

# LPC55S2x/LPC552x

Errata sheet LPC55S2x/LPC552x

Rev. 2.4 — June 16, 2023

Errata sheet

## Document information

Info	Content
<b>Keywords</b>	LPC55S28JBD100, LPC55S26JBD100, LPC55S28JEV98 LPC55S26JEV98, LPC55S28JBD64, LPC55S26JBD64, LPC55S28JBD100, LPC55S26JBD100, LPC55S28JEV98, LPC55S26JEV98, LPC55S28JBD64, LPC55S26JBD64
<b>Abstract</b>	LPC55S2x/LPC552x errata



## Revision history

Rev	Date	Description
2.4	20230616	<ul style="list-style-type: none"> <li>Added <a href="#">Section 3.14 “PLL.1: PLL LOCK bit is not reliable”</a></li> </ul>
2.3	20230118	<ul style="list-style-type: none"> <li>Updated <a href="#">Table 3 “Errata notes”</a>.</li> <li>Added <a href="#">Section 5.3 “ROM.5: ROM get-property is not functional”</a></li> <li>In <a href="#">Section 5.2 “ROM.4: Stack pointer for application code cannot be located in the SRAM area when shared with PowerQuad RAM and USB RAM”</a> Updated Title from “ROM.1: Stack pointer for application code cannot be located in the SRAM area when shared with PowerQuad RAM and USB RAM” to “ROM.4: Stack pointer for application code cannot be located in the SRAM area when shared with PowerQuad RAM and USB RAM”.</li> </ul>
2.2	20220620	<p>Added <a href="#">Section 3.13 “ROM.3: ENTER_ISP_MODE mailbox command does not work”</a></p> <p>Added <a href="#">Section 3.12 “ROM.2: ROM API can’t be used correctly to update and read monotonic counter in CFPA.VENDOR_USAGE word”</a>.</p>
2.1	20220228	Updated <a href="#">Section 5.1 “ISP.1: ROM runBootloader API to enter ISP mode does not work”</a> and short description of ISP.1 in <a href="#">Table 3 “Errata notes”</a> .
2.0	20211110	Added <a href="#">Section 5.2 “ROM.4: Stack pointer for application code cannot be located in the SRAM area when shared with PowerQuad RAM and USB RAM”</a> in <a href="#">Section 5 “Errata notes detail”</a> .
1.9	20210810	Added VBAT_DCDC.1: <a href="#">Section 3.11 “VBAT_DCDC.1: The minimum rise time of the power supply must be 2.6 ms or slower for Tamb = -40 C, and 0.5 ms or slower for Tamb = 0 C to +105 C”</a> .
1.8	20210423	<p>Added USB.5 errata, <a href="#">Section 3.8 “USB.5: In USB high-speed device mode, when device isochronous IN endpoint sends a packet of MaxPacketSize of 1024 bytes in response to IN token from host, the isochronous IN endpoint interrupt is not set and the endpoint command/status list entry for the isochronous IN endpoint is not updated”</a>.</p> <p>Added USB.6 errata, <a href="#">Section 3.9 “USB.6: In USB high-speed host mode, only one transaction per micro-frame is allowed for isochronous IN endpoints”</a>.</p> <p>Added DM-AP.1 errata, <a href="#">Section 3.10 “DM-AP.1: I2C ISP via debug mailbox is not functional”</a>.</p>
1.7	20210225	Added USB.4 errata, <a href="#">Section 3.7 “USB.4: In USB high-speed device mode, device writes extra byte(s) to the buffer”</a> .
1.6	20201214	Includes <a href="#">Section 3.5 “USB.3: For the USB high-speed device controller, the detection handshaking fails when certain full-speed hubs are connected”</a> .
1.5	20200826	Adds <a href="#">Section 3.6 “WAKEIO.1: Wake-up I/O register reports incorrect wake-up source”</a> .
1.4	20200826	Adds <a href="#">Section 5.1 “ISP.1: ROM runBootloader API to enter ISP mode does not work”</a> .
1.3	20191204	Updated ROM.1 workaround.

## Contact information

For more information, please visit: <http://www.nxp.com>

For sales office addresses, please send an email to: [salesaddresses@nxp.com](mailto:salesaddresses@nxp.com)

**Revision history** ...continued

Rev	Date	Description
1.2	20191021	Enhances product identification and adds USB.1 and USB.2 errata.
1.1	20190923	Describes ROM failure to enter ISP mode when an image is corrupted with flash pages in an erased or unprogrammed state.
1.0	20190719	Initial version.

## 1. Product identification

The LPC55S2x/LPC552x VFBGA98 package has the following top-side marking:

- First line: LPC55S2x/LPC552x
- Second line: JEV98
- Third line: xxxxxxxx
- Fourth line: zzzyywwxR
  - yyww: Date code with yy = year and ww = week.
  - xR: Device revision 1B

The LPC55S2x/LPC552x HLQFP100 package has the following top-side marking:

- First line: LPC55S2x/LPC552x
- Second line: xxxxxxxx
- Third line: zzzyywwxR
  - yyww: Date code with yy = year and ww = week.
  - xR: Device revision 1B

The LPC55S2x/LPC552x HTQFP64 package has the following top-side marking:

- First line: LPC55S2x/LPC552x
- Second line: JBD64
- Third line: xxxx
- Fourth line: xxxx
- Fifth line: zzzyywwxR
  - yyww: Date code with yy = year and ww = week.
  - xR: Device revision 1B

## 2. Errata overview

**Table 1. Functional problems table**

Functional problems	Short description	Revision identifier	Detailed description
ADC.1	Async interrupts with resume not supported.	1B	<a href="#">Section 3.1</a>
ROM.1	ROM fails to enter ISP mode when image is corrupted with flash pages in an erased or unprogrammed state.	1B	<a href="#">Section 3.2</a>
USB.1	USB HS host fails when connecting to an LS device (mouse).	1B	<a href="#">Section 3.3</a>
USB.2	Automatic USB rate adjustment not functional when using multiple hubs.	1B	<a href="#">Section 3.4</a>
USB.3	For the USB high-speed device controller, the detection handshaking fails when certain full-speed hubs are connected.	1B	<a href="#">Section 3.5</a>
WAKEIO.1	Wake-up I/O cause register identifies the wake-up I/O (WAKEUP pins) source from deep power-down mode.	1B	<a href="#">Section 3.6</a>

**Table 1. Functional problems table ...continued**

Functional problems	Short description	Revision identifier	Detailed description
USB.4	In USB high-speed device mode, device writes extra byte(s) to the buffer if the NBytes is not multiple of 8 for OUT transfer.	1B	<a href="#">Section 3.7</a>
USB.5	In USB high-speed device mode, when device isochronous IN endpoint sends a packet of MaxPacketSize of 1024 bytes in response to IN token from host, the isochronous IN endpoint interrupt is not set and the endpoint command/status list entry for the isochronous IN endpoint is not updated.	1B	<a href="#">Section 3.8</a>
USB.6	In USB high-speed host mode, only one transaction per micro-frame is allowed for isochronous IN endpoints.	1B	<a href="#">Section 3.9</a>
DM-AP.1	I2C ISP via debug mailbox is not functional.	1B	<a href="#">Section 3.10</a>
VBAT_DCDC.1	The minimum rise time of the power supply must be 2.6 ms or slower for Tamb = -40 C, and 0.5 ms or slower for Tamb = 0 C to +105 C.	1B	<a href="#">Section 3.11</a>
ROM.2	ROM API can't be used correctly to update and read monotonic counter in CFP.VENDOR_USAGE word.	0A, 1B	<a href="#">Section 3.12</a>
ROM.3	ENTER_ISP_MODE mailbox command does not work.	0A, 1B	<a href="#">Section 3.13</a>
PLL.1	PLL LOCK bit is not reliable	0A, 1B	<a href="#">Section 3.14</a>

**Table 2. AC/DC deviations table**

AC/DC deviations	Short description	Product version(s)	Detailed description
n/a	n/a	n/a	n/a

**Table 3. Errata notes**

Errata notes	Short description	Revision identifier	Detailed description
ISP.1	Devices with date code 2101 (yyww) onward, contain ROM Bootloader version T1.1.5 or higher with runBootloader API to enter into ISP mode.	1B	<a href="#">Section 5.1 "ISP.1: ROM runBootloader API to enter ISP mode does not work"</a> .
ROM.4	Stack pointer for application code cannot be located in the SRAM area when shared with PowerQuad RAM and USB RAM.	1B	<a href="#">Section 5.2 "ROM.4: Stack pointer for application code cannot be located in the SRAM area when shared with PowerQuad RAM and USB RAM"</a> .
ROM.5	Devices with date code 2222 (yyww) and onwards contain ROM Target version T1.1.0 (or higher). On these devices, the get-property command is not functional and as a result, bootloader properties like target version cannot be read.	1B	<a href="#">Section 5.3 "ROM.5: ROM get-property is not functional"</a> .

## 3. Functional problems detail

### 3.1 ADC.1: Async interrupts with resume not supported

#### Introduction

The ADC controller is available on all LPC55S2x/LPC552x devices. Trigger detect with up to 16 trigger sources is supported with priority level configuration. A software or hardware trigger option is provided for each.

#### Problem

The following problems are all related to the restart after interrupt feature:

- Low priority trigger executes twice when resumed.
- Trigger can't restart when it is configured to do so.
- Incorrect trigger resumed after exception.

#### Work-around

There is no work-around.

The async interrupts with resume is not supported on device revisions 0A and 1B.

### 3.2 ROM.1: ROM fails to enter ISP mode when image is corrupted with flash pages in an erased or unprogrammed state

#### Introduction

On the LPC55S2x/LPC552x, if the image is corrupted with flash pages in an erased or unprogrammed state, the ROM may fail to automatically enter ISP mode.

#### Problem

When secure boot is enabled in CMPA, and the flash memory contains an erased or unprogrammed memory page inside the memory region specified by the image size field in the image header, the device does not automatically enter into ISP mode using the fallback mechanism, as in the case of a failed boot for an invalid image. This problem occurs when the application image is only partially written or erased but a valid image header is still present in memory.

#### Work-around

Perform a mass-erase to remove the incomplete and corrupted image using one of the following methods:

- Execute the erase command using Debug Mailbox. The device will enter directly into ISP mode after exiting the mailbox.
- Enter into ISP mode using the Debug Mailbox command and use the flash-erase command.
- Reset the device and enter into ISP mode using the ISP pin. Use the flash-erase command to erase the corrupted (incomplete) image.

### 3.3 USB.1: HS host fails when connecting with the LS device (mouse)

#### Introduction

The USB1 high-speed controller is available on select LPC55S2x/LPC552x devices and provides a plug-and-play connection of peripheral devices to a host with three different data speeds:

- high-speed with a data rate of 480Mbps.
- full-speed with a data rate of 12 Mbps.
- low-speed with a data rate of 1.5 Mbps.

Many portable devices can benefit from the ability to communicate with each other over the USB interface without intervention of a host PC.

#### Problem

USB HS host fails when connecting with an LS device (mouse).

#### Work-around

To support Full-Speed and Low-Speed applications, it is recommended to use the USB0 Full-Speed port and the USB1 High-speed port for Device or Host. In addition, should an application require support of Low-Speed USB devices with a USB High-Speed Host, this can be accomplished by inserting a USB Hub between the USB1 High-speed port and external USB devices.

### 3.4 USB.2: Automatic USB rate adjustment is not functional when using multiple hubs

#### Introduction:

Full-speed and low-speed signaling uses bit stuffing throughout the packet without exception. If the receiver sees seven consecutive ones anywhere in the packet, then a bit stuffing error has occurred, and the packet should be ignored.

The time interval just before an End of Packet (EOP) is a special case. The last data bit before the EOP can become stretched by hub switching skews. This is known as dribble and can lead to a situation where dribble introduces a sixth bit that does not require a bit stuff. Therefore, the receiver must accept a packet where there are up to six full bit times at the port with no transitions prior to the EOP.

#### Problem:

The LPC55S2x/LPC552x devices use the start of an EOP for frequency measurements. This is not functional when going through multiple hubs that introduce a dribble bit because of hub switching skews. For this reason, the start of the EOP cannot be used for frequency measurements for automatic USB rate adjustment (by setting USBCLKADJ in the FRO192M\_CTRL register). The problem does not occur when a single hub is used.

#### Work-around:

Use the FRO calibration library provided in technical note TN00063. This library allows the application to have a crystal-less USB device operation in full-speed mode.

### 3.5 USB.3: For the USB high-speed device controller, the detection handshaking fails when certain full-speed hubs are connected

#### Introduction

See the USB2.0 specification for details regarding the USB High-speed Detection Handshake protocol.

#### Problem

As a high-speed device, when certain full-speed hubs are connected, the USB device does not detect the HOST KJ sequence correctly and, as a result, does not recognize the speed of the connected host. In this case, the USB device can act erratically due to the wrong speed detection.

#### Work-around

There are two workarounds:

1. The software work-around below can be implemented in `usb_dev_hid_mouse` where API is called `"USB_DeviceHsPhyChirpIssueWorkaround()"`. In event handler in `USB_DeviceCallback()`,
  - On `"kUSB_DeviceEventBusReset"` event, `USB_DeviceHsPhyChirpIssueWorkaround()` should be called to identify the speed of the host connected to. If full-speed host is connected or `"isConnectedToFshostFlag"` is set, `FORCE_FS` (bit 21) of `DEVCMSTAT` register should be set to force the device operating in full-speed mode.
  - On `"kUSB_DeviceEventDetach"` event, `FORCE_FS` (bit 21) of `DEVCMSTAT` register should be cleared.
2. The software work-around below is available in tech note (TN00071) In event handler in `USB_DeviceCallback()`,
  - On `"kUSB_DeviceEventAttach"` event, set `PHY_RX` register trip-level voltage to the highest. `USBPHY->RX &= ~(USBPHY_RX_ENVADJ_MASK); USBPHY->RX |= 2;`
  - On `"kUSB_DeviceEventBusReset"` event, check the `DEVCMSTAT[SPEED]` to determine the connected bus speed. (`SPEED` are bits 22 and 23). If `DEVCMSTAT[SPEED]=FS`, `FORCE_FS` (bit 21) of `DEVCMSTAT` should be set to force the device operating in full-speed mode.
  - On `"kUSB_DeviceEventGetDeviceDescriptor"` event, or first `SETUP` packet has arrived, Set the `USBPHY_RX[ENVADJ]` field back to default 0. Otherwise, `USBPHY_RX[ENVADJ]` field will remain as 2 unless a disconnect event occurs.
  - On `"kUSB_DeviceEventDetach"` event, Clear `FORCE_FS` (bit 21) of `DEVCMSTAT` register to zero. Reset `USBPHY_RX[ENVADJ]` field back to default 0.

### 3.6 WAKEIO.1: Wake-up I/O register reports incorrect wake-up source

#### Introduction

On the LPC55S2x/LPC552x, a wake-up I/O cause register is available to identify the wake-up I/O (WAKEUP pins) source from deep power-down mode.



### Problem

Before entering DEEP-POWER-DOWN, the following configuration is set for the wake-up sources:

- Wake-up I/O 0 as falling edge.
- Wake up I/O 1 as rising edge.
- Wake up I/O 2 is disabled.
- Wake up I/O 3 as falling edge.

1st case: The first wake up event is a rising edge on wake up I/O 1. PMC->WAKEIOCAUSE will indicate Wake-up I/O 0 trigger (wrong), Wake up I/O 1 trigger (correct) and Wake up I/O 3 trigger (wrong) as wake up I/O cause.

2nd case: The first wake up event is a falling edge on wake up I/O 3. PMC->WAKEIOCAUSE will indicate Wake-up I/O 0 trigger (wrong) and Wake up I/O 3 trigger (correct) as wake up I/O cause.

3rd case: The first wake up event is an RTC. PMC->WAKEIOCAUSE will indicate Wake-up I/O 0 trigger (wrong) and Wake up I/O 3 trigger (wrong) as wake up I/O cause.

### Work-around

None. This issue does not occur when the wake I/O is not configured as falling edge.

## 3.7 USB.4: In USB high-speed device mode, device writes extra byte(s) to the buffer

### Introduction

The LPC55S2x/LPC552x device family include a USB high-speed interface (USB1) that can operate in device mode at high-speed. The NBytes value represents the number of bytes that can be received in the buffer.

### Problem

The LPC55S2x/LPC552x USB device controller writes extra bytes to the receive data buffer if the size of the transfer is not a multiple of 8 bytes since the USB device controller always writes 8 bytes. For example, if the transfer length is 1 bytes, 7 extra bytes will be written to the receive data buffer. If the transfer length is 7 bytes, 1 extra bytes will be written to the receive data buffer.

### Work-around

Reserve an additional, intermediary buffer along with the buffer used by the application for USB data. After the USB data transfer into the intermediary buffer has been completed, use memcpy to move the data from the intermediary buffer into the application buffer, skipping the extraneous extra byte. This software work-around is implemented on the SDK software platform.

### 3.8 USB.5: In USB high-speed device mode, when device isochronous IN endpoint sends a packet of MaxPacketSize of 1024 bytes in response to IN token from host, the isochronous IN endpoint interrupt is not set and the endpoint command/status list entry for the isochronous IN endpoint is not updated

#### Introduction

The LPC55S2x/LPC552x device family include a USB high-speed interface (USB1) that can operate in device mode at high-speed. The isochronous IN endpoint supports a MaxPacketSize of 1024 bytes.

#### Problem

When device isochronous IN endpoint sends a packet of MaxPacketSize of 1024 bytes in response to IN token from host, the isochronous IN endpoint interrupt is not set and the endpoint command/status list entry for the isochronous IN endpoint is not updated.

#### Work-around

Restrict the isochronous IN endpoint MaxPacketSize to 1023 bytes in device descriptor.

### 3.9 USB.6: In USB high-speed host mode, only one transaction per micro-frame is allowed for isochronous IN endpoints

#### Introduction

The LPC55S2x/LPC552x device family include a USB high-speed interface which can operate in host mode. Up to three high-speed transactions are allowed in a single micro-frame to support high-bandwidth endpoints. This mode is enabled by setting the Mult (Multiple) field in the Proprietary Transfer Descriptor (PTD) and is used to indicate to the host controller the number of transactions that should be executed per micro-frame. The allowed bit settings are:

00b Reserved. A zero in this field yields undefined results.

01b One transaction to be issued for this endpoint per micro-frame.

10b Two transactions to be issued for this endpoint per micro-frame.

11b Three transactions to be issued for this endpoint per micro-frame.

#### Problem

For High-bandwidth mode, using multiple packets (MULT = 10b or 11b) in a frame causes unreliable operation. Only one transaction (MULT = 01b) can be issued per micro-frame.

#### Work-around

There is no software workaround. Only one transaction can be issued per micro-frame.

### 3.10 DM-AP.1: I2C ISP via debug mailbox is not functional

#### Introduction

The LPC55S2x/LPC552x supports I2C ISP mode via debug mailbox. By default, ISP

mode entry is determined by the state of the ISP boot selection pins at reset time. Usually this functionality is disabled through PFR configuration prior to field deployment or when ISP boot selection pins are used for some other board function. In this situation, the DM\_AP command “Enter ISP mode” can be used to enter one of the ISP modes (UART, SPI, I2C, USB-HID). Parameter value of 0x2 can be used to enter I2C ISP mode.

#### Problem

I2C ISP via debug mailbox cannot be entered and is not functional.

#### Work-around

There is no software workaround.

### 3.11 VBAT\_DCDC.1: The minimum rise time of the power supply must be 2.6 ms or slower for Tamb = -40 C, and 0.5 ms or slower for Tamb = 0 C to +105 C

#### Introduction

The datasheet specifies no power-up requirements for the power supply on the VBAT\_DCDC pin.

#### Problem

The device might not always start-up if the minimum rise time of the power supply ramp is 2.6 ms or faster for Tamb = -40 C, and 0.5 ms or faster for Tamb = 0 C to +105 C.

#### Work-around

None.

### 3.12 ROM.2: ROM API can't be used correctly to update and read monotonic counter in CFWA.VENDOR\_USAGE word

#### Introduction

Customer Field Programmable Area (CFPA) of Protected Flash Region (PFR) contains VENDOR\_USAGE word. The lower 16-bits of the VENDOR\_USAGE word implement a monotonic counter which should contain current value or higher value when new version of CFWA page is written. Upper 16-bits of the VENDOR\_USAGE word should contain inverse value of aforesaid monotonic counter.

#### Problem

In the ROM, 16-bit monotonic counter is implemented by upper 16-bits of the VENDOR\_USAGE word while lower 16-bits contain inverse value of monotonic counter i.e Monotonic Counter and its inverse value are swapped erroneously in the ROM. Due this error, ROM APIs do not access VENDOR\_USAGE monotonic counter correctly.

#### Work-around

User should increment and store Monotonic Counter value in upper 16-bits of VENDOR\_USAGE word while inverse value of the monotonic counter should be stored in the lower 16-bits of the VENDOR\_USAGE word.

### 3.13 ROM.3: ENTER\_ISP\_MODE mailbox command does not work

#### Introduction

The Debug Mailbox is used to communicate with code executing in the ROM by sending mailbox commands. ENTER\_ISP\_MODE mailbox command is used to put device in the ISP mode.

#### Problem

ENTER\_ISP\_MODE mailbox command does not work in the devices with boot ROM version T1.1.3 or lower.

#### Work-around

There is no work around. This issue is fixed in the ROM Bootloader versions T1.1.4 or higher, i.e. date code 1935 (yyww) onward.

### 3.14 PLL.1: PLL LOCK bit is not reliable

#### Introduction

On the LPC55S2x/LPC552x devices, PLLxSTAT register of PLLs contains a LOCK detector status bit (bit 0 of PLLxSTAT register).

When the LOCK detector status bit is set to 1, the PLL is considered to be locked and stable.

The PLL LOCK signal is specified to work for Fref range from 100 kHz to 20 MHz. When the Fref is below 100 kHz or above 20 MHz, software should use a 6 ms time interval to insure the PLL will be stable.

#### Problem

On the LPC55S2x/LPC552x, the PLL status LOCK bit is not always reliable in the ranges specified and as a result, the PLL doesn't initialize correctly.

#### Work-around

For  $F_{ref} \geq 20$  MHz:

Software must wait at least  $(500\mu s + 400/F_{ref})$  ( $F_{ref}$  in Hz result in s) to ensure the PLL is stable.

For  $F_{ref} < 20$  MHz:

- If the PLL lock detector status bit is 1 before the wait time duration  $((500\mu s + 400/F_{ref}))$  is completed, the PLL is stable.
- If the PLL lock detector status bit is 0 but the wait time duration  $((500\mu s + 400/F_{ref}))$  is completed, the PLL is stable.

Software workaround is implemented in SDK 2.14 clock driver version 2.3.7.

**Remark:** This errata does not apply for spread spectrum mode.

## 4. AC/DC deviations detail

---

No known errata.

## 5. Errata notes detail

---

### 5.1 ISP.1: ROM runBootloader API to enter ISP mode does not work

#### Introduction

Devices with date code 2101 (yyww) onward, contain ROM Bootloader version T1.1.5 or higher with runBootloader API to enter into ISP mode.

#### Problem

RunBootloader API does not work as described in the User Manual.

#### Work-around

User should use below code in their application to invoke auto ISP mode instead of runBootloader API.

```
PMC->AOREG1 |= (0X0A <<16);  
NVIC_SystemReset();
```

### 5.2 ROM.4: Stack pointer for application code cannot be located in the SRAM area when shared with PowerQuad RAM and USB RAM

#### Introduction

Stack pointer for application code cannot be located in the SRAM area when shared with PowerQuad RAM and USB RAM.

#### Problem

ROM fails to check image if image stack address is  $\geq 0x20040000$  and will enter ISP mode.

#### Work-around

Workaround is to modify the Startup.s and assign the stack pointer to an address  $< 0x20040000$ . SDK examples have been modified to handle this scenario.

### 5.3 ROM.5: ROM get-property is not functional

#### Introduction

ROM provides get-property command feature to provide bootloader properties.

#### Problem

Devices with date code 2222 (yyww) and onwards contain ROM Target version T1.1.0 (or higher). On these devices, the get-property command is not functional and as a result, bootloader properties like target version cannot be read. On devices with date code before 2222 (yyww), get-property command is functional.

## Work-around

None.

## 6. Contents

<b>1</b>	<b>Product identification</b>	<b>4</b>		
<b>2</b>	<b>Errata overview</b>	<b>4</b>		
<b>3</b>	<b>Functional problems detail</b>	<b>6</b>		
3.1	ADC.1: Async interrupts with resume not supported	6		
	Introduction	6		
	Problem	6		
	Work-around	6		
3.2	ROM.1: ROM fails to enter ISP mode when image is corrupted with flash pages in an erased or unprogrammed state	6		
	Introduction	6		
	Problem	6		
	Work-around	6		
3.3	USB.1: HS host fails when connecting with the LS device (mouse)	7		
	Introduction	7		
	Problem	7		
	Work-around	7		
3.4	USB.2: Automatic USB rate adjustment is not functional when using multiple hubs	7		
	Introduction	7		
	Problem	7		
	Work-around	7		
3.5	USB.3: For the USB high-speed device controller, the detection handshaking fails when certain full-speed hubs are connected	8		
	Introduction	8		
	Problem	8		
	Work-around	8		
3.6	WAKEIO.1: Wake-up I/O register reports incorrect wake-up source	8		
	Introduction	8		
	Problem	9		
	Work-around	9		
3.7	USB.4: In USB high-speed device mode, device writes extra byte(s) to the buffer	9		
	Introduction	9		
	Problem	9		
	Work-around	9		
3.8	USB.5: In USB high-speed device mode, when device isochronous IN endpoint sends a packet of MaxPacketSize of 1024 bytes in response to IN token from host, the isochronous IN endpoint interrupt is not set and the endpoint command/status list entry for the isochronous IN endpoint is not updated	10		
	Introduction	10		
	Problem	10		
	Work-around	10		
3.9	USB.6: In USB high-speed host mode, only one transaction per micro-frame is allowed for isochronous IN endpoints	10		
	Introduction	10		
	Problem	10		
	Work-around	10		
3.10	DM-AP.1: I2C ISP via debug mailbox is not functional	10		
	Introduction	10		
	Problem	11		
	Work-around	11		
3.11	VBAT_DCDC.1: The minimum rise time of the power supply must be 2.6 ms or slower for Tamb = -40 C, and 0.5 ms or slower for Tamb = 0 C to +105 C	11		
	Introduction	11		
	Problem	11		
	Work-around	11		
3.12	ROM.2: ROM API can't be used correctly to update and read monotonic counter in CFFPA.VENDOR_USAGE word	11		
	Introduction	11		
	Problem	11		
	Work-around	11		
3.13	ROM.3: ENTER_ISP_MODE mailbox command does not work	12		
	Introduction	12		
	Problem	12		
	Work-around	12		
3.14	PLL.1: PLL LOCK bit is not reliable	12		
	Introduction	12		
	Problem	12		
	Work-around	12		
<b>4</b>	<b>AC/DC deviations detail</b>	<b>13</b>		
<b>5</b>	<b>Errata notes detail</b>	<b>13</b>		
5.1	ISP.1: ROM runBootloader API to enter ISP mode does not work	13		
	Introduction	13		
	Problem	13		
	Work-around	13		
5.2	ROM.4: Stack pointer for application code cannot be located in the SRAM area when shared with PowerQuad RAM and USB RAM	13		
	Introduction	13		
	Problem	13		
	Work-around	13		

[continued >>](#)

5.3 ROM.5: ROM get-property is not functional . . . 13  
 Introduction . . . . . 13  
 Problem . . . . . 13  
 Work-around . . . . . 14

**6 Contents . . . . . 15**

---

Please be aware that important notices concerning this document and the product(s) described herein, have been included in section 'Legal information'.

---



## How To Reach Us

### Home Page:

[nxp.com](http://nxp.com)

### Web Support:

[nxp.com/support](http://nxp.com/support)

**Limited warranty and liability** — Information in this document is provided solely to enable system and software implementers to use NXP products. There are no express or implied copyright licenses granted hereunder to design or fabricate any integrated circuits based on the information in this document. NXP reserves the right to make changes without further notice to any products herein.

NXP makes no warranty, representation, or guarantee regarding the suitability of its products for any particular purpose, nor does NXP assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. "Typical" parameters that may be provided in NXP data sheets and/or specifications can and do vary in different applications, and actual performance may vary over time. All operating parameters, including "typicals," must be validated for each customer application by customer's technical experts. NXP does not convey any license under its patent rights nor the rights of others. NXP sells products pursuant to standard terms and conditions of sale, which can be found at the following address: [nxp.com/SalesTermsandConditions](http://nxp.com/SalesTermsandConditions).

**Right to make changes** - NXP Semiconductors reserves the right to make changes to information published in this document, including without limitation specifications and product descriptions, at any time and without notice. This document supersedes and replaces all information supplied prior to the publication hereof.

**Security** — Customer understands that all NXP products may be subject to unidentified or documented vulnerabilities. Customer is responsible for the design and operation of its applications and products throughout their lifecycles to reduce the effect of these vulnerabilities on customer's applications and products. Customer's responsibility also extends to other open and/or proprietary technologies supported by NXP products for use in customer's applications. NXP accepts no liability for any vulnerability. Customer should regularly check security updates from NXP and follow up appropriately. Customer shall select products with security features that best meet rules, regulations, and standards of the intended application and make the ultimate design decisions regarding its products and is solely responsible for compliance with all legal, regulatory, and security related requirements concerning its products, regardless of any information or support that may be provided by NXP. NXP has a Product Security Incident Response Team (PSIRT) (reachable at [PSIRT@nxp.com](mailto:PSIRT@nxp.com)) that manages the investigation, reporting, and solution release to security vulnerabilities of NXP products.

NXP, the NXP logo, NXP SECURE CONNECTIONS FOR A SMARTER WORLD, are trademarks of NXP B.V. All other product or service names are the property of their respective owners. AMBA, Arm, Arm7, Arm7TDMI, Arm9, Arm11, Artisan, big.LITTLE, Cordio, CoreLink, CoreSight, Cortex, DesignStart, DynamIQ, Jazelle, Keil, Mali, Mbed, Mbed Enabled, NEON, POP, RealView, SecurCore, Socrates, Thumb, TrustZone, ULINK, ULINK2, ULINK-ME, ULINK-PLUS, ULINKpro, µVision, Versatile are trademarks or registered trademarks of Arm Limited (or its subsidiaries) in the US and/or elsewhere. The related technology may be protected by any or all of patents, copyrights, designs and trade secrets. All rights reserved.

© NXP B.V. 2018-2023.

All rights reserved.

For more information, please visit: <http://www.nxp.com>

For sales office addresses, please send an email to: [salesaddresses@nxp.com](mailto:salesaddresses@nxp.com)

Date of release: 16/06/2023  
Identifier: LPC55S2x/LPC552x

