LPC51U68

Errata sheet LPC51U68

Rev. 1.4 — 11 June 2024 Errata

Document information

Information	Content
Keywords	LPC51U68JBD48, LPC51U68JBD64
Abstract	LPC51U68 errata

1 Product identification

The LPC51U68 LQFP48 and LQFP64 packages have the following top-side marking:

- First line: LPC51U68
- Second line: JBD48
- Third line: xx xx
- Fourth line: xxxyy
- Fifth: wwxR[x]
  - yyww: Date code with yy = year and ww = week.
  - xR = Boot code version and device revision.
- First line: LPC51U68
- Second line: JBD64
- Third line: xxxxxxxxxxxx
- Fourth line: xxxyywwx[R]x
  - yyww: Date code with yy = year and ww = week.
  - xR = Boot code version and device revision.

Table 1. Device revision table

Device revision	Revision description
0A	Initial device revision with boot code version 18.0.

2 Errata overview

Table 2. Functional problems table

Functional problems	Short description	Revision identifier	Detailed description
ISP.1	ISP (In-System Programming) command for UID (unique identification number) is not functional.	'A'	Section 3.1
ISP.2	ISP 'Z' command is not reliable for flash signature generation.	'A'	Section 3.2
ADC.1	High current consumption in reduced low power modes when using ADC.	'A'	Section 3.3
USB.1	Automatic USB rate adjustment not functional when using multiple hubs.	'A'	Section 3.4
I^2S.1	FIFO underflow interrupt not generated for I^2S peripheral.	'A'	Section 3.5
I^2S.2	The Most Significant Bit (MSB) of I^2S receive data is forced to 0 if DATALEN is greater than 23.	'A'	Section 3.6
I^2S.3	Incorrect synchronization to the second edge of the WS instead of the start of frame.	'A'	Section 3.7
I^2C.1	The AUTOACK feature does not work reliably when the CPU system clock frequency is three times or more than the peripheral clock to the I^2C interface.	'A'	Section 3.8
USART.1	The USART receiver timeout feature is not supported.	'A'	Section 3.9
USART.2	The USART receiver idle (RXIDLE) interrupt feature is not supported.	'A'	Section 3.10

Table 2. Functional problems table...*continued*

Functional problems	Short description	Revision identifier	Detailed description
CRP.1	Code read protection level 1 is not functional.	'A'	Section 3.11
USB_ROM.1	FRAME_INT is cleared if new SetConfiguration or USB_RESET are received.	'A'	Section 3.12
USB_ROM.2	USB full-speed device fail in the Command/Data/ Status Flow after bus reset and bus re-enumeration.	'A'	Section 3.13
PLL.1	P-divider set to 4 could generate the wrong output frequency from the PLL.	'A'	Section 3.14

Table 3. AC/DC deviations table

AC/DC deviations	Short description	Revision identifier	Detailed description
n/a	n/a	n/a	n/a

Table 4. Errata notes

Note	Short description	Revision identifier	Detailed description
n/a	n/a	n/a	n/a

3 Functional problems detail

3.1 ISP.1: ISP (In-System Programming) command for UID (unique identification number) is not functional.

Introduction:

Each LPC51U68 device contains a device serial number (four 32-bit words) for unique identification. The ISP call (ReadUID) can be performed via the USART interface to read the unique serial number where the word at the lowest address is sent first.

Problem:

On the LPC51U68, the read UID ISP command is not functional.

Work-around:

The unique serial number (four 32-bit words) can be directly read from address locations 0x0100 0100 to 0x0100 010C.

3.2 ISP.2: ISP 'Z' command is not reliable for flash signature generation

Introduction:

On the LPC51U68 devices, the ISP 'Z' command is used to determine the signature of the entire flash memory using an internal flash signature generator.

Problem:

The ISP ‘Z’ command is not reliable on the LPC51U68 devices. In most cases, the signature will be correct but due to a signature generation timing sensitivity, the result may be incorrect and therefore unreliable.

Work-around:

Use the flash signature generation registers documented in the LPC51U68 User Manual to create the signature of the entire flash memory.

3.3 ADC.1: High current consumption in reduced low power modes when using ADC.

Introduction:

The 12-bit ADC controller is available on all LPC51U68 parts. The ADC can measure the voltage on any of the input signals on the analog input channel. For accurate voltage readings, the digital pin function on the ADC input channel must be disabled by writing a 0 to the DIGIMODE bit in the related IOCON register. This enables the analog mode functionality on the ADC input channel.

Problem:

For applications using the ADC, the current consumption could be higher than expected in reduced power modes (deep-sleep and deep power-down modes) or when the ADC is disabled using the PDRUNCFG register.

Work-around:

To prevent high current consumption, use the following steps in the software:

1. Following a chip reset, all 12 ADC input channels (ADC0_0 to ADC0_11) should be in Digital Mode (DIGIMODE = 1) in the related IOCON registers until the configuration of the ADC block is complete. See the Basic Configuration section in the LPC51U68 12-bit ADC controller (ADC) chapter of the LPC51U68 User Manual
2. After configuring the ADC, change only those pins that are used as ADC input channels to Analog Mode (DIGIMODE = 0) in the related IOCON registers before starting ADC conversions.
3. Before entering any reduced power mode (deep-sleep and deep power-down) or before powering down the ADC block (by writing to the PDEN_ADC0 bit in the PDRUNCFG register), the ADC input channel(s) must be changed back to Digital Mode.
4. After waking up from the reduced power mode or when re-enabling the ADC block (PDEN_ADC0 bit in the PDRUNCFG), the software must follow step 2 before starting ADC conversions.

3.4 USB.1: 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:

LPC51U68 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 FROCTRL register). The problem does not occur when a single hub is used.

Work-around:

Use the FRO calibration library provided in TN00035. This library allows the application to have a crystal-less USB operation in full-speed mode.

3.5 I2S.1: FIFO underflow interrupt not generated for I^2S peripheral

Introduction:

Multiple Flexcomm Interfaces are available in the LPC51U68 devices. Flexcomm Interface 6 and Flexcomm Interface 7 can be configured for I^2S peripheral function and the data for all I^2S traffic within one Flexcomm Interface uses the Flexcomm Interface FIFO. During I^2S data transfers, when the transmit FIFO is empty, a FIFO underflow occurs and an interrupt is generated, which is flagged by the UNDERRUN bit in the I^2S FIFOSTAT register.

Problem:

When the FIFO underflow condition occurs, the interrupt from the I^2S peripheral function might not be generated and as a result, the UNDERRUN bit does not get set. This issue does not affect the SPI and USART peripherals.

Work-around:

There is no work-around.

3.6 I^2S.1: The Most Significant Bit (MSB) of I^2S receive data is forced to zero if DATALEN > 23

Introduction:

On the LPC51U68 devices, the I^2S function is included in Flexcomm Interface 6 and Flexcomm Interface 7. Each of these Flexcomm Interfaces implements one I^2S channel pair. The Data Length (DATALEN) defines the number of data bits to be transmitted or received for all I^2S channel pairs.

Problem:

If the I^2S interface is configured for DATALEN (in I^2S CFG1 register) greater than 23 (25-bit data or greater), the MSB of any received data will be forced to 0. If DATALEN = 24 (25-bit data), bit 24 of received data will always be 0. If DATALEN = 31 (32-bit data), bit 31 of received data will always be 0. The issue occurs regardless of the I^2S operating mode (selected by MODE bits).

Work-around:

There is no work-around.

3.7 I²S.3: Incorrect synchronization to the second edge of the Word Select (WS) signal instead of the start of frame.

Introduction:

Multiple Flexcomm Interfaces are available in the LPC51U68 devices. Flexcomm Interface 6 and Flexcomm Interface 7 can be configured for I²S peripheral function. The Word Select (WS) pin is the synchronizing signal for the beginning of each data frame and in some modes, left versus right channel data. It is normally driven by the master and received by one or more slaves.

Problem:

When I²S receives the WS signal from another master, that is, if MSTSLVCFG = 0x1 (WS Synchronized master) or 0x0 (Normal Slave mode) in the CFG1 register, and the I²S bus is already running, it can incorrectly synchronize to the second edge of the WS instead of the start of frame. This does not happen when the slave is started before the WS master.

Work-around:

The work-around is to detect the trailing edge on the WS pin and then enable the I²S interface. This can be achieved by using a rising or falling edge interrupt on a pin. The edge is determined by the starting mode of the WS pin. In I²S mode, the rising edge interrupt can be used so that the data frame starts from the falling edge. In DSP mode, the falling edge interrupt can be used so that the data frame starts from the rising edge.

3.8 I²C.1

Introduction:

In LPC51U68 devices, the I²C interface has an AUTOACK bit in the Slave Control register. In the slave mode, when this bit is set, it will cause an I²C header, which matches the slave address SLVADR0 and the direction set by the AUTOMATCHREAD to be ACKed immediately. This is used with the DMA to allow processing of the data without intervention.

Problem:

The AUTOACK feature does not work reliably when the CPU system clock frequency is three times or more than the peripheral clock to the I²C interface.

Work-around:

The I²C peripheral clock frequency should be the same or half of the CPU system clock.

3.9 USART.1

Introduction:

A receiver timeout feature for the USART provides a means to get data left for a time in a FIFO that has not reached its threshold to be transferred. This feature exists in LPC5410x devices.

Problem:

The LPC51U68 devices do not support the USART receiver timeout feature.

Work-around:

Timer0 can be used as a USART RX timeout timer and Flexcomm0 as USART0 peripheral in loop back mode. See the technical note TN00013 for more details.

3.10 USART.2

Introduction:

In the USART peripheral, the receiver idle (RXIDLE) interrupt occurs when the RX channel becomes idle. This feature exists in LPC5410x devices.

Problem:

The LPC51U68 devices do not support the USART receiver idle (RXIDLE) interrupt feature.

Work-around:

There is no work-around.

3.11 CRP.1: Code read protection level 1 is not functional

Introduction:

Code Read Protection is a mechanism that allows the user to enable different levels of security in the system so that access to the on-chip flash and use of the ISP can be restricted. When needed, CRP is invoked by programming a specific pattern in the flash image at offset 0x0000 02FC. There are three levels of code read protection available to the user.

For CRP1 level, erase page command can erase pages in sector 0 only when all pages in the user flash are selected for erase.

Problem:

All pages in sector 0 except page 0 can be erased, which results in erasing CRP1 level.

Work-around:

Use CRP2 level or CRP3 level for code read protection.

3.12 USB_ROM.1: FRAME_INT is cleared if new SetConfiguration or USB_RESET are received.

Introduction:

In the USB ROM API, the function call EnableEvent can be used to enable and disable FRAME_INT.

Problem:

When the FRAME_INT is enabled through the USB ROM API call:

```
ErrorCode_t(* USBD_HW_API::EnableEvent)(USBD_HANDLE_T hUsb, uint32_t EPNum,
 uint32_t event_type, uint32_t enable),
```

the FRAME_INT is cleared if new SetConfiguration or USB_RESET are received.

Work-around:

Implement the following software work-around in the ISR to ensure that the FRAME_INT is enabled:

```
void USB_IRQHandler(void)
{
USBD_API->hw->EnableEvent(g_hUsb, 0, USB_EVT_SOF, 1);
USBD_API->hw->ISR(g_hUsb);
}
```

3.13 USB_ROM.2: USB full-speed device fail in the Command/Data/Status Flow after bus reset and bus re-enumeration

Introduction:

The LPC51U68 device family includes a USB full-speed interface that can operate in device mode and also, includes USB ROM based drivers. A Bulk-Only Protocol transaction begins with the host sending a CBW to the device and attempting to make the appropriate data transfer (In, Out or none). The device receives the CBW, checks and interprets it, attempts to satisfy the request of the host, and returns status via a CSW.

Problem:

When the device fails in the Command/Data/Status Flow, and the host does a bus reset / bus re-enumeration without issuing a Bulk-Only Mass Storage Reset, the USB ROM driver does not re-initialize the MSC variables. This causes the device to fail in the Command/Data/Status Flow after the bus reset / bus re-enumeration.

Work-around:

Implement the following software work-around to re-initialize the MSC variables in the USBD stack.

```
void        *g_pMscCtrl;
ErrorCode_t mwMSC_Reset_workaround(USBD_HANDLE_T hUsb)
{
((USB_MSC_CTRL_T *)g_pMscCtrl)->CSW.dSignature = 0;
        ((USB_MSC_CTRL_T *)g_pMscCtrl)->BulkStage = 0;
        return LPC_OK;
}
ErrorCode_t mscDisk_init(USBD_HANDLE_T hUsb, USB_CORE_DESCS_T *pDesc,
 USBD_API_INIT_PARAM_T *pUsbParam)
{        USBD_MSC_INIT_PARAM_T msc_param;
        ErrorCode_t ret = LPC_OK;
        memset((void *) &msc_param, 0, sizeof(USBD_MSC_INIT_PARAM_T));
        msc_param.mem_base = pUsbParam->mem_base;
        msc_param.mem_size = pUsbParam->mem_size;
        g_pMscCtrl = (void *)msc_param.mem_base;
        ret = USBD_API->msc->init(hUsb, &msc_param);
```

```
        /* update memory variables */
        pUsbParam->mem_base = msc_param.mem_base;
        pUsbParam->mem_size = msc_param.mem_size;
        return ret;
}
        usb_param.USB_Reset_Event = mwMSC_Reset_workaround;
        ret = USBD_API->hw->Init(&g_hUsb, &desc, &usb_param);
```

3.14 PLL.1: P-divider set to 4 could generate the wrong output frequency from the PLL

Introduction:

On the LPC51U68 PLL, the Fcco frequency must be either the actual desired output frequency, or the desired output frequency times 2 x P, where P is range from 1 to 32 (2^5). The Fcco frequency must also be a multiple of the PLL reference frequency, which is either the PLL input, or the PLL input divided by N, where N is from 2 to 256.

Problem:

The P-divider when set for divide by 4 mode can erroneously arrive in divide by 2 mode. The high frequency spikes coming from the level shifter during startup of the PLL can cause the P-divider to jump into the wrong division state only when set in divide by 4 mode. This issue affects both the System PLL and Audio PLL.

Work-around:

Use other P values other than 4 to achieve desired output frequency. P = 1 – 32 and P ≠ 4.

4 AC/DC deviations detail

No known errata.

5 Errata notes

No known errata.

6 Revision history

Table 5. Revision history

Document ID	Release Date	Description
ES_LPC51U68 v.1.4	11 June 2024	• Added Section 3.14.
ES_LPC51U68 v.1.3	17 May 2018	• Removed ES from the document identifier variable.
ES_LPC51U68 v.1.2	14 March 2018	• Added work-around for Section 3.4
ES_LPC51U68 v.1.1	9 March 2018	• Added Section 3.13
ES_LPC51U68 v.1	15 December 2017	• Initial version

7 Note about the source code in the document

Example code shown in this document has the following copyright and BSD-3-Clause license:

Legal information

Definitions

Draft — A draft status on a document indicates that the content is still under internal review and subject to formal approval, which may result in modifications or additions. NXP Semiconductors does not give any representations or warranties as to the accuracy or completeness of information included in a draft version of a document and shall have no liability for the consequences of use of such information.

Disclaimers

Limited warranty and liability — Information in this document is believed to be accurate and reliable. However, NXP Semiconductors does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information and shall have no liability for the consequences of use of such information. NXP Semiconductors takes no responsibility for the content in this document if provided by an information source outside of NXP Semiconductors.

In no event shall NXP Semiconductors be liable for any indirect, incidental, punitive, special or consequential damages (including - without limitation - lost profits, lost savings, business interruption, costs related to the removal or replacement of any products or rework charges) whether or not such damages are based on tort (including negligence), warranty, breach of contract or any other legal theory.

Notwithstanding any damages that customer might incur for any reason whatsoever, NXP Semiconductors' aggregate and cumulative liability towards customer for the products described herein shall be limited in accordance with the Terms and conditions of commercial sale of NXP Semiconductors.

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.

Suitability for use — NXP Semiconductors products are not designed, authorized or warranted to be suitable for use in life support, life-critical or safety-critical systems or equipment, nor in applications where failure or malfunction of an NXP Semiconductors product can reasonably be expected to result in personal injury, death or severe property or environmental damage. NXP Semiconductors and its suppliers accept no liability for inclusion and/or use of NXP Semiconductors products in such equipment or applications and therefore such inclusion and/or use is at the customer's own risk.

Applications — Applications that are described herein for any of these products are for illustrative purposes only. NXP Semiconductors makes no representation or warranty that such applications will be suitable for the specified use without further testing or modification.

Customers are responsible for the design and operation of their applications and products using NXP Semiconductors products, and NXP Semiconductors accepts no liability for any assistance with applications or customer product design. It is customer's sole responsibility to determine whether the NXP Semiconductors product is suitable and fit for the customer's applications and products planned, as well as for the planned application and use of customer's third party customer(s). Customers should provide appropriate design and operating safeguards to minimize the risks associated with their applications and products.

NXP Semiconductors does not accept any liability related to any default, damage, costs or problem which is based on any weakness or default in the customer's applications or products, or the application or use by customer's third party customer(s). Customer is responsible for doing all necessary testing for the customer's applications and products using NXP Semiconductors products in order to avoid a default of the applications and the products or of the application or use by customer's third party customer(s). NXP does not accept any liability in this respect.

Terms and conditions of commercial sale — NXP Semiconductors products are sold subject to the general terms and conditions of commercial sale, as published at https://www.nxp.com/profile/terms, unless otherwise agreed in a valid written individual agreement. In case an individual agreement is concluded only the terms and conditions of the respective agreement shall apply. NXP Semiconductors hereby expressly objects to applying the customer's general terms and conditions with regard to the purchase of NXP Semiconductors products by customer.

Export control — This document as well as the item(s) described herein may be subject to export control regulations. Export might require a prior authorization from competent authorities.

Suitability for use in non-automotive qualified products — Unless this document expressly states that this specific NXP Semiconductors product is automotive qualified, the product is not suitable for automotive use. It is neither qualified nor tested in accordance with automotive testing or application requirements. NXP Semiconductors accepts no liability for inclusion and/or use of non-automotive qualified products in automotive equipment or applications.

In the event that customer uses the product for design-in and use in automotive applications to automotive specifications and standards, customer (a) shall use the product without NXP Semiconductors' warranty of the product for such automotive applications, use and specifications, and (b) whenever customer uses the product for automotive applications beyond NXP Semiconductors' specifications such use shall be solely at customer's own risk, and (c) customer fully indemnifies NXP Semiconductors for any liability, damages or failed product claims resulting from customer design and use of the product for automotive applications beyond NXP Semiconductors' standard warranty and NXP Semiconductors' product specifications.

HTML publications — An HTML version, if available, of this document is provided as a courtesy. Definitive information is contained in the applicable document in PDF format. If there is a discrepancy between the HTML document and the PDF document, the PDF document has priority.

Translations — A non-English (translated) version of a document, including the legal information in that document, is for reference only. The English version shall prevail in case of any discrepancy between the translated and English versions.

Security — Customer understands that all NXP products may be subject to unidentified vulnerabilities or may support established security standards or specifications with known limitations. 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) that manages the investigation, reporting, and solution release to security vulnerabilities of NXP products.

NXP B.V. — NXP B.V. is not an operating company and it does not distribute or sell products.

Trademarks

Notice: All referenced brands, product names, service names, and trademarks are the property of their respective owners.

NXP — wordmark and logo are trademarks of NXP B.V.

Contents