ES_LPC540xx

Errata sheet LPC540xx

Rev. 1.4 — 21 March 2018 Errata sheet

Document information

Info	Content
Keywords	LPC54018JET180, LPC54018JBD208, LPC54016JET180, LPC54016JBD208, LPC54016JBD100, LPC54005JET100, LPC54005JBD100
Abstract	LPC540xx errata

Revision history

Rev	Date	Description
v.1.4	20180321	• ADC.1 • SHA.1
v.1.3	20180313	• USB.2
v.1.2	20180302	• Added ROM.1 • Added USB.ROM.1
v.1.1	20170104	• Added IOCON.1
v.1	20171205	• Initial version.

Contact information

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

For sales office addresses, please send an email to: **salesaddresses@nxp.com**

1. Product identification

The ES_LPC540xx TFBGA180 and TFBGA100 packages have the following top-side marking:

- First line: ES_LPC540xxJ
- Second line: ET180 or ET100
- Third line: xxxxxxxxxxxx
- Fourth line: xxxyywwx[R]x
  - yyww: Date code with yy = year and ww = week.
  - x[R] = boot code version and device revision.

The ES_LPC540xx LQFP208 and LQFP100 packages have the following top-side marking:

- First line: ES_LPC540xxJ
- Second line: BD208 or BD100
- Third line: xxxxxxxxxxxx
- Fourth line: xxxyywwx[R]x
  - yyww: Date code with yy = year and ww = week.
  - x[R] = Boot code version and device revision.

Table 1. Device revision table

Revision identifier (R)	Revision description
0A	Initial device revision with Boot ROM version 21.0.

2. Errata overview

Table 2. Functional problems table

Functional problems	Short description	Revision identifier	Detailed description
ISP.1	Reinvoke ISP using USB0 DFU and USB1 DFU interfaces are not functional.	0A	Section 3.1
OTP.1	SWD interface cannot be disabled using OTP bits.	0A	Section 3.2
OTP.2	OTP APIs are not functional.	0A	Section 3.3
USB.1	Resetting interrupt endpoint resets DATAx sequence to DATA.1.	0A	Section 3.4
USB.2	In USB full-speed device mode, the ROOT2 endpoint test fails.	0A	Section 3.5
IOCON.1	On power-up the standard GPIO pins are not in high Z mode by default.	0A	Section 3.6
ROM.1	On boot failure peripheral pins remain configured or drive	0A	Section 3.7
USB.ROM.1	USB full-speed device fail in the Command/Data/Status Flow after bus reset and bus re-enumeration.	0A	Section 3.8
ADC.1	High current consumption in reduced low power modes when using ADC.	0A	Section 3.9
SHA.1	Using MEMCTRL after DIGEST Ready to include more blocks via Mastering does not clear DIGEST bit.	0A	Section 3.10

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: Reinvoke ISP using USB0 DFU and USB1 DFU interfaces are not functional

Introduction:

On the LPC540xx, reinvoke ISP command is available to invoke the bootloader in ISP mode. This command may be used when a valid user program is present in the internal SRAM memory and the ISP entry pins are not accessible to force the ISP mode. ISP communication can be selected via auto-detect, I2C0 (Flexcomm Interface 1), SPI0 (Flexcomm Interface 3), SPI1 (Flexcomm Interface 10), UART0 (Flexcomm Interface 0), USB0 DFU, and USB1 DFU interfaces.

Problem:

Options to select USB0 DFU, and USB1 DFU interfaces does not work.

Work-around:

There is no work-around. This will be fixed on the next silicon revision.

3.2 OTP.1: SWD interface cannot be disabled using OTP bits

Introduction:

On the LPC540xx devices, bits 6 and 13 (SWD_JTAG_ENx) of the OTP memory bank 3, word 0, can be used to disable SWD access.

Problem:

Setting these bits to disable SWD is not functional.

Work-around:

There is no work-around. This will be fixed on the next silicon revision.

3.3 OTP.2: OTP APIs are not functional

Introduction:

On the LPC540xx devices, the Boot ROM provides API support for programming the OTP.

Problem:

The OTP programming APIs are not functional.

Work-around:

There is no work-around. This will be fixed on the next silicon revision.

3.4 USB.1: Resetting interrupt endpoint resets DATAx sequence to DATA.1

Introduction:

The LPC540xx includes a USB High Speed interface (USB1) that can operate in device mode at high speed. The T bit in command/status list determines if the endpoint type is generic endpoint or periodic endpoint. When the endpoint type is set to periodic, the RF/TV bit in command/status list determines if the endpoint type is isochronous endpoint or interrupt endpoint. When the TR bit in command/status list is set to ‘1’, the toggle value will set to the value indicated in the RF/TV bit.

Problem:

When the endpoint type is set to interrupt with T bit as ‘1’ and RF/TV bit as ‘1’, the data toggle for the interrupt endpoint with TR bit as ‘1’ resets to DATA1.

Work-around:

For applications that have strict requirements of the data toggle value, the following is the the software work-around:

1. Set INTONNAK_AO and INTONNAK_AI bits to ‘1’ in the Device Command/Status register.
2. Set A=0, TR=1, RF/TV=0, T=0 (this will force the device to return a NAK handshake and reset the internal toggle value to zero).
3. Wait until an interrupt is received. Read the Endpoint Toggle register and check the value of the endpoint toggle. If the toggle is reset to ‘0’, then go to step 4 else wait for the next interrupt.
4. Set A=1, TR=0, RF/TV=1, T=1 (the endpoint is back to the normal operation)

The result of this work-around is when an endpoint is reset, a NAK handshake is returned on the first received token.

3.5 USB.2: In USB full-speed device mode, the ROOT2 endpoint test fails

Introduction:

The LPC540xx includes a USB full speed interface (USB0) that can operate in device mode at full speed. It supports 10 physical (5 logical) endpoints including control endpoints. The device should not respond to those endpoints which are not supported.

Problem:

The device NAKed the OUT token addressed to an endpoint that is not present on the device causing the ROOT2 endpoint test to fail.

Work-around:

There is no work-around.

3.6 IOCON.1: On power-up the standard GPIO pins are not in high Z mode by default

Introduction:

On the LPC540xx devices, on power-up, pins PIO0_0 to PIO0_31, PIO1_0 to PIO1_31, PIO2_0 to PIO2_31, PIO3_0 to PIO3_31, PIO4_0 to PIO4_31, and PIO5_0 to PIO5_10 are in high Z mode by default (internal pull-up resistor and internal pull-down resistor are disabled).

Problem:

Only on the LPC540xx device revision 0A (Boot ROM version 21.0), on power-up, pins PIO0_0 to PIO0_31, PIO1_0 to PIO1_31, PIO2_0 to PIO2_31, PIO3_0 to PIO3_31, PIO4_0 to PIO4_31, and PIO5_0 to PIO5_10 have the internal pull-ups enabled and are not in high Z mode by default. Depending on the application, this can cause a higher in-rush current and/or external circuits connected to the GPIO pins to work in an unexpected manner.

Work-around:

This issue will be fixed in the next device revision 1B (Boot ROM version 21.1). Users should carefully design their application and consider the impact of the GPIOs’ default change from device revision 0A to device revision 1B. The user can identify the device revisions by reading the DEVICE_ID1 register or by calling the Boot ROM version ISP/IAP function.

3.7 ROM.1: On boot failure peripheral pins remain configured or driven

Introduction:

On the LPC540xx devices, images can be booted into on-chip SRAM from external flash (SPI, QSPI, or parallel flash) or downloaded via the serial ports (UART, I2C, SPI, USB0, USB1). Depending on the values of the OTP bits and ISP pins, and the image header type definition, the bootloader decides whether to download code into the on-chip SRAM or run from external memory. If OTP BOOT_SRC bits are not set and the ISP pins (PIO0_4, PIO0_5, PIO0_6) are all HIGH, the LPC540xx will perform auto boot and look for valid image in the following order: external SPIFI flash device, external SPI flash, and external parallel flash memory.

Problem:

If LPC540xx Rev 0A devices boot in auto mode, pins may remain configured and even driven depending on boot failure:

- If a SPIFI or SPI flash is populated, but not programmed, after an unsuccessful boot attempt, the SPIFI/SPI pins are disabled and placed in a pull-up state.
- If neither a SPIFI or SPI flash is populated, SPIFI/SPI pins continue to remain configured after boot attempt. If those pins are used for other purposes after boot, such as GPIO, their respective IOCON registers will require reconfiguration. Also, the SPIFI and SPI clocks are still enabled. The affected pins include PIO0_23 to PIO0_28.
- If a parallel flash is not present or is present, but is not programmed with a valid image, the EMC (parallel) pins continue to remain configured after a boot attempt.
- The following EMC port pins remain configured for EMC. If these pins are used for other purposes after boot, such as GPIO, or if there are power related concerns, their respective IOCON registers will require reconfiguration. Also the EMC clocks are still enabled.
  - Output pins:

    PIO0_15 to PIO0_21

    PIO1_22 to PIO1_27

    PIO1_15 to PIO1_18

    PIO3_10

    PIO3_12 to PIO3_14

    PIO3_23 to PIO3_31

    PIO4_0 to PIO4_6

    PIO4_17 to PIO4_20

    PIO5_5 to PIO5_9
  - I/O (Tri-state) pins:

    PIO0_2 to PIO0_9

    PIO1_4

    PIO1_19 to PIO1_21

    PIO1_28 to PIO1_31

    PIO4_21 to PIO4_31

PIO5_0 to PIO5_4

Work-around:

1. In Auto boot mode, program SPIFI or SPI flash with a valid image so that boot completes and does not attempt EMC boot.
2. Choose a different ISP mode such as SPI, SPIFI or serial boot mode, especially if developing code using debugger (download image directly to SRAM). This configures less number of pins compared to auto mode.

The issue occurs only on LPC540xx Rev 0A devices and will be fixed in Rev 1B silicon.

3.8 USB_ROM.1: USB full-speed device fail in the Command/Data/Status Flow after bus reset and bus re-enumeration

Introduction:

The LPC540xx 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);
```

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

Introduction:

The 12-bit ADC controller is available on all LPC540xx. 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 LPC540xx. 12-bit ADC controller (ADC) chapter of the LPC540xx. 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.

3.10 SHA.1: Using MEMCTRL after DIGEST Ready to include more blocks via Mastering does not clear DIGEST bit.

Introduction:

The LPC540xx includes a SHA hash block to compute SHA1 and SHA2-256 hash digests on flash images or messages in RAM. For maximum performance and ease of use, the hash block includes a master on the internal buses of the chip to read multiple blocks of memory while hashing, without involvement of the processor. This mastering model permits hashing up to 128 K bytes of memory (Flash, RAM, or SPI Flash).

Problem:

If the application uses the mastering on up to 128 K bytes and then uses it for additional blocks (without starting new), the DIGEST (digest ready) status does not clear when starting the next sequence via mastering. If the processor or DMA is used for the additional blocks, the DIGEST status is cleared.

Work-around:

If the purpose for the additional block(s) is to hash the last block (with padding and length), then the processor or DMA may be used to write the 16 words via INDATA, and the DIGEST status will clear when the 1st word is written.

If the purpose for additional blocks is to do a large number of blocks (for example, after doing 128 K, another 64 K is to be hashed), then the 1st block may be started by the processor (that is, the processor writes the 16 words to INDATA) followed by configuring MEMADDR and MEMCTRL for the remaining blocks. The MEMCTRL should be written within 64 cycles of writing the last word to INDATA to ensure DIGEST is 0.

4. AC/DC deviations detail

No known errata.

5. Errata notes

No known errata.

6. Legal information

6.1 Definitions

Draft — The document is a draft version only. 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 herein and shall have no liability for the consequences of use of such information.

6.2 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.

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 accepts 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.

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 national authorities.

6.3 Trademarks

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

7. Contents

1 Product identification 3

2 Errata overview 4

3 Functional problems detail 5

3.1 ISP.1: Reinvoke ISP using USB0 DFU and USB1 DFU interfaces are not functional 5

3.2 OTP.1: SWD interface cannot be disabled using OTP bits 5

3.3 OTP.2: OTP APIs are not functional 6

3.4 USB.1: Resetting interrupt endpoint resets DATAx sequence to DATA.1 7

3.5 USB.2: In USB full-speed device mode, the ROOT2 endpoint test fails 8

3.6 IOCON.1: On power-up the standard GPIO pins are not in high Z mode by default 9

3.7 ROM.1: On boot failure peripheral pins remain configured or driven 10

3.8 USB_ROM.1: USB full-speed device fail in the Command/Data/Status Flow after bus reset and bus re-enumeration 12

3.9 ADC.1: High current consumption in reduced low power modes when using ADC. 13

3.10 SHA.1: Using MEMCTRL after DIGEST Ready to include more blocks via Mastering does not clear DIGEST bit. 14

4 AC/DC deviations detail 15

5 Errata notes 15

6 Legal information. 16

6.1 Definitions 16

6.2 Disclaimers 16

6.3 Trademarks. 16

7 Contents 17

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

For more information, please visit: http://www.nxp.com
For sales office addresses, please send an email to: salesaddresses@nxp.com

Date of release: 21 March 2018

Document identifier: ES_LPC540xx