# ARM PrimeCell
# Universal Asynchronous Receiver Transmitter (PL011)
# **Errata Notice**

This document contains all errata known at the date of issue in releases up to and including revisionr1p5 of PL011 UART -Perpetual.

## Proprietary notice



## Document confidentiality status

This document is Non Confidential.

## Web address

**http://www.arm.com/**

## Feedback on the product

If you have any comments or suggestions about this product, contact your supplier giving:

- The product name
- A concise explanation of your comments.

## Feedback on this document

If you have any comments on about this document, please send email to mailto:errata@arm.com giving:

- The document title
- The documents number
- The page number(s) to which your comments refer
- A concise explanation of your comments

General suggestion for additions and improvements are also welcome.

# Contents

# Introduction

## Scope

This document describes errata categorised by level of severity. Each description includes:

- the current status of the defect
- where the implementation deviates from the specification and the conditions under which erroneous behavior occurs
- the implications of the erratum with respect to typical applications
- the application and limitations of a 'work-around' where possible

## Categorisation of Errata

Errata recorded in this document are split into three levels of severity:

Category 1 Behavior that is impossible to work around and that severely restricts the use of the product in all, or the majority of applications, rendering the device unusable.

Category 2 Behavior that contravenes the specified behavior and that might limit or severely impair the intended use of specified features, but does not render the product unusable in all or the majority of applications.

Category 3 Behavior that was not the originally intended behavior but should not cause any problems in applications.

## Change Control

**04 Feb 2008: Changes in Document v6**

| Page | Status | ID | Cat | Summary |
|---|---|---|---|---|
| 18 | Updated | 459631 | Cat 2 | UART Break error issue |

**23 Jan 2008: Changes in Document v5**

| Page | Status | ID | Cat | Summary |
|---|---|---|---|---|
| 18 | Updated | 459631 | Cat 2 | UART Break error issue |
| 18 | Updated | 333824 | Cat 3 | Two Parameters Instead of Three Are Being Passed To Clockgen |
| 23 | Updated | 406210 | Doc | DDI0183F UART (PL011) r1p4 TRM - UARTPeriphID2 revision bits are incorrect |

**29 Sep 2006: Changes in Document v4**

| Page | Status | ID | Cat | Summary |
|---|---|---|---|---|
| 23 | New | 406210 | Doc | TRM Error with UARTPeriphID2 revision |

**02 Nov 2005: Changes in Document v3**

| Page | Status | ID | Cat | Summary |
|---|---|---|---|---|
| 17 | New | 346440 | Cat 2 | FIFO overflow not correctly flagged |
| 21 | Updated | 330540 | Doc | TRM Error with UARTPeriphID2 revision |
| 20 | Updated | 326409 | Doc | Incorrect Information in TRM for register UARTILPR |

**28 Oct 2004: Changes in Document v2**

| Page | Status | ID | Cat | Summary |
|---|---|---|---|---|
| 9 | New | 333822 | Cat 1 | DMA Signals Not Synchronised to UART PCLK |
| 10 | New | 333829 | Cat 1 | Uart Ignores the Disable Function |
| 11 | New | 333831 | Cat 1 | Uart CharTxComp Signal Is Not Synchronised |
| 12 | New | 333821 | Cat 2 | Fractional Baud Rate Divisor Does Not Function With Some Divisor Values |
| 15 | New | 333823 | Cat 2 | IrDA low power counter consumes power when IrDA mode is disabled |
| 16 | New | 333826 | Cat 2 | Some redundant logic in the Integration test logic |
| 19 | New | 333824 | Cat 3 | Two Parameters Instead of Three Are Being Passed To Clockgen |
| 20 | Updated | 326409 | Doc | Incorrect Information in TRM for register UARTILPR |
| 21 | New | 330540 | Doc | TRM Error with UARTPeriphID2 revision |
| 22 | New | 333827 | Doc | Design Manual Documentation Error |
| 24 | Updated | 179060 | Soft | Driver code reads a write only register |

**08 Jun 2004: Changes in Document v1**

| Page | Status | ID | Cat | Summary |
|---|---|---|---|---|
| 20 | New | 326409 | Doc | Incorrect Information in TRM for register UARTILPR |

| | | | | |
|---|---|---|---|---|
| 24 | New | 179060 | Soft | Driver code reads a write only register |

# Errata Summary Table

The errata associated with this product affect product versions as below.

A cell shown thus **X** indicates that the defect affects the revision shown at the top of that column.

| ID | Cat | Summary of Erratum | 1-0 | 1-v0 | 1-v1 | 1-v3 | r1p4-00rel0 | r1p5-00rel0 |
|---|---|---|---|---|---|---|---|---|
| 179060 | Soft | Driver code reads a write only register | | | | X | X | X |
| 326409 | Doc | Incorrect Information in TRM for register UARTILPR | | | | X | | |
| 330540 | Doc | TRM Error with UARTPeriphID2 revision | | | | X | | |
| 333827 | Doc | Design Manual Documentation Error | | X | X | | | |
| 406210 | Doc | DDI0183F UART (PL011) r1p4 TRM - UARTPeriphID2 revision bits are incorrect | | | | | X | |
| 333822 | Cat 1 | DMA Signals Not Synchronised to UART PCLK | | X | | | | |
| 333829 | Cat 1 | Uart Ignores the Disable Function | | | X | | | |
| 333831 | Cat 1 | Uart CharTxComp Signal Is Not Synchronised | | | X | | | |
| 333821 | Cat 2 | Fractional Baud Rate Divisor Does Not Function With Some Divisor Values | | X | | | | |
| 333823 | Cat 2 | IrDA low power counter consumes power when IrDA mode is disabled | | X | | | | |
| 333826 | Cat 2 | Some redundant logic in the Integration test logic | | X | | | | |
| 346440 | Cat 2 | FIFO overflow not correctly flagged | X | X | X | X | | |
| 459631 | Cat 2 | UART Break error issue | X | X | X | X | X | |
| 333824 | Cat 3 | Two Parameters Instead of Three Are Being Passed To Clockgen | | X | X | X | X | |

# Errata - Category 1

## 333822: DMA Signals Not Synchronised to UART PCLK

### Status

Affects: product PL011 UART -Perpetual.

Fault status: Cat 1, Present in: 1-v0, Fixed in 1-v1.

### Description

The UARTTXDMACLR and UARTRXDMACLR from the DMA controller are not being synchronised to the UARTPCLK clock domain.

The DMA capable peripheral requires additional external synchronisation logic on the DMACLR and DMATC signals if the peripherals do not use the same clock as the DMA controller. This synchronisation logic should ideally be in the peripheral itself.

This synchronisation logic can be either always used, or enabled and disabled using a register bit.

### Affected Items

Uart.v/vhd

UartApif.v/vhd

UartDMA.v/vhd

UartRegBlock.v/vhd

UartTest.v/vhd

### Implications

No symptoms available to demonstrate this issue as problem was discovered through observation of the code.

### Workaround

None

## 333829: Uart Ignores the Disable Function

### Status

Affects: product PL011 UART -Perpetual.

Fault status: Cat 1, Present in: 1-v1, Fixed in 1-v3.

### Description

Whenever the FIFO is disabled during a transmit, the UART should continue to process the current data byte, and then pause. However, the Uart appears to ignore the disable function and continues transmitting under the following conditions:

- Under hardware control and multiple words in the transmit FIFO, the UART transmits the most recent character repeatedly if it was disabled before the character ended; thus the hardware flow control overrides the enable signal;
- When the UART was disabled before a character ended and there was a single word in the FIFO. The current character finished transmission as expected. However, when the UART was re-enabled, the last character was retransmitted.

### Affected Items

UartTxCntl.v

### Implications

Repetition of characters or the last character transmitted may be seen.

### Workaround

For the case where the hardware flow control overrides the enable signal. This can be fixed using the following code:

```
UartTxCntl.v Line 620
//
-----------------------------------------------------------------
// For Flow Control need to check whether it is still okay to send
// further data (CTSEn = '1' and nCTSSyncUARTCLK = '0'), provided
// that TXEnable is still true
// Otherwise normal operation resumes.
//
-----------------------------------------------------------------
if (CTSEn == 1'b1) begin
if ((nCTSSyncUARTCLK == 1'b0) && (TXEnable == 1'b1))
```

For the case when the UART is disabled during the transmission of a character, then re-enabled after that character completes; there is no suitable workaround without affecting the rest of the UART verification code.

## 333831: Uart CharTxComp Signal Is Not Synchronised

### Status

Affects: product PL011 UART -Perpetual.

Fault status: Cat 1, Present in: 1-v1, Fixed in 1-v3.

### Description

The CharTxComp signal from UartTXCtrl.v (uartclock domain) needs to be synchronised before being used in UartTXFIFO.v (pclk domain).

### Symptoms

Characters may be retransmitted.

### Affected Items

UartTXCtrl.v and UartTXFIFO.v

### Implications

There are issues with using multiple synchronised signals to control state machines in the TxFifo if the order of signals from UARTCLK are significant.

### Workaround

None

# Errata - Category 2

## 333821: Fractional Baud Rate Divisor Does Not Function With Some Divisor Values

### Status

Affects: product PL011 UART -Perpetual.

Fault status: Cat 2, Present in: 1-v0, Fixed in 1-v1.

### Description

### Summary

The original counter used the fractional part to select between the output of two separate 6-bit counters. These were /n, and /(n+1). If a value f is programmed into the UARTFBRD register then out of the 64 pulses generated, f comes from the (n+1) counter, and (64-f) comes from the n counter. All counters were clocked off UARTCLK. It was found that when f was an integer fraction of 64 (e.g.16) the switching could be synchronized with one of the counters and it was possible for pulses to be repeatedly lost. The resulting baud rate deviation errors could be up to about 25%.

The factor integer fraction of 64 is due to the fact that the 6-bit fractional part of the baud rate divisor (m), can be calculated by taking the fractional part of the required baud rate and multiplying it by and adding 0.5 to account for the rounding errors. It is this bit that more or less does not work as it should do.

### Symptoms

Examples of errors encountered:

Whenever the fractional baud rate divider register (FBRD) is set to a value other than 0, the resulting baud rate is wrong, where as when FBRD is 0 the baud rate matches exactly what is expected

Examples:

IBRD = 16, FBRD = 18: resulting division factor is approximately 21

IBRD = 37, FBRD = 10: resulting division factor is approximately 38

IBRD = 16, FBRD = 0: resulting division factor is 16 (correct)

IBRD = 1, FBRD = 0: resulting division factor is 1 (correct)

The example:

IBRD = 16, FBRD = 18

has been simulated and the results are as described above (i.e. division factor of approximately21).

However, the examples in the original test bench do seem to work (with approximately 2-3% error).

With a Uart clock of 80MHz and a required baud rate of 115,200:

UARTIBRD should be = 43 & UARTFBRD = 26.

The UART does not work correctly with these values.

However, with:

UARTIBRD = 43 & UARTFBRD = 27, the UART works correctly.

### Affected Items

Uart.v/vhd

UartApbif.v/vhd

UartBaudCntr.v/vhd

UartRegBlock.v/vhd

### Implications

The error occurs when the fractional divider function is utilized. The product works correctly if this function is not utilized. However, how to overcome and implement this is documented below.

### Workaround

The fractional baud rate divider register UARTFBRD mentioned on page 2-11 and 3-10 of the Technical Reference Manual (ARM DDI 0183A), used by the programmable baud generator should not be used and should be programmed to zero. Hence you should only use the integer register UARTIBRD.

Both these registers UARTFBRD and UARTIBRD are cleared on reset. Therefore it must be ensured that as the fractional baud rate generator is not being used, at no point during the working of the UART should there be a zero value in the integer UARTIBRD register. Otherwise no transmission or reception will take place (see note on page 3-10).

Some manual testing of different UART CLK / baud rate combinations has been carried out and the baud rate has been verified with 'scope as well as the reception of characters on our prospector board.

The table below shows the baud rate errors for a fixed input frequency.

| UART CLK (hz) prog | Uart CLK measured (AV) | | Baud | Pulse width Average (s) | Baud (Measured) | %error | | Character Transmission ok? |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| 7.3728.0e+6 | 7.37382.0e+6 | | 1200 | 1.56228E-04 | 1200.16898 | -0.01 | | OK (80%) |
| | 7.37382.0e+6 | | 2400 | 7.81139E-05 | 2400.34104 | -0.01 | | OK (80%) |
| | 7.37382.0e+6 | | 9600 | 1.95218E-05 | 9604.64711 | -0.05 | | OK (80%) |
| | 7.37382.0e+6 | | 14400 | 1.30187E-05 | 14402.35968 | -0.02 | | OK (80%) |
| | 7.37382.0e+6 | | 19200 | 9.76390E-06 | 19203.39209 | -0.02 | | OK (80%) |
| | 7.37382.0e+6 | | 38400 | 4.88168E-06 | 38408.90841 | -0.02 | | OK (100%) |
| | 7.37382.0e+6 | | 57600 | 3.25426E- | 57616.78538 | -0.03 | | OK (100%) |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | | | | 06 | | | | |
| | 7.37382.0e+6 | | 115200 | 1.62691E-06 | 115249.15330 | -0.04 | | OK (100%) |
| | 7.37382.0e+6 | | 230400 | 8.13230E-07 | 230562.07961 | -0.07 | | |
| | | | | | | | | |
| 5.00000E+06 | 5.00066E+06 | | 1200 | 1.56233E-04 | 1200.13057 | -0.01 | | OK (70%) |
| | | | 2400 | 7.81062E-05 | 2400.57768 | -0.02 | | OK (100%) |
| | | | 9600 | 1.94750E-05 | 9627.72786 | -0.29 | | OK (80%) |
| | | | 14400 | 1.30153E-05 | 14406.12203 | -0.04 | | OK (100%) |
| | | | 19200 | 9.74510E-06 | 19240.43878 | -0.21 | | OK (100%) |
| | | | 38400 | 4.86330E-06 | 38554.06823 | -0.40 | | OK |
| | | | 57600 | 3.25070E-06 | 57679.88433 | -0.14 | | |
| | | | 115200 | 1.62854E-06 | 115133.80083 | 0.06 | | OK (99%) |
| | | | 230400 | 4.06340E-07 | 461436.23566 | -100.28 | | total garbage |

In summary, to ensure that no further problems are encountered, only use the programmable 16-bit integer part, loaded through the UARTIBRD register, page 3-10. However, should the fractional baud rate register be required, then a spreadsheet has been produced that has a series of macro's built in, enabling the integer and fractional values to be calculated for a given input frequency and baud-rate requirement. The spreadsheet models the error component known to exist in the Rel1v0 release enabling values to be picked that have low/minimal error rates.

A spreadsheet for calculating the values that can be used in the fractional divider is available from ARM Support: email Support-primecell@arm.com.

A new release of the PL011, Rel1v1 was made available in which the Fractional Baud Rate divider was redesigned. The new implementation of the fractional baud rate divider only uses one 16-bit counter instead of the previous 2.

The legal values of the divisor have changed with the new implementation and logic has been introduced, such that the line control register (UARTLCR_H) and the baud rate divider registers (UARTIBRD and UARTFBRD) only get updated once the current character has completed.

## 333823: IrDA low power counter consumes power when IrDA mode is disabled

### Status

Affects: product PL011 UART -Perpetual.

Fault status: Cat 2, Present in: 1-v0, Fixed in 1-v1.

### Description

An eight bit counter within the IrDA section continues counting even when the IrDA mode is disabled. The counting action should be disabled whenever the IrDA mode is disabled.

### Affected Items

UartBaudCntr.v/vhd

UartIrDa.v/vhd

UartSynctoUCLK.v/vhd

UartTest.v/vhd

### Implications

No symptoms available to demonstrate this issue as problem was discovered through simulations.

### Workaround

None

## 333826: Some redundant logic in the Integration test logic

### Status

Affects: product PL011 UART -Perpetual.

Fault status: Cat 2, Present in: 1-v0, Fixed in 1-v1.

### Description

1) UARTITIP is defined as an 8-bit register but only bits 7 and 6 are used. It should be defined as UARTITIP [7:6].

2) The primary input signals which are read via the UARTITIP register should be taken from before the synchronisers, as they will only ever be read in a test mode.

### Symptoms

No symptoms available to demonstrate this issue.

### Affected Items

Uart_Integ.c

Uart.h

### Implications

Does not affect functionality of the device.

### Workaround

None

## 346440: FIFO overflow not correctly flagged

### Status

Affects: product PL011 UART -Perpetual.

Fault status: Cat 2, Present in: 1-0,1-v0,1-v1,1-v3, Fixed in r1p4-00rel0.

### Description

There is a condition where received data can be lost due to FIFO overflow, but no indication of the FIFO having overflowed is given.

If a data byte is received when the FIFO is full, and at the same instant a byte is read from the data register, the data byte is not written to the FIFO. There is, however, space in the FIFO to accept this byte, and a FIFO overflow is not signaled.

### Implications

A typical system using a UART will be designed to avoid FIFO overflow, so it is expected that this defect would rarely be observed. In a system which uses the UART to transfer many bytes of data back-to-back (bulk transfer) then the system response time to FIFO data should ensure that the FIFO will never fill completely.

Systems which use the UART to transfer small quantities of data (packetised transfer) may rely on the average data rate being much lower than the peak data rate. These systems are more prone to occasional FIFO overflow, and would rely on the FIFO overflow flag to signal that a packet had been corrupted. The impact of this defect could be serious in such systems.

### Workaround

None

## 459631: UART Break error issue

### Status

Affects: product PL011 UART -Perpetual.

Fault status: Cat 2, Present in: 1-0,1-v0,1-v1,1-v3,r1p4-00rel0, Fixed in r1p5-00rel0.

### Description

The PL011 UART incorrectly interprets BREAK condition on the receive IrDA.

Due to the IrDA receive block inserting regular rising edges at the end of each bit, a single long break is detected as a sequence of break characters which will all be placed in the fifo.

Affected Items

UartIrDA.v

### Implications

Improper decoding of IrDA receiver results in

1. UART Receive controller comes out of BREAK state even while receiving continuous break characters.

2. RXFIFO gets filled with '0' data when the break condition is still true.

### Workaround

none

# Errata - Category 3

## 333824: Two Parameters Instead of Three Are Being Passed To Clockgen

### Status

Affects: product PL011 UART -Perpetual.

Fault status: Cat 3, Present in: 1-v0,1-v1,1-v3,r1p4-00rel0, Fixed in r1p5-00rel0.

### Description

The module clockgen.v requires three parameters to be passed to it, but in it's instantiation, only two are being passed, resulting in one value being held constant.

### Symptoms

There will be a parameter mismatch between CLOCKGEN module call in apbslave_tb.v and the CLOCKGEN module definition in clockgen.v.

### Affected Items

apbslave_tb.v file

### Implications

Mismatch does not affect functionality of the device.

### Workaround

Change, within the apbslave_tb.v file:

```
CLOCKGEN #(`Tclkh, `Tclkl) U_CLOCKGEN
(.PCLK(I_CLK));
```

to:

```
CLOCKGEN #(`Tclks, `Tclkh, `Tclkl) U_CLOCKGEN
(.PCLK(I_CLK));
```

This will resolve the mismatch.

# Errata - Documentation

## 326409: Incorrect Information in TRM for register UARTILPR

### Status

Affects: product PL011 UART -Perpetual.

Fault status: Doc, Present in: 1-v3, Fixed in r1p4-00rel0.

### Description

There is a mismatch between the Technical Reference Manual(TRM) and the PL011 Design.

Section 3.3 Register description:

3.3.4 IrDA low-power counter register, UARTILPR, states:

**The minimum frequency of IrLPBaud16 ensures that pulses less than one period of IrLPBaud16 are rejected, but that pulses greater than 1.4µs are accepted as valid pulses.**

However, in in UartIrDA.v line 435- where SIRINStag2Final signal is generated, one cycle pulse of UARTCLK is rejected as noise but more than two pulses are accepted.

Ordinarily IrLPBaud16 is divided from UARTCLK so the noise margin in RTL is smaller than is explained in the TRM.

3.3.4 IrDA low-power counter register, UARTILPR should say:

**The minimum frequency of IrLPBaud16 ensures that pulses less than one period of UARTCLK are rejected as random noise, but that pulses greater than two periods of UARTCLK are accepted as valid pulse.**

### Implications

This is a documentation error and so should not affect the functionality of the device

### Workaround

none

## 330540: TRM Error with UARTPeriphID2 revision

### Status

Affects: product PL011 UART -Perpetual.

Fault status: Doc, Present in: 1-v3, Fixed in r1p4-00rel0.

### Description

In the TRM, ARM DDI 0183E, there is the following error:

p 3-25, UARTPeriphID2 register:

Bits 7:4 Revision These bits read back as 0x2, not 0x1 as in the text.

The release note PL011 RELN 0000 D also gives incorrect versions for some documentation:

The TRM is ARM DDI 0183E, not ARM DDI 0183D as in the release note

The Design Manual is PL011 DDES 0000 D, not PL011 DDES 0000 C as in the release note

### Implications

This register is used by software to determine which version of the UART is being used.

### Workaround

None

## 333827: Design Manual Documentation Error

### Status

Affects: product PL011 UART -Perpetual.

Fault status: Doc, Present in: 1-v0,1-v1, Fixed in 1-v3.

### Description

An equation did not match the state diagram in Figure 26 of the Design Manual.

### Symptoms

The state diagrams in the design manual do not match the equations implemented in the code.

The equation in the design manual is:

```
((CTSEn + CTSSyncUARTCLK') * CTSEn) + TXDataAvlblSync*Baud16*AbortTransmit'
```

The equation implemented in the code is:

```
((CTSEn * CTSSyncUARTCLK') + CTSEn') *
TXDataAvlblSync*Baud16*TXEnable*AbortTransmit'
```

### Affected Items

PL011 DDES 0000A.pdf Page 47

### Implications

These errors are issued as documentation errata and do not affect the operation or verification of the device.

### Workaround

None

## 406210: DDI0183F UART (PL011) r1p4 TRM - UARTPeriphID2 revision bits are incorrect

### Status

Affects: product PL011 UART -Perpetual.

Fault status: Doc, Present in: r1p4-00rel0, Fixed in r1p5-00rel0.

### Description

In the TRM, ARM DDI 0183F, there is the following error:

p 3-24, UARTPeriphID2 register:

Bits 7:4 Revision These bits read back as 0x2, not 0x3 as in the text.

### Implications

This register is used by software to determine which version of the UART is being used.

### Workaround

In TRM, ARM DDI 0183G, p 3-24,

UARTPeriphID2 Revision register Bits 7:4 is updated and will read

0x2 for r1p4-00rel0 and

0x3 for r1p5-00rel0

# Errata – Software

## 179060: Driver code reads a write only register

### Status

Affects: product PL011 UART -Perpetual.

Fault status: Soft, Present in: 1-v3,r1p4-00rel0,r1p5-00rel0, Open.

### Description

In the file uart.c the macro UART_INTCLEAR reads a write only register and the interrupts it clears are dependant on that read.

### Implications

It is possible that the driver will incorrectly clear interrupts.

For example, if a modem interrupt occurs while a receive interrupt is being processed, the routine servicing the receive interrupt could clear the modem interrupt, which would then not be processed.

This will only occur on a system where the interrupt clear register reads as non-zero.

### Workaround

In the file uart.c the macro UART_INTCLEAR should be redefined as

```
#define UART_INTCLEAR(__type) apBIT_SET_NOREAD(pBase->IntClear, UART_ ## __type, 0xFFF);
```