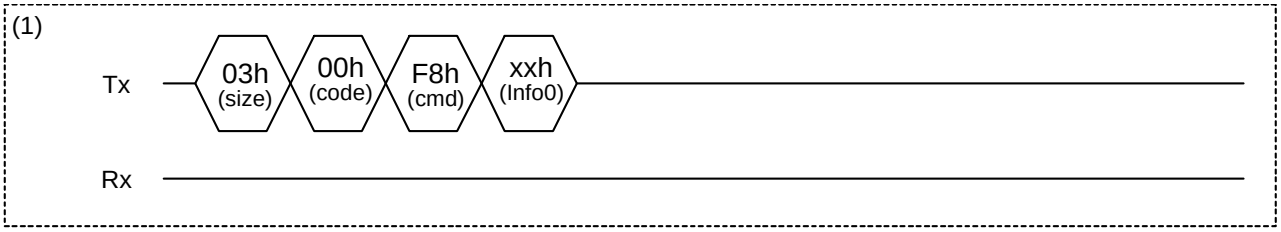
This EI request sets the LSI's serial port. When the LSI recognizes a command, it sets the SIO_READY pin to Low (for approximately 1 ms). When the setting takes effect, the pin changes to High.

Table 7-10 SERIAL PORT

Content		Block	Value	Description
EIreq	Block size	0	03h	Sum of Control code + EIreq code + Information data
	Control code	1	00h	Fixed value
	EIreq code	2	F8h	Fixed value
	Information data	3	xxh	Serial port setting

Bit	Content	Setting value
7-6	Parity	00b: None 01b: Odd number 10b: Even number 11b: None
5	Stop bit	0b: 1 bit 1b: 2 bits
4-0	Baud rate	00000b: 300 00001b: 600 00010b: 1,200 00011b: 2,400 00100b: 4,800 00101b: 9,600 00110b: 19,230.77 00111b: 38,461.54 01000b: 57,692.31 01001b: 115,384.62 01010b: 230,769.23 01011b: 300,000 01100b: 461,538.46 01101b: 600,000 01110b: 1,000,000 01111b: 1,200,000 10000b: 1,500,000 10001b: 2,000,000 10010b to 11111b: 3,000,000

Achieve access using protocol procedure (1).



7.2.11 FFh_EVENT INT CONTROL

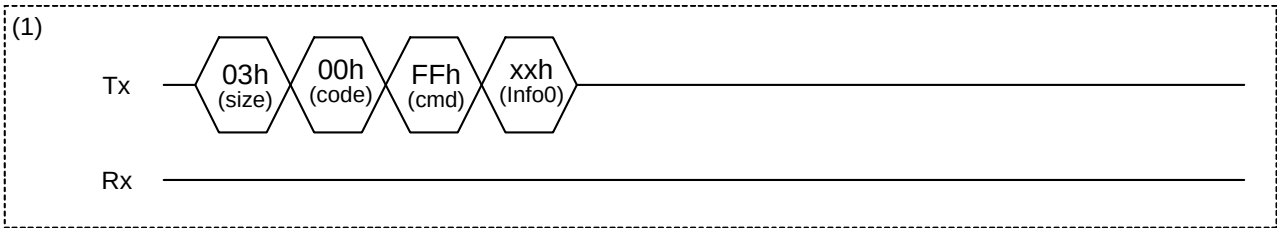
This sets the LSI XIRQ_EVENT pin to “enable” or “disable”. The default setting for the LSI is “disable”. The method for controlling event information depends on the XIRQ_EVENT pin setting. For more information, refer to Section “5.1.5 Event control”.

The XIRQ_EVENT pin changes in the Sleep state, regardless of the setting for this EIreq. Refer to Section “5.2.9.5 SLEEP (Remote Wakeup) control with Device connected” or Section “5.3.9.5 SLEEP (Remote Wakeup) control when connected to Host”.

Table 7-11 EVENT INT CONTROL

Content		Block	Value	Description
EIreq	Block size	0	03h	Sum of Control code + EIreq code + Information data
	Control code	1	00h	Fixed value
	EIreq code	2	FFh	Fixed value
	Information data	3	xxh	XIRQ_EVENT pin setting 00h: disable (default) 01h: enable 02h to FFh: Must not be set

Achieve access using protocol procedure (1).



7.3 HID Class EI requests

Table 7-12 shows HID Class EI requests. Command parameters and protocols are also explained in the sections that follow.

Table 7-12 HID Class EI requests

DEVICE		HOST		EIreq name	Description
Control code	EIreq code	Control code	EIreq code		
81h	10h	C1h	10h	HID START	HID Class control
	-		11h	REPORT ID REGISTRATION	Report ID registration information setting
	-		12h	DEVICE POWER MANAGEMENT	Power management
	-		13h	DEVICE RESET	USB BUS reset
	20h		20h	SEND FEATURE REPORT	Feature Report write
	21h		21h	RECV FEATURE REPORT	Feature Report read
	22h		22h	SEND REPORT	Report write
	23h		23h	RECV REPORT	Report read
	24h		-	INITIAL FEATURE REPORT	Feature Report initialization setting
	25h		-	GET PROTOCOL MODE	Protocol mode read
	-		24h	GET DESCRIPTOR	Descriptor read

7.3.1 Operating conditions

(1) HID Class EI requests are designed to be used after “HID START” (10h) has been started. The HID Class EI request that can be used while “HID START” (10h) is at Stop is “INITIAL FEATURE REPORT” (24h). Refer to Section “5.3.3 Device initialization flow”.

(2) EIreq cannot contain mixed data with differing Report IDs.

(3) Report data being written cannot be split into multiple EIreq.

(4) For information on Report settings, refer to Section “5.3.6.10 Report ID Registration Information”.

7.3.2 Precautions

(1) Use the appropriate Control code. The Control code depends on the USB function (Host/Device).

(2) When the LSI is operated as a Device and the XIRQ_EVENT pin is set to “disable”, “RECV FEATURE REPORT” (21h) and “RECV REPORT” (23h) become unused commands for the Main CPU, since they are sent from the LSI automatically. Likewise, when the LSI is operated as a Host, “RECV REPORT” (23h) becomes an unused command.

(3) “HID START” (10h) for the Host should use “Report Protocol HID Class Start” if connection to a standard HID Class Device is assumed. “Boot Protocol HID Class Start” should be used when it is necessary to limit to a Boot Protocol dedicated Device.

7.3.3 10h_HID START

This EI request sets the USB HID Class control. Table 7-13 shows Device settings. Table 7-14 gives Host settings.

Table 7-13 Device HID START

Content		Block	Value	Description
EIreq	Block size	0	03h	Sum of Control code + EIreq code + Information data
	Control code	1	81h	Fixed value
	EIreq code	2	10h	Fixed value
	Information data	3	xxh	Mode setting 00h: HID Class Stop (default) 01h: Low Speed HID Class Start 02h: Full Speed HID Class Start 03h to FFh: reserved

Table 7-14 Host HID START

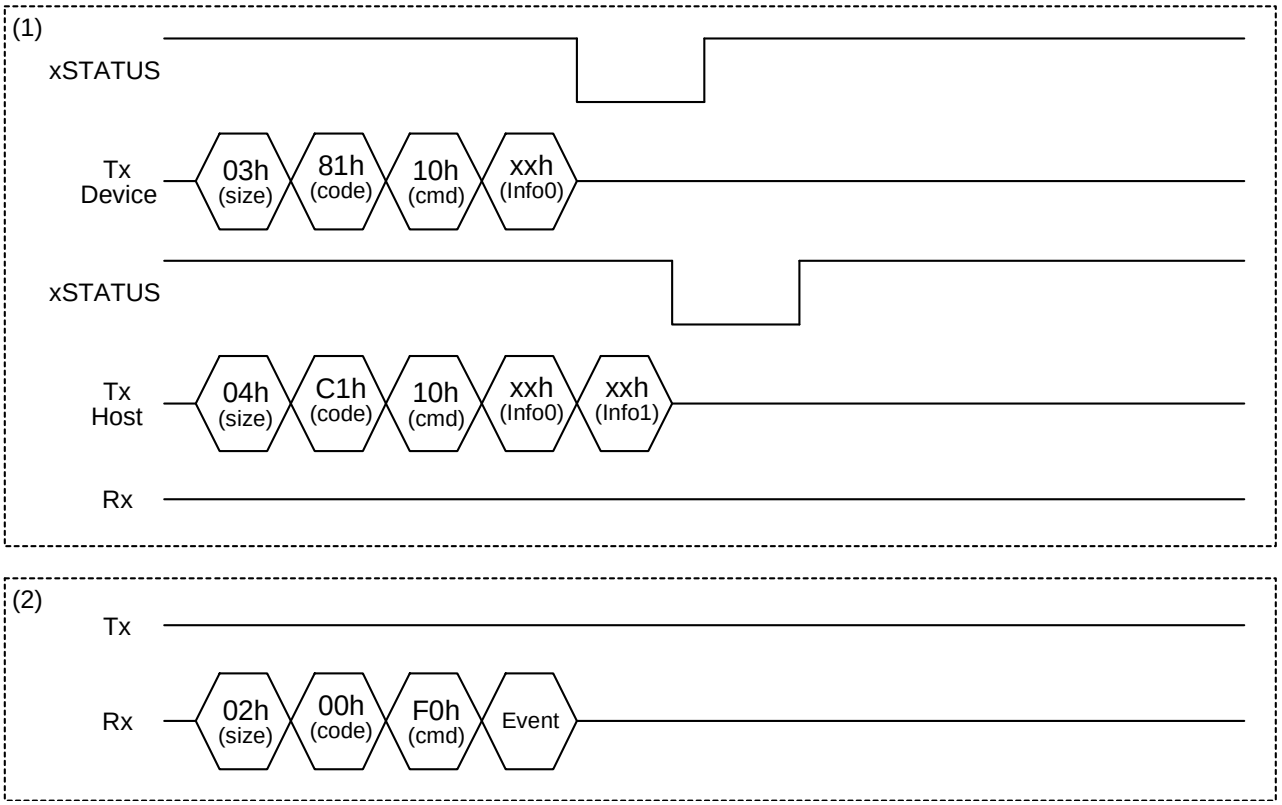
Content		Block	Value	Description
EIreq	Block size	0	04h	Sum of Control code + EIreq code + Information data
	Control code	1	C1h	Fixed value
	EIreq code	2	10h	Fixed value
	Information data	3	xxh	Protocol mode setting 00h: HID Class Stop (default) 01h: Report Protocol HID Class Start 02h: Boot Protocol HID Class Start 03h to FFh: reserved
		4	xxh	Interrupt frequency setting 00h: Responds only when change occurs 01h to FFh: Responds at specified frequency of between 4 and 1,020 ms

The following protocol is valid when the XIRQ_EVENT pin is set to “disable”.

Achieve access using protocol procedures (1) and (2).

When the LSI recognizes a command, xSTATUS in (1) changes.

When the connection to the Device or Host is completed, notification of the event information in (2) is issued to the Main CPU.



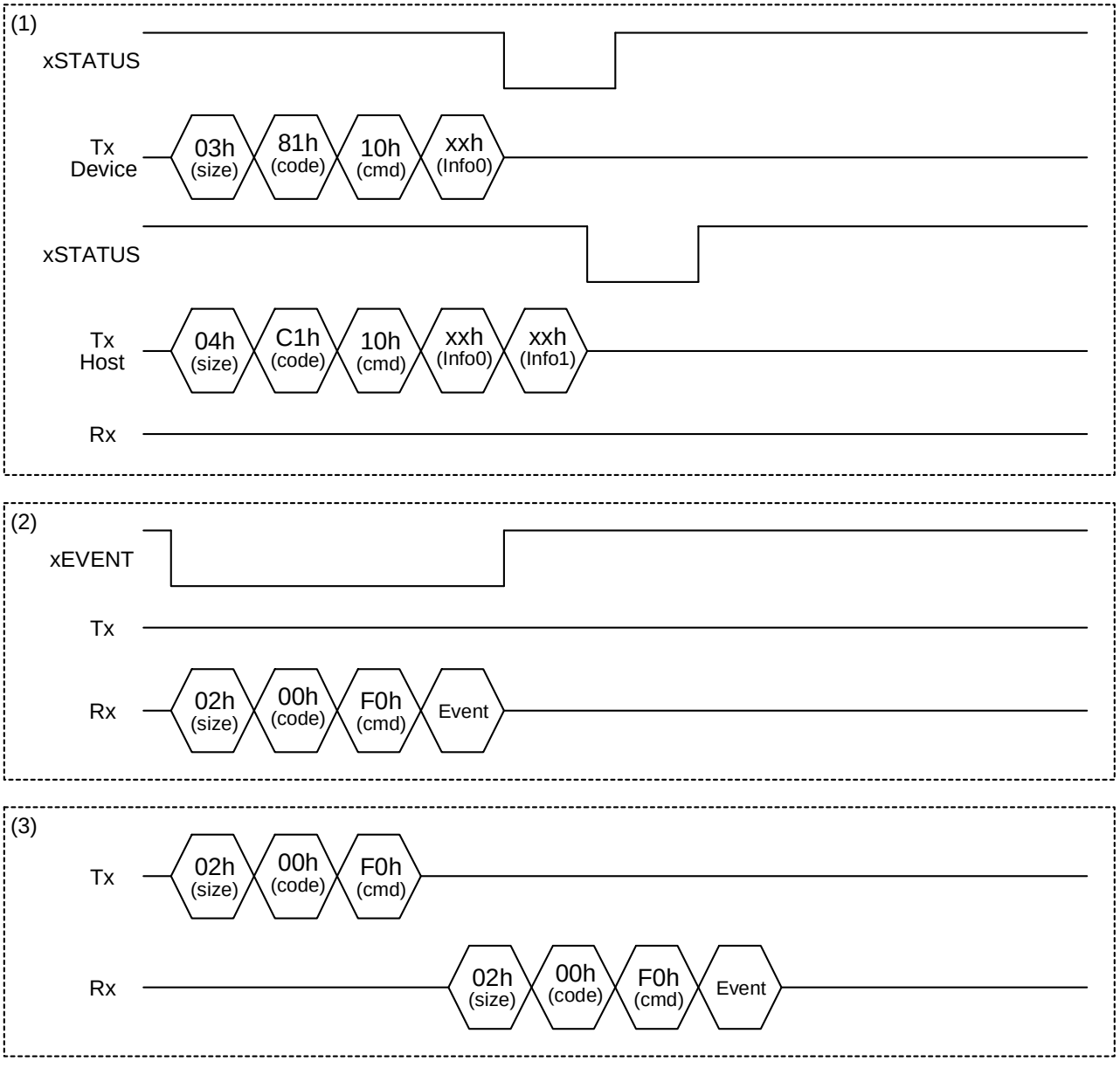
The following protocol is valid when the XIRQ_EVENT pin is set to “enable”.

Achieve access using protocol procedures (1) through (3).

When the LSI recognizes a command, xSTATUS in (1) changes.

When the connection to the Device or Host is completed, the event information is notified to the Main CPU concurrently with xEVENT assert in (2).

The event information is cleared when the read operation in (3) is executed.



7.3.4 11h_REPORT ID REGISTRATION

This EIreq sets the Report ID registration information obtained from the descriptor of the Device to the LSI. The Report ID registration information data should be written following the EIreq.

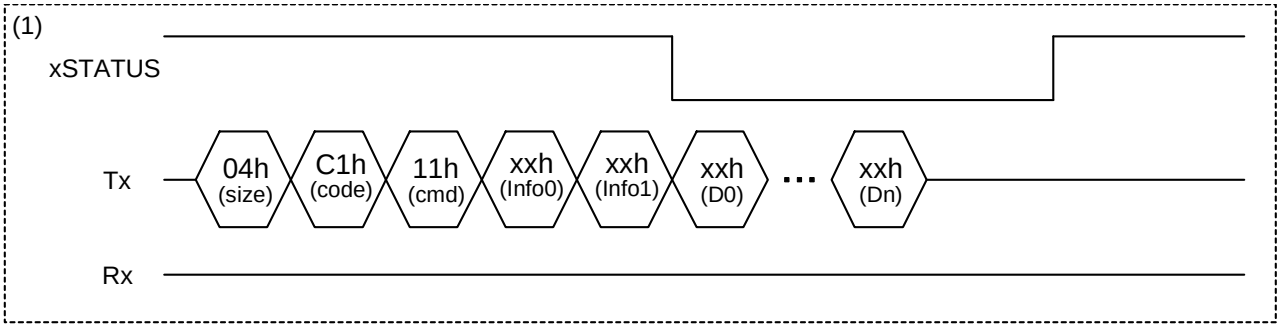
For more information on Report ID registration information, refer to Section “5.2.6.1 Connection flow”. The Report ID registration information data format will be as described in Section “5.3.6.10 Report ID Registration Information”.

Table 7-15 Host REPORT ID REGISTRATION

Content		Block	Value	Description
EIreq	Block size	0	04h	Sum of Control code + EIreq code + Information data
	Control code	1	C1h	Fixed value
	EIreq code	2	11h	Fixed value
	Information data	3	xxxxh	Size of Report ID registration information data 0008h to 0084h (Block 4: MSB, Block 3: LSB)
		4		
Data		-	xxh	Report ID registration information data

Achieve access using protocol procedures (1).

The LSI asserts xSTATUS when it recognizes a command. When the data write from the Main CPU is completed and internal processing ends, the xSTATUS is negated.



7.3.5 12h_DEVICE POWER MANAGEMENT

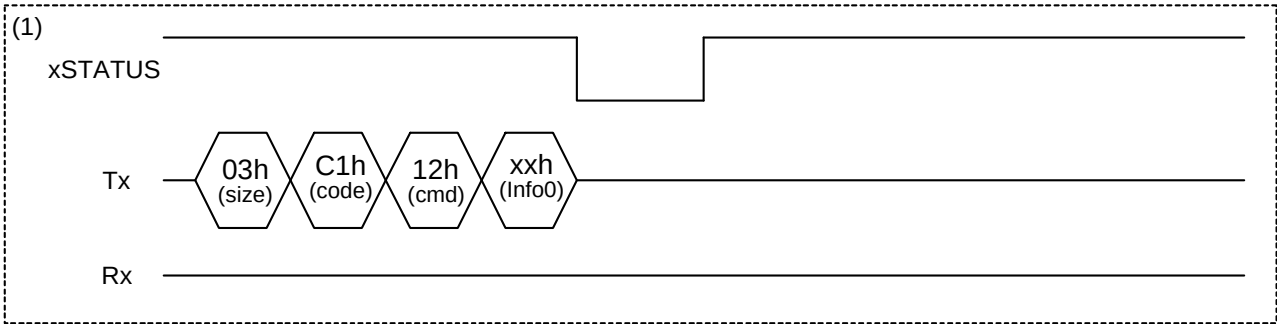
This EI request controls Device Suspend/Resume operations.

Table 7-16 Host DEVICE POWER MANAGEMENT

Content		Block	Value	Description
EIreq	Block size	0	03h	Sum of Control code + EIreq code + Information data
	Control code	1	C1h	Fixed value
	EIreq code	2	12h	Fixed value
	Information data	3	xxh	Operation setting 00h: Resume 01h: Suspend & Remote Wakeup prohibited 02h: Suspend & Remote Wakeup permitted 03h to FFh: reserved

Achieve access using protocol procedure (1).

xSTATUS in (1) is asserted while the LSI is processing a command.



7.3.6 13h_DEVICE RESET

This EI request performs a BUS reset for the Device in compliance with the USB standard.

Table 7-17 Host DEVICE RESET

Content		Block	Value	Description
EIreq	Block size	0	02h	Sum of Control code + EIreq code
	Control code	1	C1h	Fixed value
	EIreq code	2	13h	Fixed value

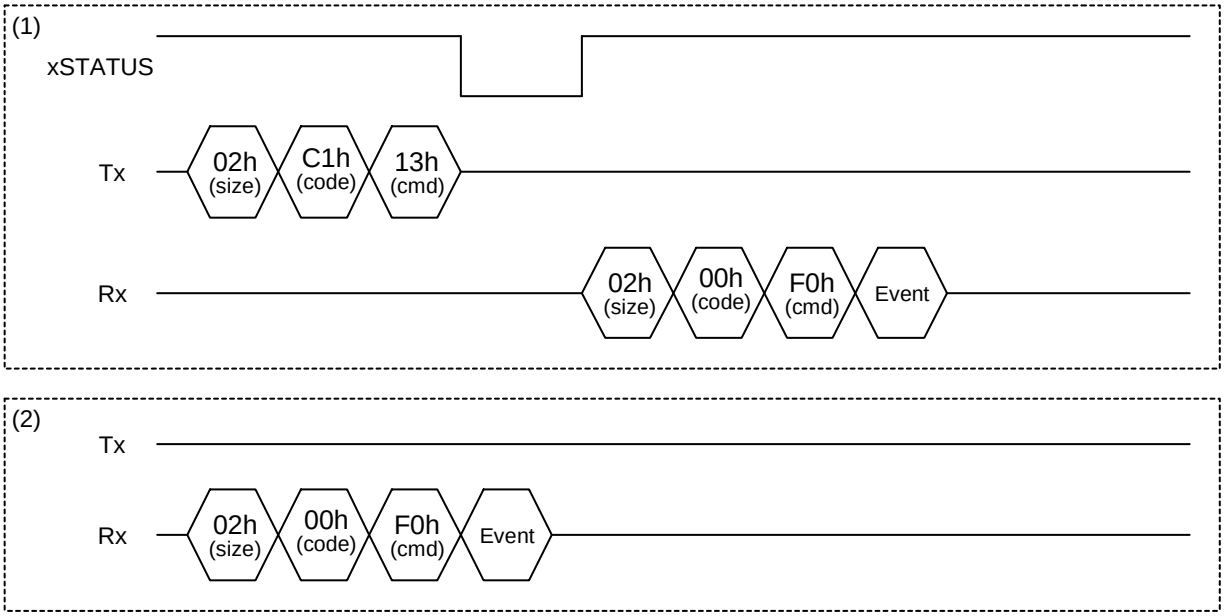
The following protocol is valid when the XIRQ_EVENT pin is set to “disable”.

Achieve access using protocol procedures (1) and (2).

xSTATUS in (1) is asserted while the LSI is processing a command.

When the Device is disconnected by a BUS reset, the event information in (1) is passed on to the Main CPU.

When the reconnection to the Device is completed, the event information in (2) is passed on to the Main CPU.



The following protocol is valid when the XIRQ_EVENT pin is set to “enable”.

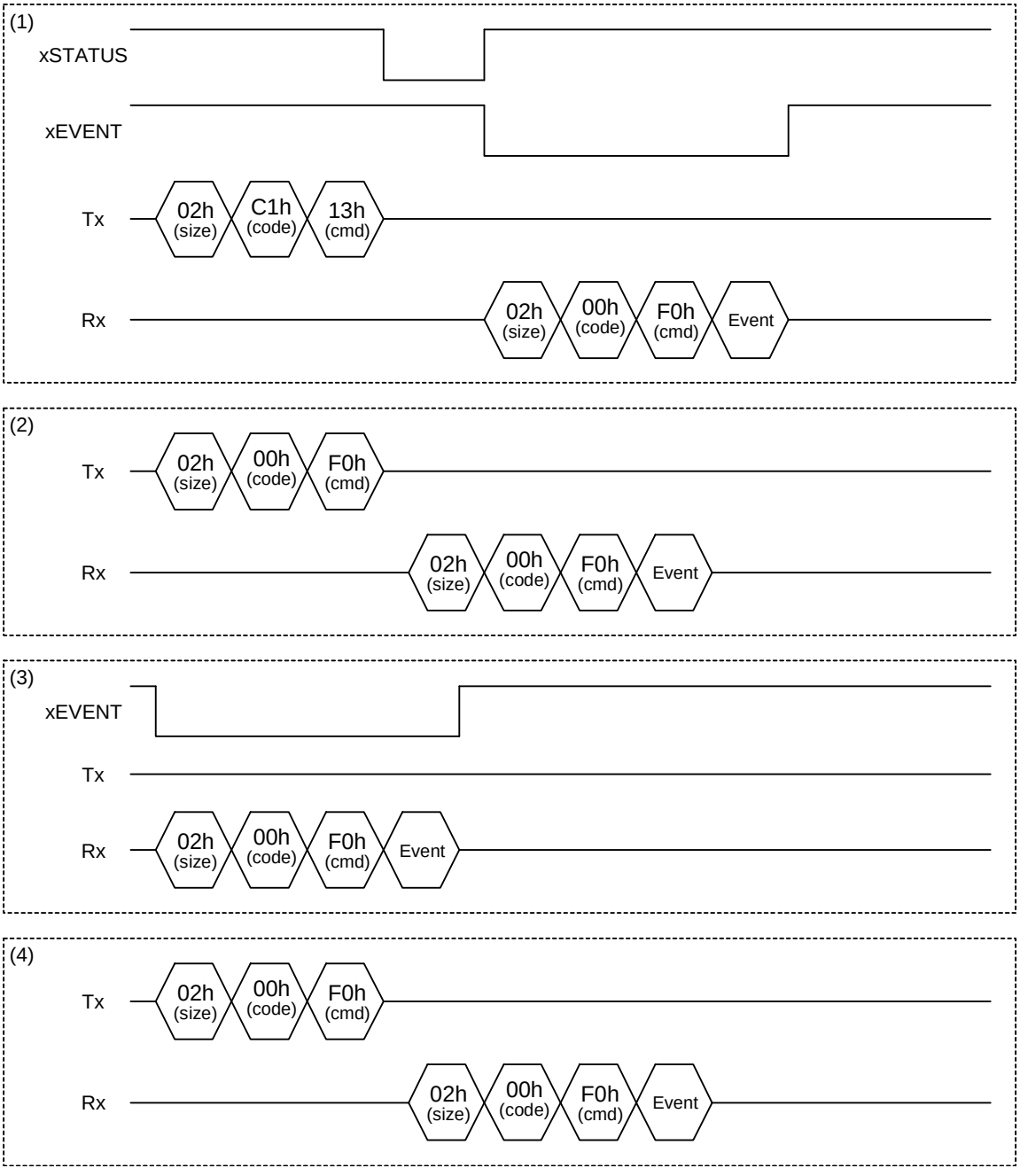
Achieve access using protocol procedures (1) through (4).

xSTATUS in (1) is asserted while the LSI is processing a command. When the Device is disconnected by a BUS reset, the event information is passed on to the Main CPU concurrently with xEVENT assert.

The event information is cleared when the read operation in (2) is executed.

When the reconnection to the Device is completed, the event information is passed on to the Main CPU concurrently with xEVENT assert in (3).

The event information is cleared when the read operation in (4) is executed.



7.3.7 20h_SEND FEATURE REPORT

This writes Feature Report data to the LSI. Table 7-18 shows the Device settings. Table 7-19 shows the Host settings. The Feature Report data should be written following the EIreq.

Set “Write data size” to the Feature Report data size. For information on data size, refer to Section “5.3.6.10 Report ID Registration Information”.

Table 7-18 Device SEND FEATURE REPORT

Content		Block	Value	Description
EIreq	Block size	0	04h	Sum of Control code + EIreq code + Information data
	Control code	1	81h	Fixed value
	EIreq code	2	20h	Fixed value
	Information data	3	xxxxh	Size of write data 0001h to 0101h (Block 4: MSB, Block 3: LSB)
		4		
Data		-	xxh	Feature Report data

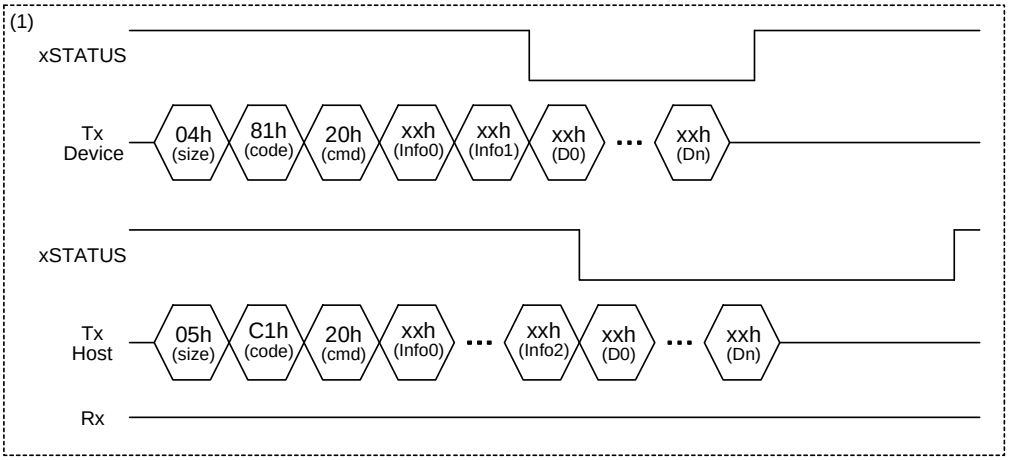
Table 7-19 Host SEND FEATURE REPORT

Content		Block	Value	Description
EIreq	Block size	0	05h	Sum of Control code + EIreq code + Information data
	Control code	1	C1h	Fixed value
	EIreq code	2	20h	Fixed value
	Information data	3	00h	Fixed value
		4	xxxxh	Size of write data 0001h to 0101h (Block 5: MSB, Block 4: LSB)
		5		
Data		-	xxh	Feature Report data

Achieve access using protocol procedure (1).

Based on the protocol for the Device, when the LSI recognizes a command, it asserts xSTATUS. When the data write from the Main CPU is completed and internal processing ends, the xSTATUS is negated. This status is maintained until the Host generates a request. The transmission timing depends on the request from the Host.

According to the protocol for the Host, when the LSI recognizes a command, it asserts xSTATUS. When transmission to the Device is completed, the xSTATUS is negated.



7.3.8 21h_RECV FEATURE REPORT

This reads Feature Report data from the LSI. Table 7-20 shows the Device settings. Table 7-21 shows the Host settings. The Feature Report data is added after EIhead when transferred from the LSI.

The LSI should set the EIhead “Information data” to the read data size. For information on data size, refer to Section “5.3.6.10 Report ID Registration Information”.

Table 7-20 Device RECV FEATURE REPORT

Content		Block	Value	Description
EIreq	Block size	0	04h	Sum of Control code + EIreq code + Information data
	Control code	1	81h	Fixed value
	EIreq code	2	21h	Fixed value
	Information data	3	0000h	Fixed value (Block4: MSB, Block3: LSB) * Indicates read data size for EIhead.
		4		
Data		-	xxh	Feature Report data

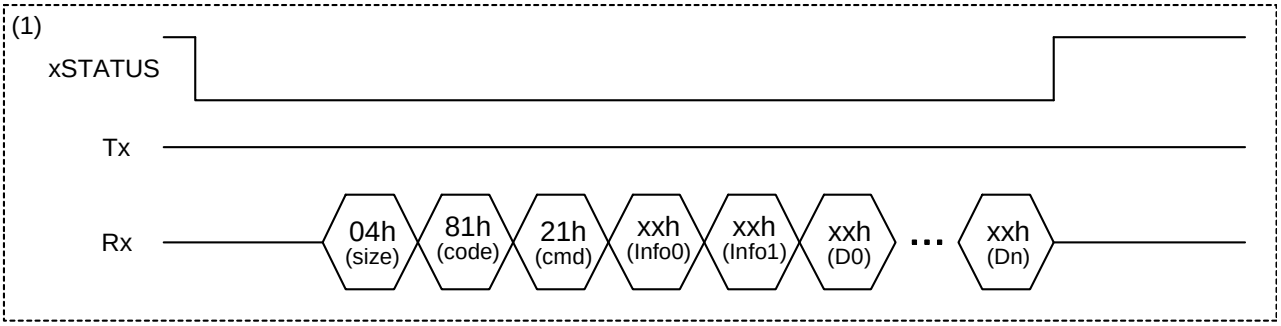
Table 7-21 Host RECV FEATURE REPORT

Content		Block	Value	Description
EIreq	Block size	0	05h	Sum of Control code + EIreq code + Information data
	Control code	1	C1h	Fixed value
	EIreq code	2	21h	Fixed value
	Information data	3	xxh	Report ID [00h: ID not used]
		4	0000h	Fixed value (Block5: MSB, Block4: LSB) * Indicates read data size for EIhead.
		5		
Data		-	xxh	Feature Report data

The following protocol is valid when the Device and XIRQ_EVENT pin are set to “disable”.

Achieve access using protocol procedure (1).

xSTATUS in (1) is asserted on reception of a Feature Report from the Host, after which the data is transferred. Read this data. After the data transfer is completed, the xSTATUS is negated.



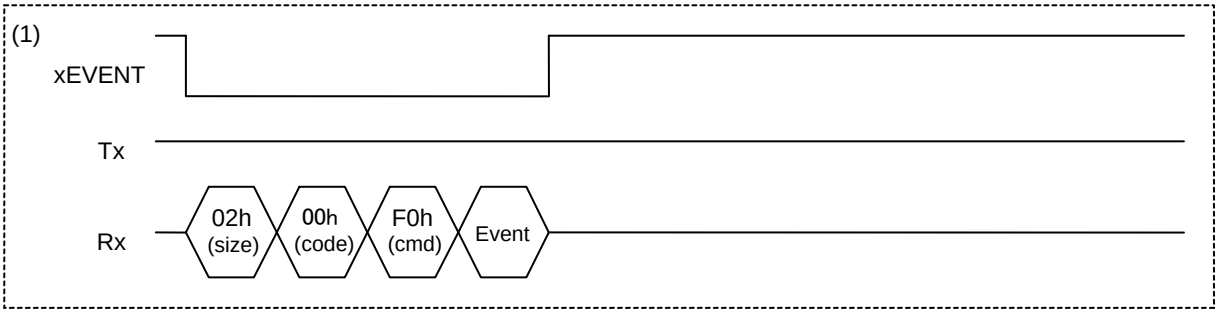
The following protocol is valid when the Device and XIRQ_EVENT pin are set to “enable”.

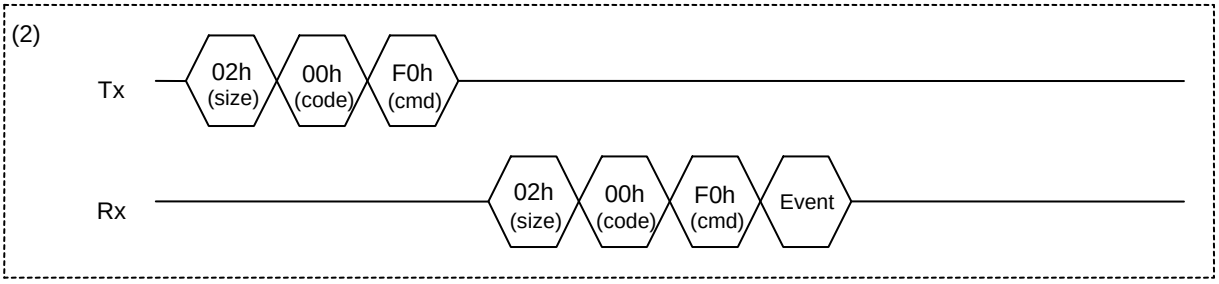
Achieve access using protocol procedures (1) through (4).

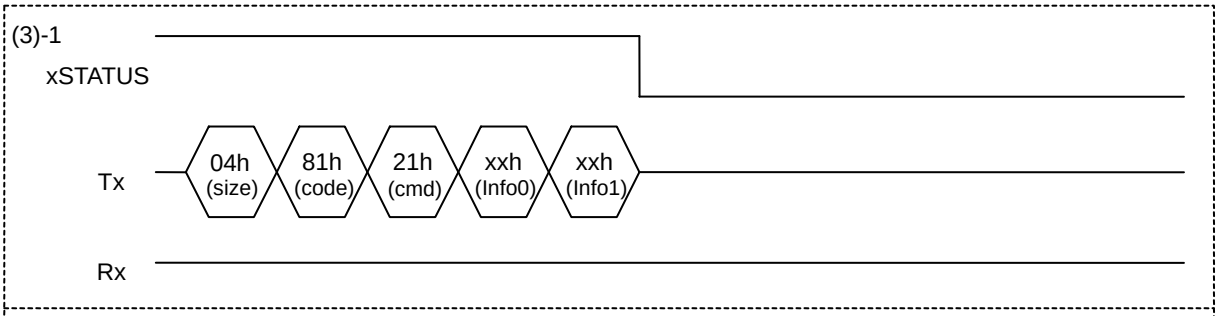
When a Feature Report is received from the Host, the event information is passed on to the Main CPU concurrently with xEVENT assert in (1).

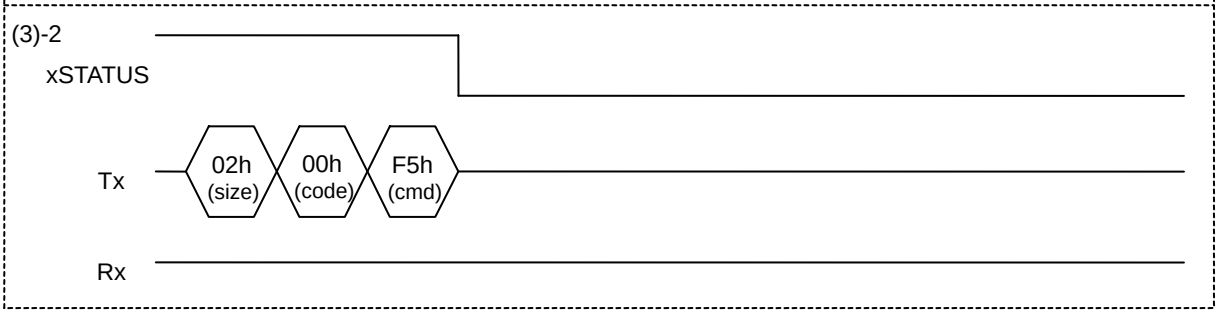
The event information is cleared when the read operation in (2) is executed.

This protocol allows either (3)-1 or (3)-2 commands. The LSI asserts xSTATUS when it recognizes a command.

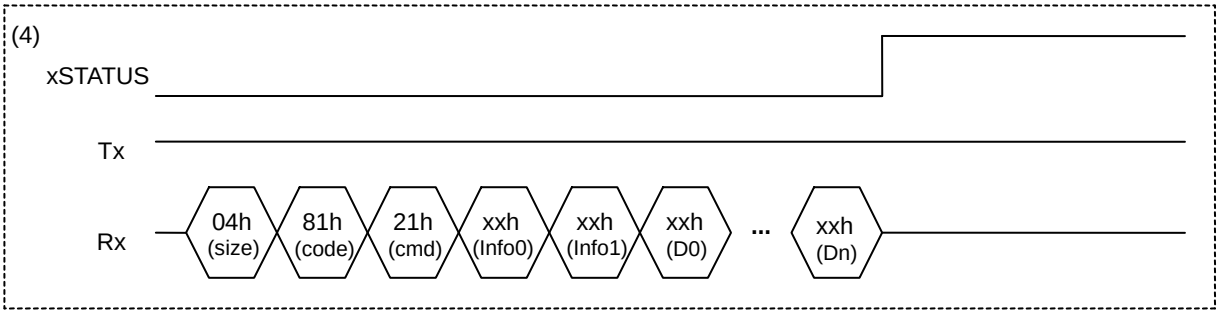








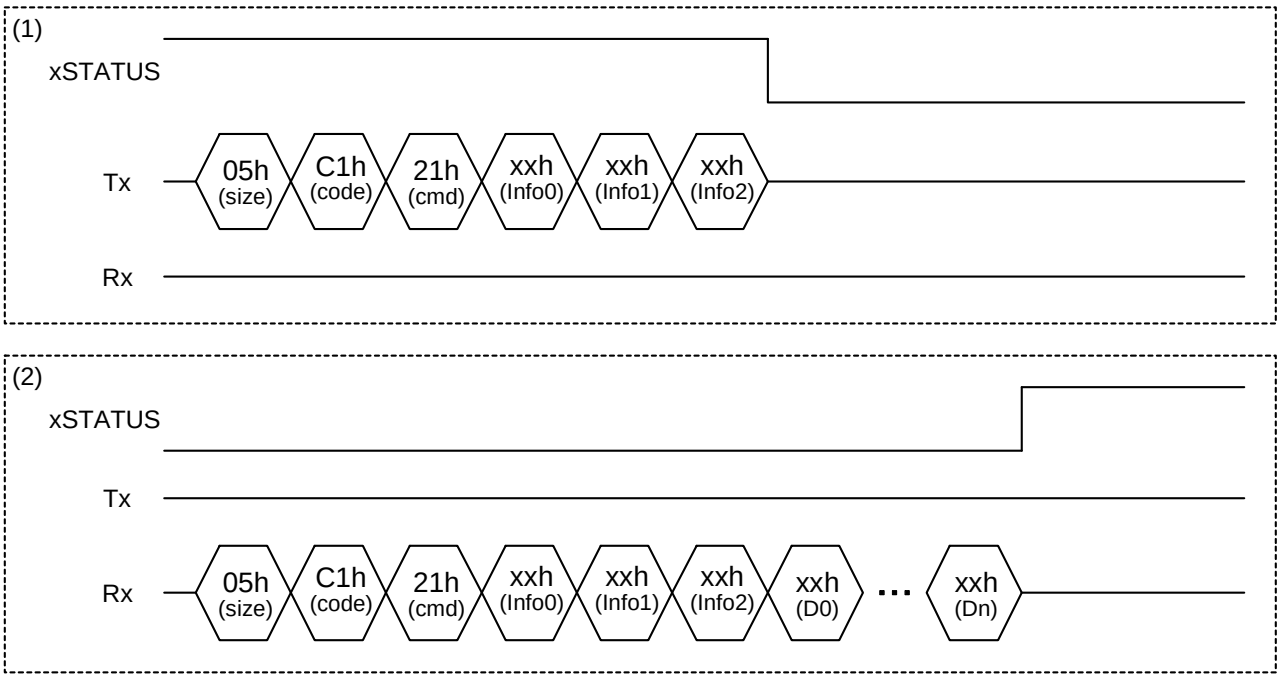
Data transfer in (4) starts. Read this data. When the data transfer is completed, the xSTATUS is negated.



The following is a protocol for the Host. Achieve access using protocol procedures (1) and (2).

The LSI asserts xSTATUS in (1) when it recognizes a command.

When the reception from the Device is completed, transfer of data in (2) starts. Read this data. When the data transfer is completed, the xSTATUS is negated.



7.3.9 22h_SEND REPORT

This EIreq writes Report data to the LSI. Table 7-22 shows Device settings. Table 7-23 gives Host settings. The Report data should be written following the EIreq.

This command allows multiple Report data to be written together. When Report data is combined, “Write data size” should be set to an integer multiple of the Report data size. For information on data size, refer to Section “5.3.6.10 Report ID Registration Information”.

Table 7-22 Device SEND REPORT

Content		Block	Value	Description
EIreq	Block size	0	04h	Sum of Control code + EIreq code + Information data
	Control code	1	81h	Fixed value
	EIreq code	2	22h	Fixed value
	Information data	3	xxxxh	Size of write data 0001h to 0800h (Block 4: MSB, Block 3: LSB)
		4		
Data		-	xxh	Report data

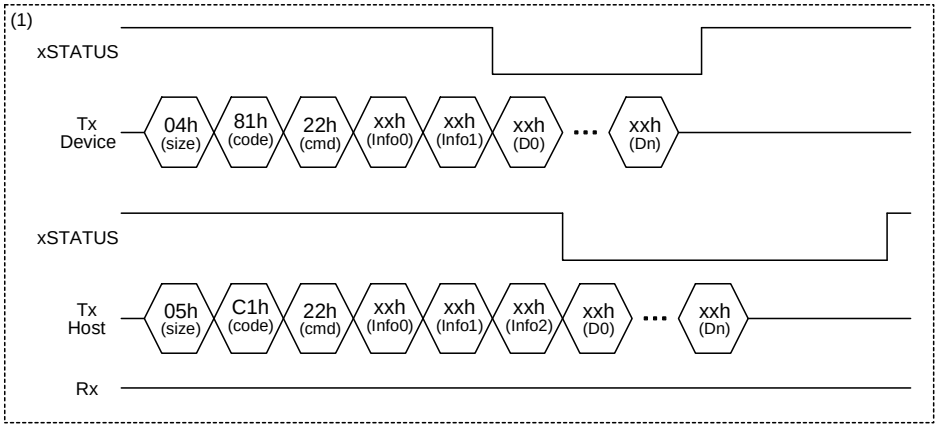
Table 7-23 Host SEND REPORT

Content		Block	Value	Description
EIreq	Block size	0	05h	Sum of Control code + EIreq code + Information data
	Control code	1	C1h	Fixed value
	EIreq code	2	22h	Fixed value
	Information data	3	00h	Fixed value
		4	xxxxh	Size of write data 0001h to 0800h (Block 5: MSB, Block 4: LSB)
		5		
Data		-	xxh	Report data

Achieve access using protocol procedure (1).

Based on the protocol for the Device, when the LSI recognizes a command, it asserts xSTATUS. When the data write from the Main CPU is completed and internal processing ends, the xSTATUS is negated. This status is maintained until the Host generates a request. The transmission timing depends on the request from the Host.

According to the protocol for the Host, when the LSI recognizes a command, it asserts xSTATUS. When the transmission to the Device is completed, the xSTATUS is negated.



7.3.10 23h_RECV REPORT

This EI request reads Report data from the LSI. Table 7-24 shows Device settings. Table 7-25 gives Host settings. The Report data is added after EIhead when transferred from the LSI.

The LSI sets the EIhead “Information data” to the read data size. For information on data size, refer to Section “5.3.6.10 Report ID Registration Information”.

Table 7-24 Device RECV REPORT

Content		Block	Value	Description
EIreq	Block size	0	04h	Sum of Control code + EIreq code + Information data
	Control code	1	81h	Fixed value
	EIreq code	2	23h	Fixed value
	Information data	3	0000h	Fixed value (Block4: MSB, Block3: LSB) * Indicates read data size for EIhead.
		4		
Data		-	xxh	Report data

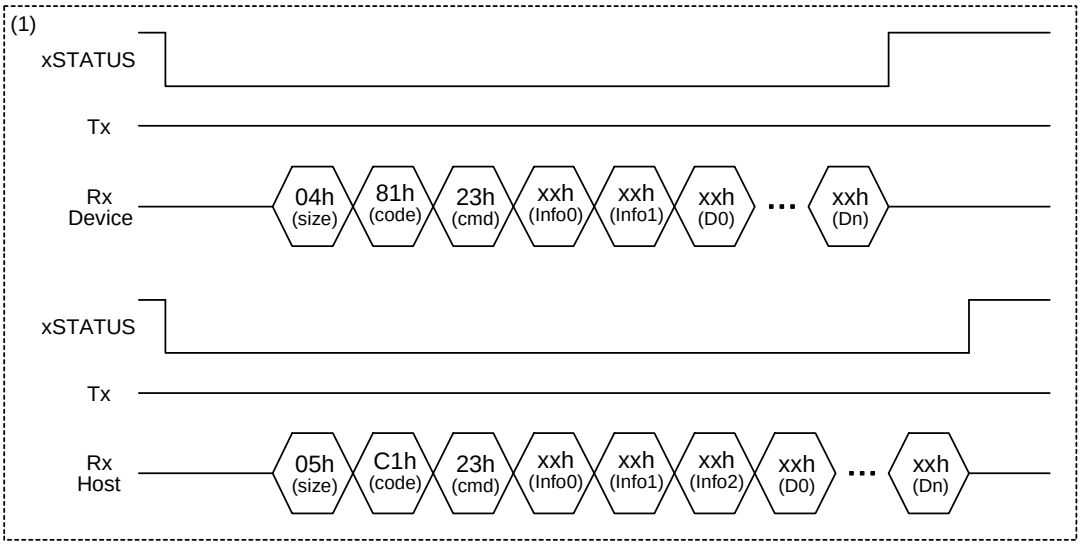
Table 7-25 Host RECV REPORT

Content		Block	Value	Description
EIreq	Block size	0	05h	Sum of Control code + EIreq code + Information data
	Control code	1	C1h	Fixed value
	EIreq code	2	23h	Fixed value
	Information data	3	00h	Fixed value
		4	0000h	Fixed value (Block5: MSB, Block4: LSB) * Indicates read data size for EIhead.
		5		
Data		-	xxh	Report data

The following protocol is valid when the XIRQ_EVENT pin is set to “disable”.

Achieve access using protocol procedure (1).

xSTATUS in (1) is asserted on reception of a Report, after which the data is transferred. Read this data. When the data transfer is completed, the xSTATUS is negated.

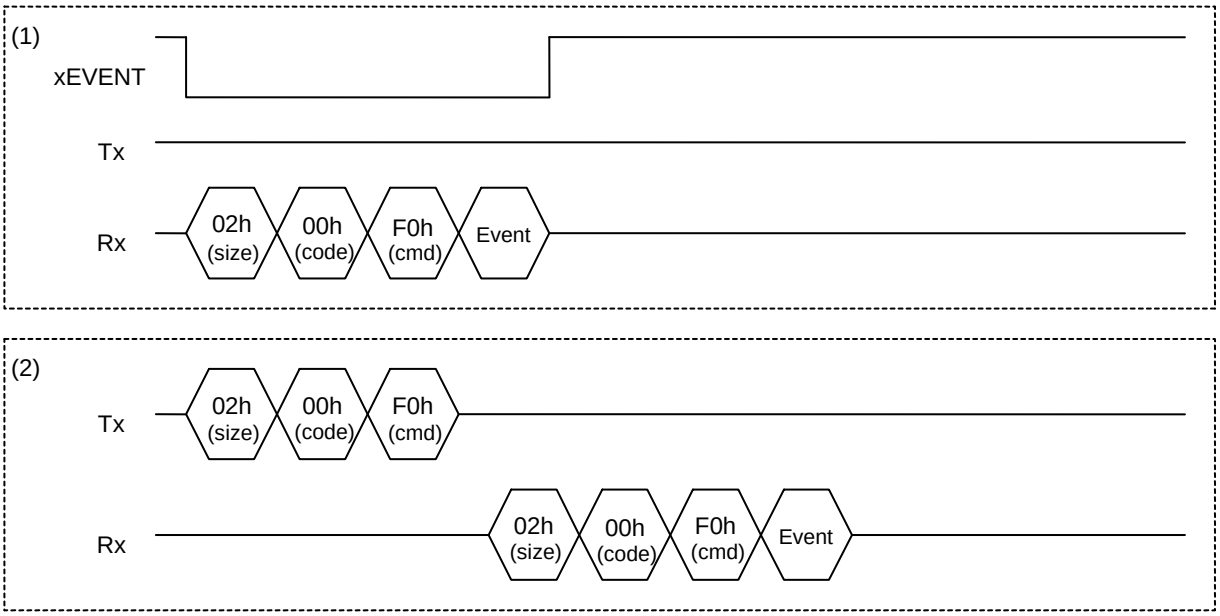


The following protocol is valid when the XIRQ_EVENT pin is set to “enable”.

Achieve access using protocol procedures (1) through (4).

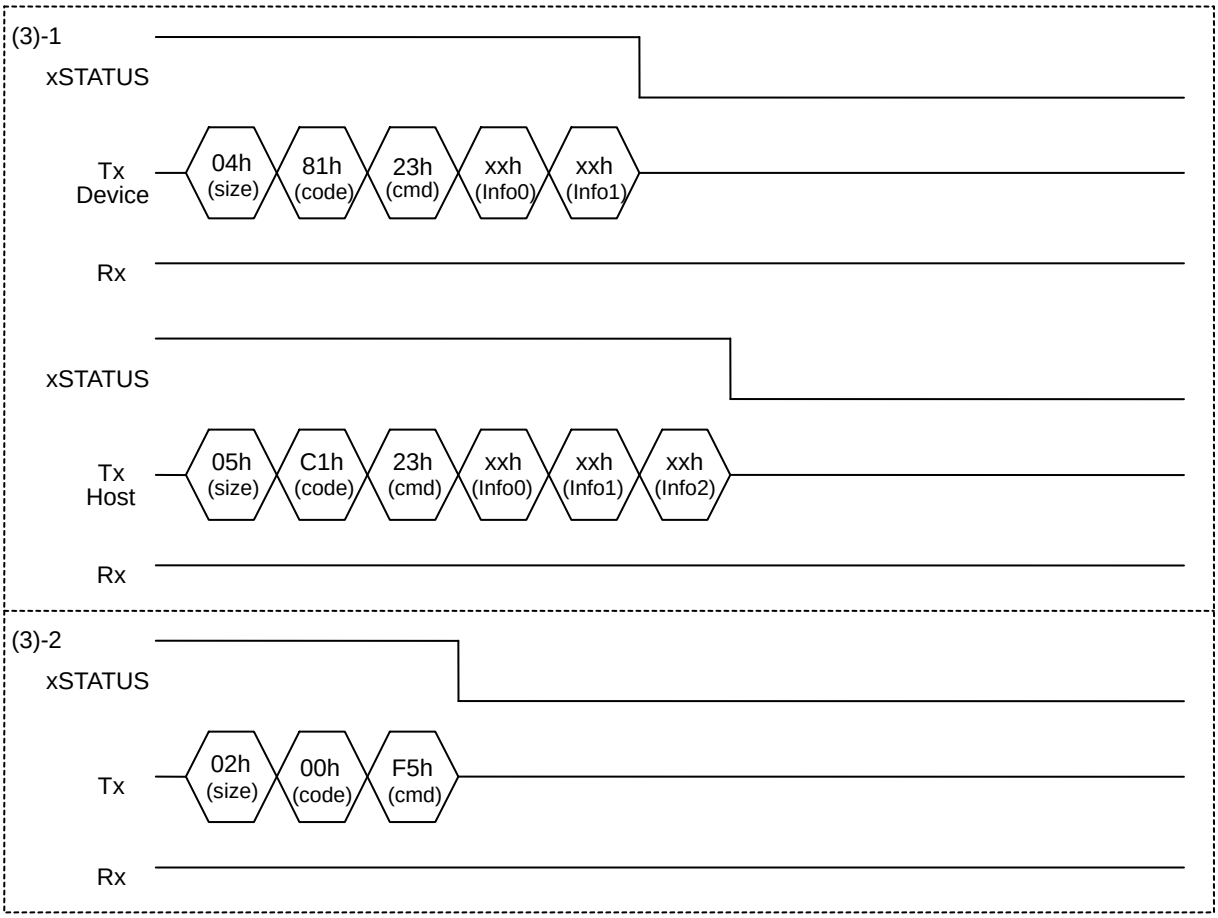
When a Report is received, the event information is passed on to the Main CPU concurrently with xEVENT assert in (1).

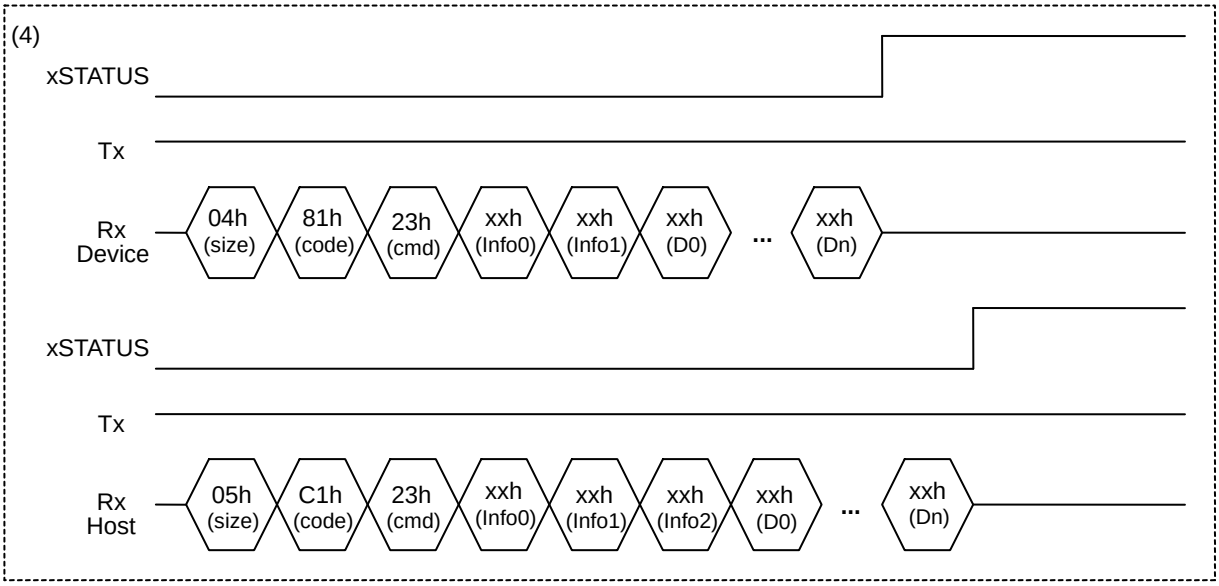
The event information is cleared when the read operation in (2) is executed.



This protocol allows either the (3)-1 or (3)-2 commands to be used. The LSI asserts xSTATUS when it recognizes a command.

When the reception from the USB is completed, data transfer in (4) starts. Read this data. When the data transfer is completed, the xSTATUS is negated.





7.3.11 24h_INITIAL FEATURE REPORT

This EIreq writes the initial values of Feature Report data to the LSI. The Feature Report data default values should be written following the EIreq.

Set “Write data size” to the Feature Report data size. For information on data size, refer to Section “5.3.6.10 Report ID Registration Information”.

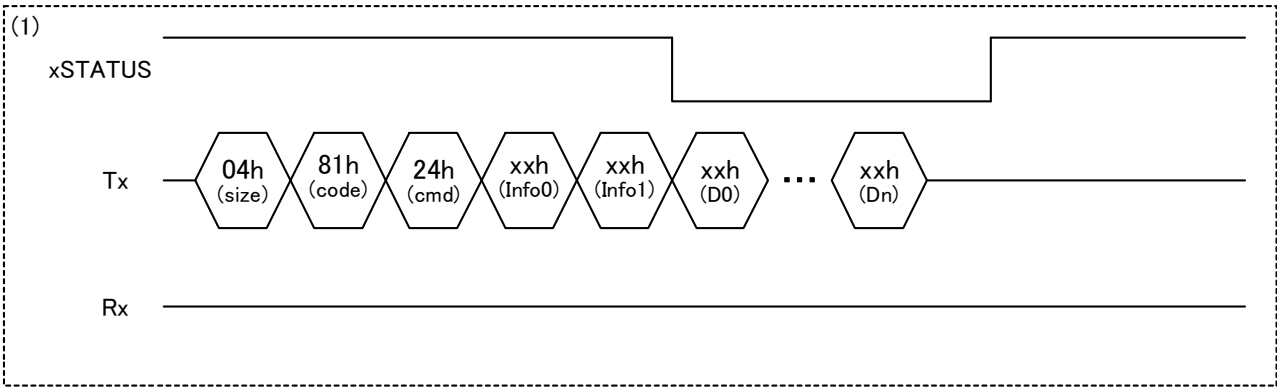
Table 7-26 Device INITIAL FEATURE REPORT

Content		Block	Value	Description
EIreq	Block size	0	04h	Sum of Control code + EIreq code + Information data
	Control code	1	81h	Fixed value
	EIreq code	2	24h	Fixed value
	Information data	3	xxxxh	Size of write data 0001h to 0101h (Block 4: MSB, Block 3: LSB)
		4		
Data		-	xxh	Feature Report data default values

Achieve access using protocol procedure (1).

The LSI asserts xSTATUS when it recognizes a command. When the data write from the Main CPU is completed and internal processing ends, the xSTATUS is negated.

This status is maintained until the Host generates a request. The transmission timing depends on the request from the Host.



7.3.12 25h_GET PROTOCOL MODE

This EIreq reads the protocol mode received from the Host from the LSI. “Data” is added when transferred from the LSI.

Table 7-27 Device GET PROTOCOL MODE

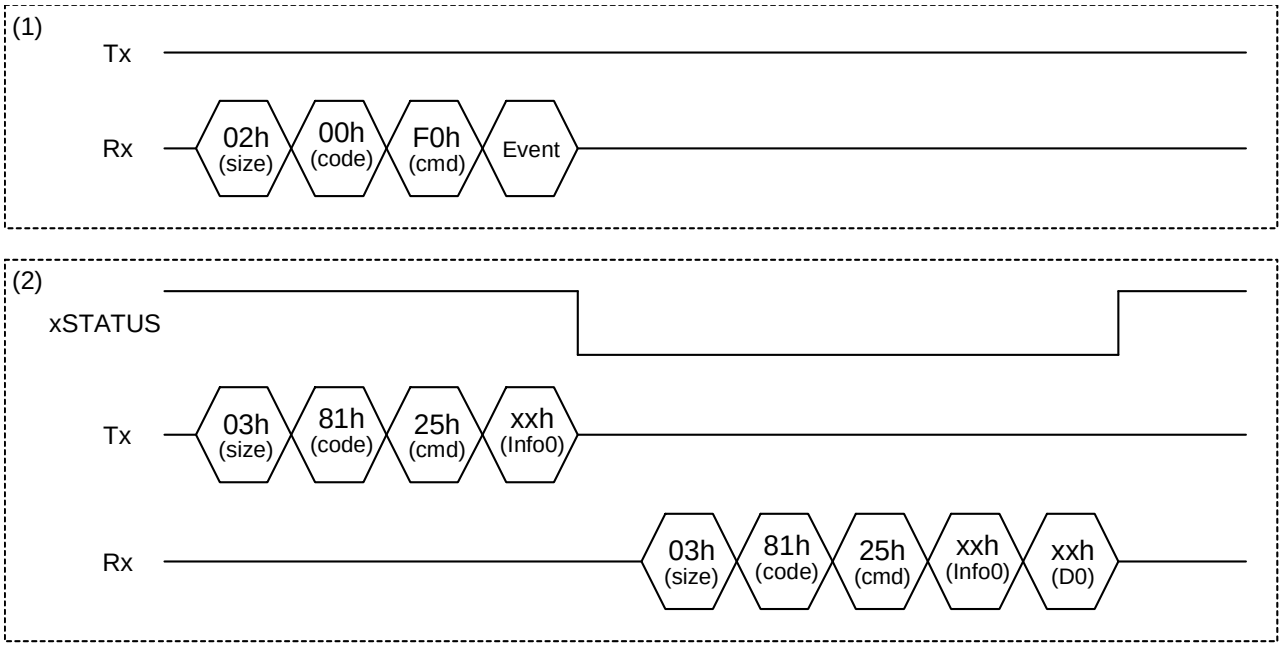
Content		Block	Value	Description
EIreq	Block size	0	03h	Sum of Control code + EIreq code + Information data
	Control code	1	81h	Fixed value
	EIreq code	2	25h	Fixed value
	Information data	3	01h	Size of read data (1 byte fixed)
Data		-	xxh	00h: Boot Protocol 01h: Report Protocol

The following protocol is valid when the XIRQ_EVENT pin is set to “disable”.

Achieve access using protocol procedures (1) and (2).

When Protocol Mode is received from the Host, the event information in (1) is passed on to the Main CPU.

xSTATUS in (2) is asserted and data transferred when the LSI recognizes the command. Read this data.
xSTATUS is negated once data transfer is completed.



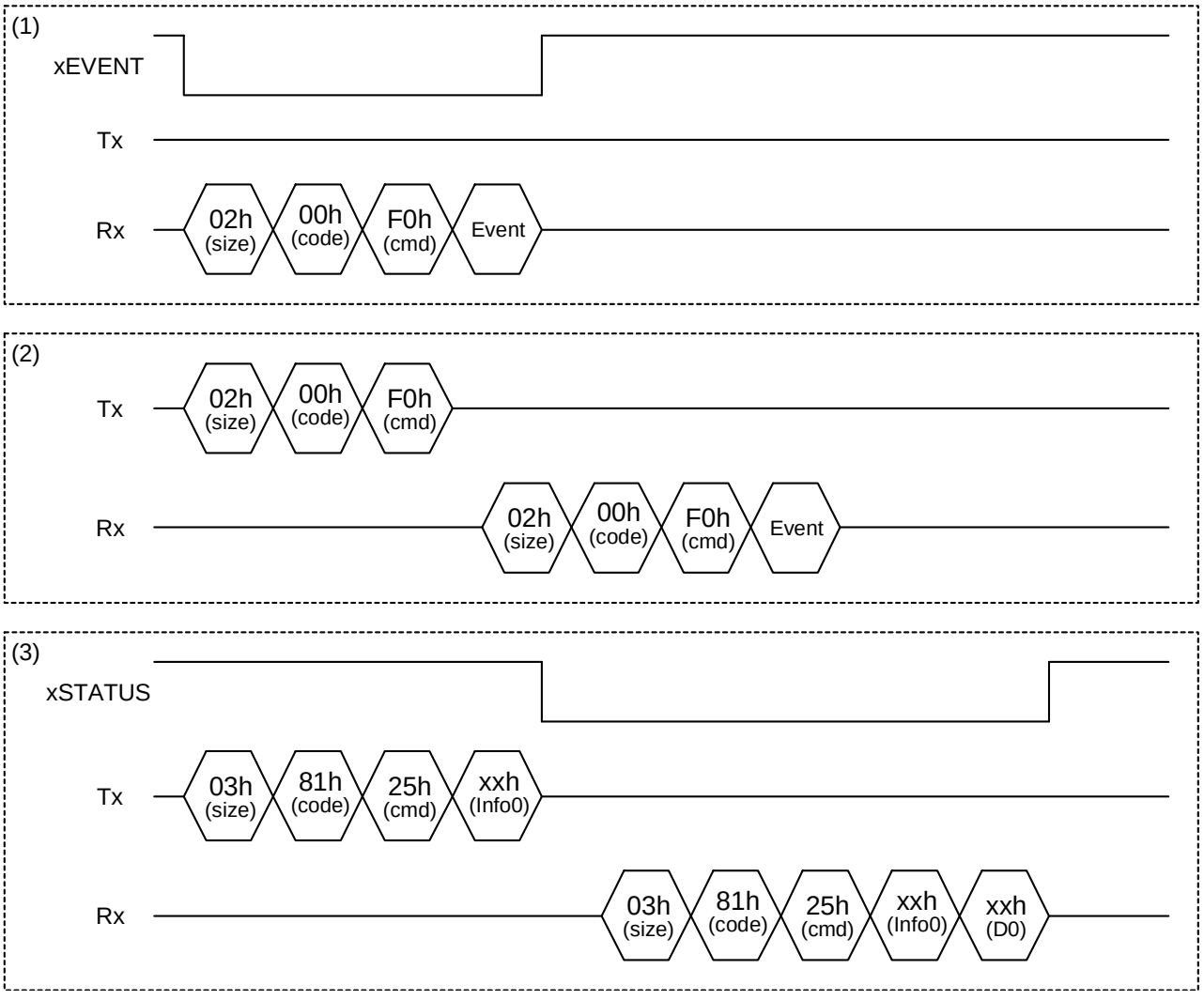
The following protocol is valid when the XIRQ_EVENT pin is set to “enable”.

Achieve access using protocol procedures (1) through (3).

When Protocol Mode is received from the Host, the event information is passed on to the Main CPU concurrently with xEVENT assert in (1).

The event information is cleared when the read operation in (2) is executed.

The LSI asserts xSTATUS in (3) when it recognizes a command and data is transferred. Read this data. When the data transfer is completed, the xSTATUS is negated.



7.3.13 24h_GET DESCRIPTOR

This EIreq reads the descriptor received from the Device from the LSI. “Data” is added during the transfer from the LSI. Note that the Configuration Descriptor includes the Interface Descriptor, HID Descriptor, and Endpoint Descriptor.

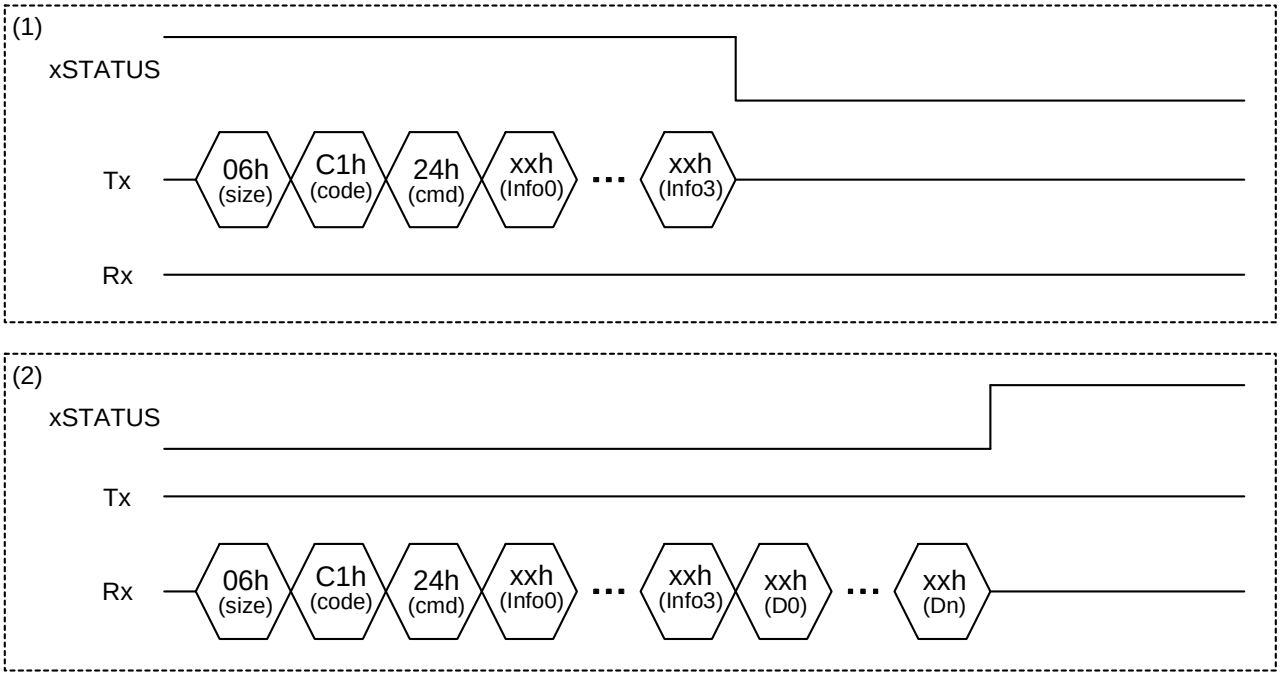
Table 7-28 Host GET DESCRIPTOR

Content		Block	Value	Description
EIreq	Block size	0	06h	Sum of Control code + EIreq code + Information data
	Control code	1	C1h	Fixed value
	EIreq code	2	24h	Fixed value
	Information data	3	xxh	Descriptor type 00h: HID Descriptor 01h: Report Descriptor 02h: String Descriptor 03h: Device Descriptor 04h: Configuration Descriptor 05h to FFh: reserved
		4	xxh	String Index number [00h: Other than String]
		5	xxxxh	Read size [FFFFh: All Descriptor Read setting] 0001h to FFFFh (Block 6: MSB, Block 5: LSB)
		6		
Data		-	xxh	For more information, refer to Section “5.3.6 Device operation setting”.

Achieve access using protocol procedures (1) and (2).

The LSI asserts xSTATUS in (1) when it recognizes a command.

When reception from the Device is completed, data transfer in (2) starts. Read this data. When the data transfer is completed, the xSTATUS is negated.



Appendix-A Descriptor Setting Examples

Descriptor setting examples are listed below for keyboard HID Class Devices.

A.1 Descriptor Header

Table A-1 gives examples of Descriptor Header settings.

Table A-1 Descriptor Header

Details	Size (Byte)	Value	Remarks
wTotalSize	2	00E3h	
wDescriptorInfo	2	0000h	Fixed
WOffsetForDescriptorInfo	2	000Eh	
wClassPeculiarInfo1	2	0301h	Fixed
wOffsetForClassPeculiarInfo1	2	0096h	
wClassPeculiarInfo2	2	0302h	Fixed
wOffsetForClassPeculiarInfo2	2	00D7h	

A.2 Device Descriptor

Table A-2 gives examples of Device Descriptor settings.

Table A-2 Device Descriptor

Details	Size (Byte)	Value	Remarks
bLength	1	12h	Fixed
bDescriptorType	1	01h	Fixed
bcdUSB	2	0200h	Fixed
bDeviceClass	1	00h	Fixed
bDeviceSubClass	1	00h	Fixed
bDeviceProtocol	1	00h	Fixed
bMaxPacketSize0	1	08h	
idVendor	2	04B8h	*
idProduct	2	0314h	
bcdDevice	2	0110h	
iManufacturer	1	01h	
iProduct	1	02h	
iSerialNumber	1	00h	
bNumConfigurations	1	01h	Fixed

* Set the value assigned by USB-IF.

A.3 Configuration Descriptor

Table A-3 gives examples of Configuration Descriptor settings.

Table A-3 Configuration Descriptor

Details	Size (Byte)	Value	Remarks
bLength	1	09h	Fixed
bDescriptorType	1	02h	Fixed
wTotalLength	2	0022h	
bNumInterfaces	1	01h	Fixed
bConfigurationValue	1	01h	Fixed
iConfiguration	1	00h	Fixed
bmAttributes	1	E0h	
bMaxPower	1	31h	

A.4 Interface Descriptor

Table A-4 gives examples of Interface Descriptor settings.

Table A-4 Interface Descriptor

Details	Size (Byte)	Value	Remarks
bLength	1	09h	Fixed
bDescriptorType	1	04h	Fixed
bInterfaceNumber	1	00h	Fixed
bAlternateSetting	1	00h	Fixed
bNumEndpoints	1	01h	
bInterfaceClass	1	03h	Fixed
bInterfaceSubClass	1	01h	
bInterfaceProtocol	1	01h	
iInterface	1	00h	Fixed

A.5 HID Descriptor

Table A-5 gives examples of HID Descriptor settings.

Table A-5 HID Descriptor

Details	Size (Byte)	Value	Remarks
bLength	1	09h	Fixed
bDescriptorType	1	21h	Fixed
bcdHID	2	0111h	Fixed
bCountryCode	1	21h	
bNumDescriptors	1	01h	Fixed
bDescriptorType	1	22h	Fixed
wDescriptorLength	2	0041h	

A.6 Endpoint Descriptor

Table A-6 gives examples of Endpoint Descriptor settings.

Table A-6 Endpoint Descriptor

Details	Size (Byte)	Value	Remarks
bLength	1	07h	Fixed
bDescriptorType	1	05h	Fixed
bEndpointAddress	1	81h	
bmAttributes	1	03h	Fixed
wMaxPacketSize	2	0008h	
bInterval	1	0Ah	

A.7 String Language ID Descriptor

Table A-7 gives examples of String Language ID Descriptor settings.

Table A-7 String Language ID Descriptor

Details	Size (Byte)	Value	Remarks
bLength	1	04h	Fixed
bDescriptorType	1	03h	Fixed
wLangID	2	0409h	

A.8 String Descriptor

Tables A-8 and A-9 give examples of String Descriptor settings

Table A-8 String Descriptor

Details	Size (Byte)	Value	Remarks
bLength	1	24h	
bDescriptorType	1	03h	Fixed
bString［0］	1	53h	
bString［1］	1	00h	
…			
bString［32］	1	2Eh	
bString［33］	1	00h	

Table A-9 String Descriptor

Details	Size (Byte)	Value	Remarks
bLength	1	2Ch	
bDescriptorType	1	03h	Fixed
bString［0］	1	53h	
bString［1］	1	00h	
…			
bString［40］	1	64h	
bString［41］	1	00h	

A.9 Report Descriptor

Table A-10 gives examples of Report Descriptor settings.

Table A-10 Report Descriptor

Item Tag	Size (Byte)	Value	Remarks
USAGE_PAGE	2	05h, 01h	
USAGE	2	09h, 06h	
COLLECTION	2	A1h, 01h	
USAGE	2	05h, 07h	
USAGE_MINIMUM	2	19h, E0h	
USAGE_MAXIMUM	2	29h, E7h	
LOGICAL_MINIMUM	2	15h, 00h	
LOGICAL_MAXIMUM	2	25h, 01h	
REPORT_SIZE	2	75h, 01h	
REPORT_COUNT	2	95h, 08h	
INPUT	2	81h, 02h	
REPORT_COUNT	2	95h, 01h	
REPORT_SIZE	2	75h, 08h	
INPUT	2	81h, 01h	
REPORT_COUNT	2	95h, 03h	
REPORT_SIZE	2	75h, 01h	
USAGE PAGE	2	05h, 08h	
USAGE_MINIMUM	2	19h, 01h	
USAGE_MAXIMUM	2	29h, 03h	
OUTPUT	2	91h, 02h	
REPORT_COUNT	2	95h, 05h	
REPORT_SIZE	2	75h, 01h	
OUTPUT	2	91h, 01h	
REPORT_COUNT	2	95h, 06h	
REPORT_SIZE	2	75h, 08h	
LOGICAL_MINIMUM	2	15h, 00h	
LOGICAL_MAXIMUM	3	26h, FFh, 00h	
USAGE PAGE	2	05h, 07h	
USAGE_MINIMUM	2	19h, 00h	
USAGE_MAXIMUM	3	2A, FFh, 00h	
INPUT	2	81h, 00h	
END_COLLECTION	1	C0h	

A.10 Report ID Registration Information

Table A-11 gives examples of Report ID Registration Information settings.

Table A-11 Report ID Registration Information

Details	Size (Byte)	Value	Remarks
bNumHID_Interfaces	1	01h	Fixed
bInterfaceNo	1	00h	Fixed
bNumInitReports	1	02h	Number of Reports
bReserve	1	00h	Fixed
bReportType	1	01h	INPUT
bReportID	1	00h	
wReportLen	2	0008h	8 bytes
bReportType	1	02h	OUTPUT
bReportID	1	00h	
wReportLen	2	0001h	1 byte

Appendix-B UNICODE

Table B-1 shows part of the details for “Basic Latin” within the internationally standardized UNICODE text code system. Refer to the relevant standards for the latest information.

Table B-1 UNICODE text strings

UNICODE	String
0030h	0
0031h	1
0032h	2
0033h	3
0034h	4
0035h	5
0036h	6
0037h	7
0038h	8
0039h	9

UNICODE	String
0041h	A
0042h	B
0043h	C
0044h	D
0045h	E
0046h	F
0047h	G
0048h	H
0049h	I
004Ah	J
004Bh	K
004Ch	L
004Dh	M
004Eh	N
004Fh	O
0050h	P
0051h	Q
0052h	R
0053h	S
0054h	T
0055h	U
0056h	V
0057h	W
0058h	X
0059h	Y
005Ah	Z

UNICODE	String
0061h	a
0062h	b
0063h	c
0064h	d
0065h	e
0066h	f
0067h	g
0068h	h
0069h	i
006Ah	j
006Bh	k
006Ch	l
006Dh	m
006Eh	n
006Fh	o
0070h	p
0071h	q
0072h	r
0073h	s
0074h	t
0075h	u
0076h	v
0077h	w
0078h	x
0079h	y
007Ah	z

Appendix-C Country Code

Table C-1 lists HID Descriptor country codes for reference. Refer to the USB standard for the latest information.

Table C-1 Country Code

Code	Country	Code	Country	Code	Country
00h	Not Supported	10h	Korean	20h	UK
01h	Arabic	11h	Latin American	21h	US
02h	Belgian	12h	Netherlands/Dutch	22h	Yugoslavia
03h	Canadian-Bilingual	13h	Norwegian	23h	Turkish-F
04h	Canadian-French	14h	Persian (Farsi)	24h-FFh	reserved
05h	Czech Republic	15h	Poland		
06h	Danish	16h	Portuguese		
07h	Finnish	17h	Russia		
08h	French	18h	Slovakia		
09h	German	19h	Spanish		
0Ah	Greek	1Ah	Swedish		
0Bh	Hebrew	1Bh	Swiss/French		
0Ch	Hungary	1Ch	Swiss/German		
0Dh	International (ISO)	1Dh	Switzerland		
0Eh	Italian	1Eh	Taiwan		
0Fh	Japan (Katakana)	1Fh	Turkish-Q		

Appendix-D Report Descriptor Setting Examples

The following sections describe various examples of Report Descriptor settings for a mouse, keyboard, and vendor-defined HID Class. To ensure flexibility in settings even within the USB standard, precise specifications are not provided for some Report Descriptor settings. Refer to the examples for more information.

D.1 Mouse Setting Examples

Table D-1 lists examples of mouse Report Descriptor settings. The “Group” column refers to the group setting for each Item Tag.

Table D-1 Mouse Report Descriptor

Item Tag	Size (Byte)	Value	Description	Group
USAGE_PAGE	2	05h, 01h	Generic Desktop Controls	A
USAGE	2	09h, 02h	Mouse	
COLLECTION	2	A1h, 01h	Application	B
USAGE	2	09h, 01h	Pointer	C
COLLECTION	2	A1h, 00h	Physical	D
REPORT_COUNT	2	95h, 03h	Report count = 3	E
REPORT_SIZE	2	75h, 01h	Report size = 1 bit	
USAGE_PAGE	2	05h, 09h	Button	
USAGE_MINIMUM	2	19h, 01h	Usage minimum value = 1 (Button 1)	
USAGE_MAXIMUM	2	29h, 03h	Usage maximum value = 3 (Button 3)	
LOGICAL_MINIMUM	2	15h, 00h	Report logical minimum value = 0	
LOGICAL_MAXIMUM	2	25h, 01h	Report logical maximum value = 1	
INPUT	2	81h, 02h	Input Report (Data, Variable, Absolute)	
REPORT_COUNT	2	95h, 01h	Report count = 1	F
REPORT_SIZE	2	75h, 05h	Report size = 5 bits	
INPUT	2	81h, 01h	Input Report (Constant) 5-bit padding	
REPORT_SIZE	2	75h, 08h	Report size = 8 bits	G
REPORT_COUNT	2	95h, 03h	Report count = 3 *1	
USAGE_PAGE	2	05h, 01h	Generic Desktop Controls	
USAGE	2	09h, 30h	X direction	
USAGE	2	09h, 31h	Y direction	
USAGE	2	09h, 38h	Wheel *2	
LOGICAL_MINIMUM	2	15h, 81h	Report logical minimum value = -127	
LOGICAL_MAXIMUM	2	25h, 7Fh	Report logical maximum value = 127	
INPUT	2	81h, 06h	Input Report (Data, Variable, Relative)	
END_COLLECTION	1	C0h	COLLECTION (Physical) end	D
END_COLLECTION	1	C0h	COLLECTION (Application) end	B

Make the following changes when using just Boot Protocol.
*1 “95h”, “02h”
*2 Delete item

D.2 Keyboard Setting Examples

Table D-2 lists examples of keyboard Report Descriptor settings. The “Group” column refers to the group setting for each Item Tag.

Table D-2 Keyboard Report Descriptor

Item Tag	Size (Byte)	Value	Description	Group
USAGE_PAGE	2	05h, 01h	Generic Desktop Controls	A
USAGE	2	09h, 06h	Keyboard	
COLLECTION	2	A1h, 01h	Application	B
REPORT_SIZE	2	75h, 01h	Report size = 1 bit	C
REPORT_COUNT	2	95h, 08h	Report count = 8	
USAGE_PAGE	2	05h, 07h	Key Codes	
USAGE_MINIMUM	2	19h, E0h	Usage minimum value = 224 (Left Control key)	
USAGE_MAXIMUM	2	29h, E7h	Usage maximum value = 231 (Right GUI)	
LOGICAL_MINIMUM	2	15h, 00h	Report logical minimum value = 0	
LOGICAL_MAXIMUM	2	25h, 01h	Report logical maximum value = 1	
INPUT	2	81h, 02h	Input Report (Data, Variable, Absolute)	
REPORT_COUNT	2	95h, 01h	Report count = 1	D
REPORT_SIZE	2	75h, 08h	Report size = 8 bits	
INPUT	2	81h, 01h	Input Report (Constant) fixed value	
REPORT_COUNT	2	95h, 05h	Report count = 5	E
REPORT_SIZE	2	75h, 01h	Report size = 1 bit	
USAGE_PAGE	2	05h, 08h	LEDs	
USAGE_MINIMUM	2	19h, 01h	Usage minimum value = 1 (Num Lock)	
USAGE_MAXIMUM	2	29h, 05h	Usage maximum value = 5 (Kana)	
OUTPUT	2	91h, 02h	Output Report (Data, Variable, Absolute)	
REPORT_COUNT	2	95h, 01h	Report count = 1	F
REPORT_SIZE	2	75h, 03h	Report size = 3 bits	
OUTPUT	2	91h, 01h	Output Report (Constant) 3-bit padding	
REPORT_COUNT	2	95h, 06h	Report count = 6	G
REPORT_SIZE	2	75h, 08h	Report size = 8 bits	
LOGICAL_MINIMUM	2	19h, 00h	Report logical minimum value = 0	
LOGICAL_MAXIMUM	3	26h, FFh, 00h	Report logical maximum value = 255	
USAGE_PAGE	2	05h, 07h	key codes	
USAGE_MINIMUM	2	19h, 00h	Usage minimum value = 0	
USAGE_MAXIMUM	3	2Ah, FFh, 00h	Usage maximum value = 255	
INPUT	2	81h, 00h	Input Report (Data, Array)	
END_COLLECTION	1	C0h	COLLECTION (Application) end	B

D.3 Vendor-defined Setting Examples

Table D-3 lists examples of vendor-defined Report Descriptor settings. The “Group” column refers to the group setting for each Item Tag.

Table D-3 Vendor Defined Report Descriptor

Item Tag	Size (Byte)	Value	Description	Group
USAGE_PAGE	3	06h, 01h, FFh	Vendor Defined Page	A
USAGE	2	09h, 01h	Vendor Defined Usage	
COLLECTION	2	A1h, 01h	Application	B
REPORT_ID	2	85h, 01h	Report ID = 1	C
REPORT_SIZE	2	75h, 08h	Report size = 8 bits	
REPORT_COUNT	2	95h, 08h	Report count = 8	
LOGICAL_MINIMUM	2	15h, 00h	Report logical minimum value = 0	
LOGICAL_MAXIMUM	3	26h, FFh, 00h	Report logical maximum value = 255	
USAGE	2	09h, 02h	Vendor Defined Usage=2	
OUTPUT	2	91h, 02h	Output Report (Data, Variable, Absolute)	
USAGE	2	09h, 03h	Vendor Defined Usage=3	D
REPORT_COUNT	2	95h, 08h	Report count = 8	
INPUT	2	81h, 02h	Input Report (Data, Variable, Absolute)	
END_COLLECTION	1	C0h	COLLECTION (Application) end	B
USAGE_PAGE	3	06h, 02h, FFh	Vendor Defined Page	E
USAGE	2	09h, 01h	Vendor Defined Usage	
COLLECTION	2	A1h, 01h	Application	F
REPORT_ID	2	85h, 02h	Report ID = 2	G
REPORT_SIZE	2	75h, 08h	Report size = 8 bits	
REPORT_COUNT	3	96h, 00h, 01h	Report count = 256	
LOGICAL_MINIMUM	2	15h, 00h	Report logical minimum value = 0	
LOGICAL_MAXIMUM	3	26h, FFh, 00h	Report logical maximum value = 255	
USAGE	2	09h, 02h	Vendor Defined Usage=2	
INPUT	3	82h, 02h, 01h	Input Report (Data, Variable, Absolute, Buf)	
USAGE	2	09h, 03h	Vendor Defined Usage=3	H
OUTPUT	3	92h, 02h, 01h	Output Report (Data, Variable, Absolute, Buf)	
END_COLLECTION	1	C0h	COLLECTION (Application) end	F

D.4 Report Descriptor Notes

This section provides additional information on the Report Descriptor. For specifics, refer to the USB standard.

D.4.1 Report

(1) Reports may exist as INPUT, OUTPUT, or FEATURE Reports.

(2) Report IDs are defined only when used.

(3) Report size is specified by REPORT_COUNT and REPORT_SIZE.
REPORT_SIZE is set in bits.

(4) COLLECTION and END COLLECTION group data.

(5) The Report size is calculated for each Report within the range specified by COLLECTION.
Using Report ID allows different Report sizes to be defined even for the same type.

(6) Details included before INPUT (OUTPUT, FEATURE) are inherited.
Certain details such as REPORT_COUNT or REPORT_SIZE may be omitted.

D.4.2 Item Tag

(1) Report Descriptor sets the data as required after the Item Tag in Section “5.3.6.9 Report Descriptor”.

(2) The Item Tag types are listed in Table D-4. (This is not a comprehensive list.)

Table D-4 Item Tags

Main Item Tag	Global Item Tag	Local Item Tag
INPUT	USAGE PAGE	USAGE
OUTPUT	LOGICAL MINIMUM	USAGE MINIMUM
FEATURE	LOGICAL MAXIMUM	USAGE MAXIMUM
COLLECTION	REPORT COUNT	STRING INDEX
END COLLECTION	REPORT SIZE	STRING MINIMUM
−	REPORT ID	STRING MAXIMUM
−	UNIT	DELIMITER
−	UNIT EXPONENT	−

Appendix-E Report ID Registration Information Setting Examples

Setting examples are given below for Report ID registration information for mouse, keyboard, and vendor definitions with a USB Host. Report ID registration information is generated by obtaining the Report Descriptor from the Device connected.

E.1 Mouse Setting Examples

Table E-1 lists Report ID registration information setting examples for a mouse. The settings given in Table E-1 are examples created from the Report Descriptor in Appendix-D.1, “Mouse Setting Examples”.

Table E-1 Mouse Report ID Registration Information

Details	Size (Byte)	Value	Remarks
bNumHID_Interfaces	1	01h	Fixed value
bInterfaceNo	1	00h	Fixed value
bNumInitReports	1	01h	Number of Reports
bReserve	1	00h	Fixed value
bReportType	1	01h	INPUT
bReportID	1	00h	
wReportLen	2	0004h	4 bytes

E.2 Keyboard Setting Examples

Table E-2 lists Report ID registration information setting examples for a keyboard. The settings given in Table E-2 are examples created from the Report Descriptor in Appendix-D.2 “Keyboard Setting Examples”.

Table E-2 Keyboard Report ID Registration Information

Details	Size (Byte)	Value	Remarks
bNumHID_Interfaces	1	01h	Fixed value
bInterfaceNo	1	00h	Fixed value
bNumInitReports	1	02h	Number of Reports
bReserve	1	00h	Fixed value
bReportType	1	01h	INPUT
bReportID	1	00h	
wReportLen	2	0008h	8 bytes
bReportType	1	02h	OUTPUT
bReportID	1	00h	
wReportLen	2	0001h	1 byte

E.3 Vendor-defined Setting Examples

Table E-3 lists Report ID registration information setting examples for vendor definitions. The settings given in Table E-3 are examples created from the Report Descriptor in Appendix-D.3, “Vendor-defined Setting Examples”.

Table E-3 Vendor Defined Report ID Registration Information

Details	Size (Byte)	Value	Remarks
bNumHID_Interfaces	1	01h	Fixed value
bInterfaceNo	1	00h	Fixed value
bNumInitReports	1	04h	Number of Reports
bReserve	1	00h	Fixed value
bReportType	1	02h	OUTPUT
bReportID	1	01h	
wReportLen	2	0009h	9 bytes
bReportType	1	01h	INPUT
bReportID	1	01h	
wReportLen	2	0009h	9 bytes
bReportType	1	01h	INPUT
bReportID	1	02h	
wReportLen	2	0101h	257 bytes
bReportType	1	02h	OUTPUT
bReportID	1	02h	
wReportLen	2	0101h	257 bytes

Appendix-F Other Configuration Examples

The LSI can be connected to the Main CPU in various configurations, depending on the control method used. Select the optimum configuration to suit the particular system used.

F.1 Minimum Connection Configuration Example

The configuration example shown in Fig. F-1 may be used when the XIRQ_EVENT pin is set to “disable” and power management is disabled by “SLEEP” (01h). The SIO_READY, XIRQ_STATUS, XIRQ_EVENT, and WAKEUP pins do not have to be connected.

Note that an alternative method when not connecting SIO_READY pin is to wait for a specific time.

Refer to Appendix-F.2, “Alternative Method When SIO_READY Pin Is Left Open”.

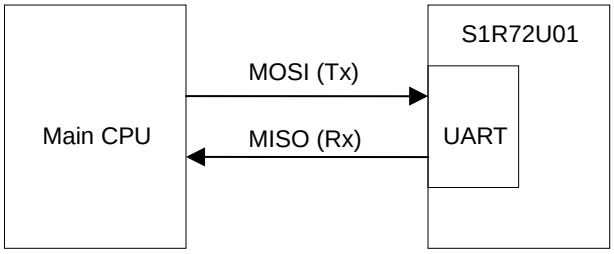


Fig. F-1 Minimum connection configuration example

The configuration example shown in Fig. F-2 may be used when power management is enabled by “SLEEP” (01h) or when the XIRQ_EVENT pin is set to “enable”.

The WAKEUP pin can be controlled from devices other than the Main CPU.

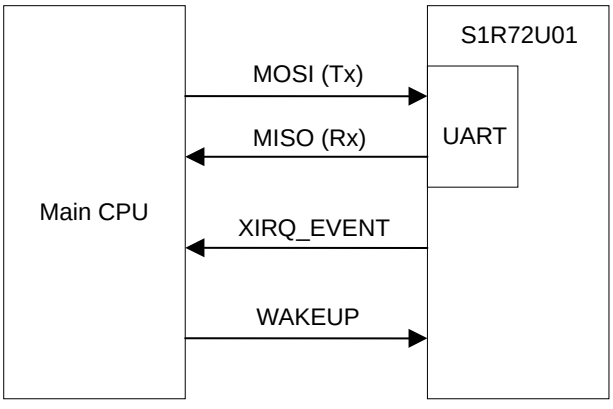


Fig. F-2 Power management configuration example

F.2 Alternative Method When SIO_READY Pin Is Left Open

This section describes the alternative method when the SIO_READY pin is left open (not connected).

F.2.1 Initialization

The confirmation processes for the SIO_READY pin described in "5.1.3 UART Initialization flow" can be performed by an alternative method of waiting for a specific time. These are the sections in Fig. F-3 enclosed in bold boxes.

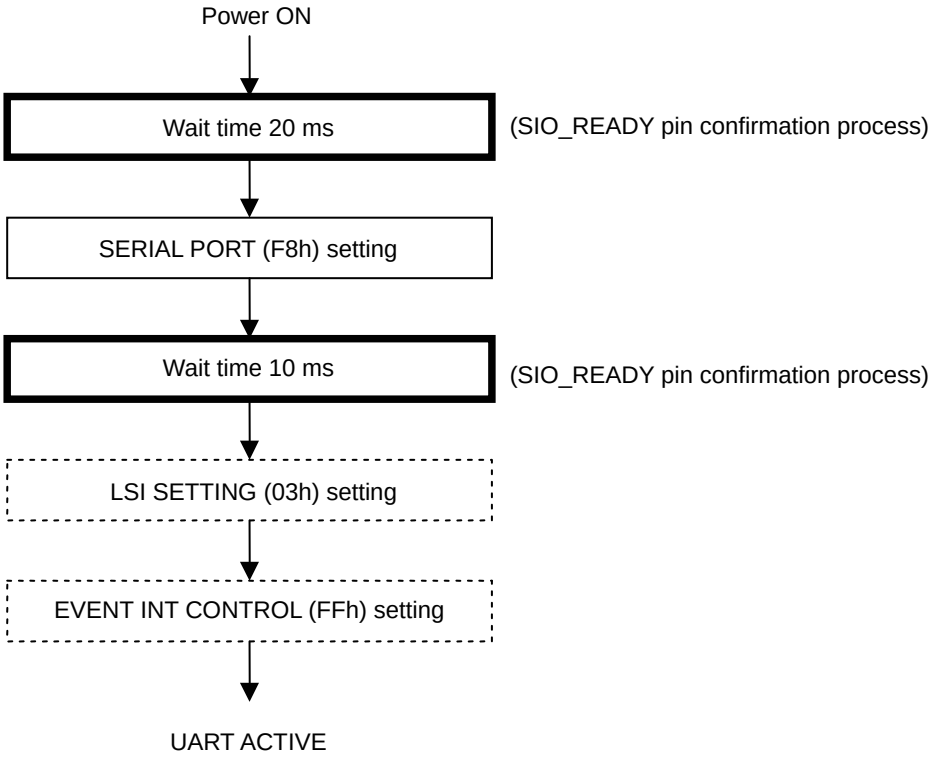


Fig. F-3 Alternative initialization flow

F.2.2 Power management processing

The confirmation process for Low to High for the SIO_READY pin can be performed by an alternative method of waiting for a specific time in power management.

Table F-1 Alternative power management processing

USB	Item covered	Wait time
Host	5.2.9.1 Control when Device is not connected [Step 2]	20 ms
	5.2.9.3 SLEEP control when Device is connected [Step 4]	150 ms
	5.2.9.5 SLEEP (Remote Wakeup) control when Device is connected [Step 6]	150 ms
Device	5.3.9.1 Control when Host is not connected [Step 2]	15 ms
	5.3.9.3 SLEEP control when connected to Host [Step 6]	15 ms
	5.3.9.5 SLEEP (Remote Wakeup) control when connected to Host [Step 4]	15 ms

Appendix-G Initialization Flow

This describes the initialization flow from the point at which the power supply is switched on until the HID Class is ready to operate.

G.1 HID Class Host Flow

The LSI can be initialized using the “5.1.3 UART initialization flow” and “5.2.3 Host initialization flow”. If a Device is subsequently connected, it can be used as an HID Class Host by the procedural flow specified in “5.2.6 Device connection”.

Refer to the corresponding section for detailed information on the corresponding procedural flow.

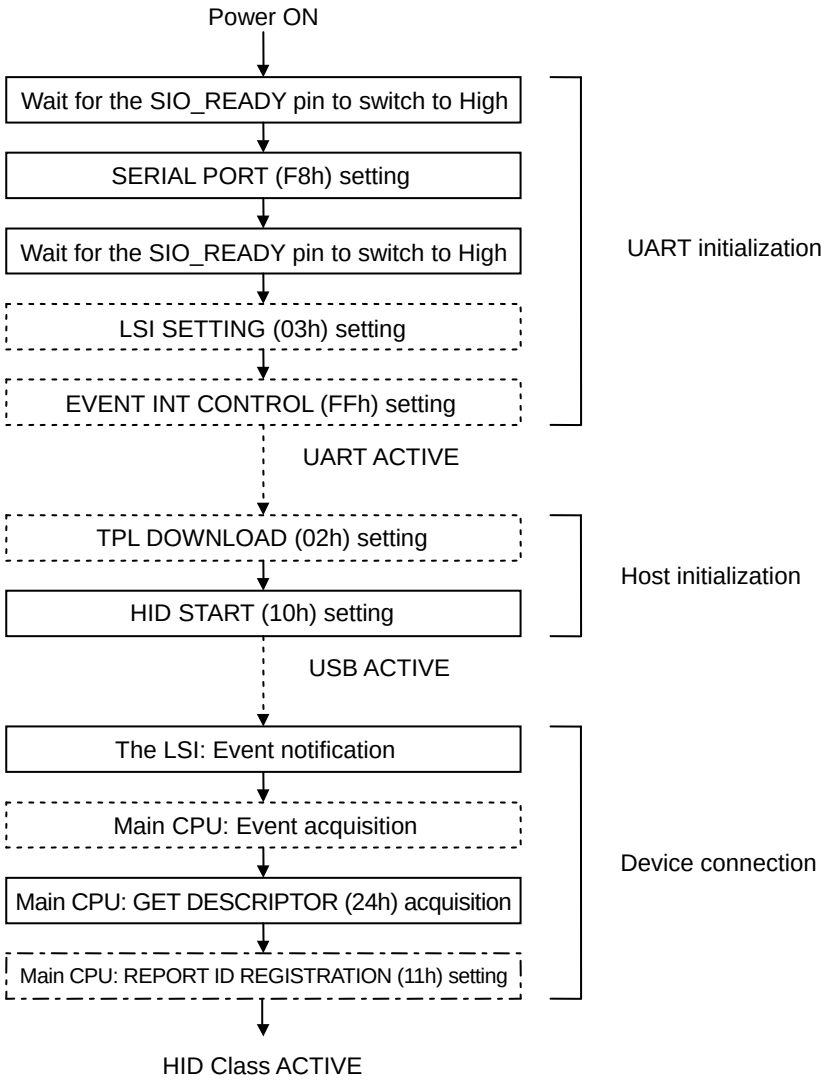


Fig. G-1 HID Class Host flow

G.2 HID Class Device Flow

The LSI can be initialized using the “5.1.3 UART initialization flow” and “5.3.3 Device initialization flow”. If a Host is subsequently connected, it can be operated as an HID Class Device by the procedural flow specified in “5.3.7 Host connection”.

Refer to the corresponding section for detailed information on the corresponding procedural flow.

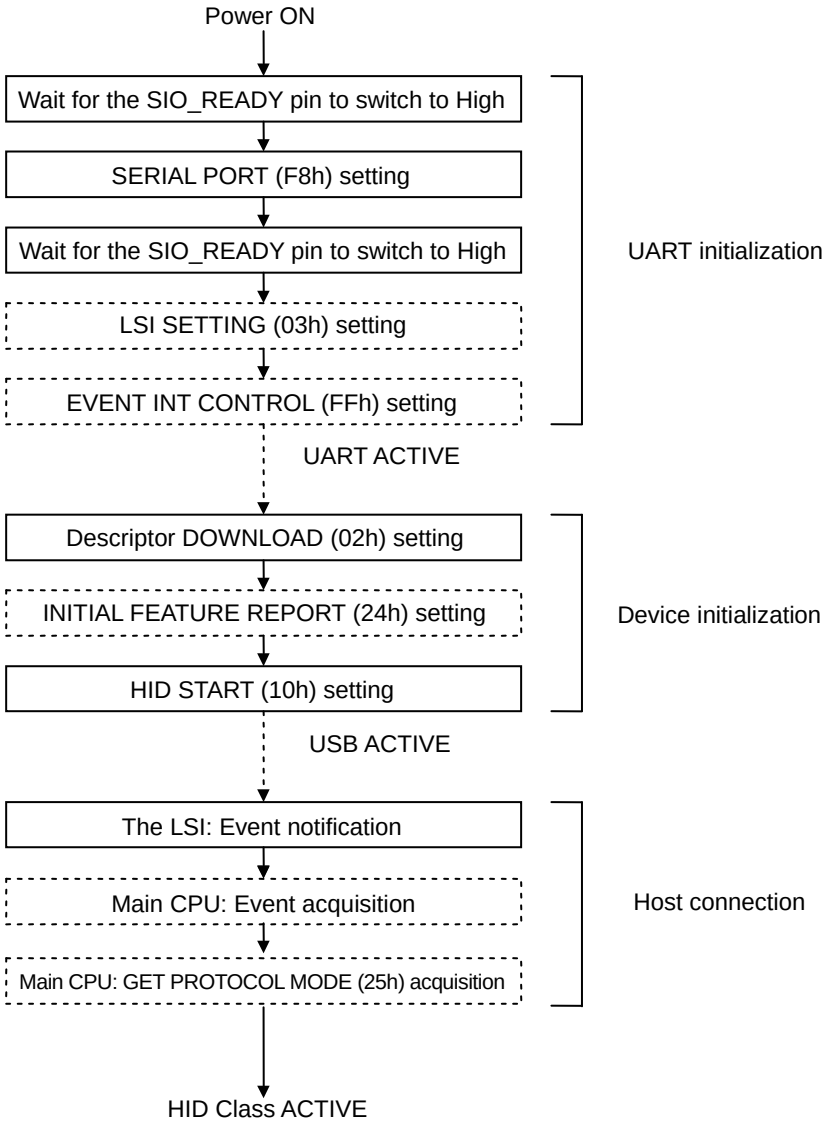


Fig. G-2 HID Class Device flow

Revision History

Attachment-1

Rev. No.	Date	Page	Category	Revision details (including previous details) and reason
Rev.1.00	03/05/2009	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: 411544402
First Issue March 2009 in JAPAN Ⓓ