

# ERRATA SHEET

**Date:** 2007 July 27  
**Document Release:** Version 1.0  
**Device Affected:** LPC2194/01

This errata sheet describes both the functional deviations and any deviations from the electrical specifications known at the release date of this document.

Each deviation is assigned a number and its history is tracked in a table at the end of the document.

2007 July 27

**Document revision history**

Rev	Date	Description
1.0	July 27,2007	First version

## Identification

The typical LPC2194/01 devices have the following top-side marking:

LPC2194xxxxxx

/01

xxxxxx

xxYYWW R

The last letter in the last line (field 'R') will identify the device revision. This Errata Sheet covers the following revisions of the LPC2194/01:

Revision Identifier (R)	Comment
'C'	First device revision

Field 'YY' states the year the device was manufactured. Field 'WW' states the week the device was manufactured during that year.

**Errata Overview - Functional Problems**

Functional Problem	Short Description	Device Revision
Core.1	Incorrect update of the Abort Link register in Thumb state	C
CAN.1	Data overrun condition can lock the CAN controller	C

**Errata Overview - AC/DC Deviations**

AC/DC Deviation	Short Description	Device Revision
NA	NA	NA

**Errata Notes**

Note	Short Description
Note.1	New CAN controller with some changes

## Functional Deviations of LPC2194/01

### Core.1 Incorrect update of the Abort Link register in Thumb state

**Introduction:** If the processor is in Thumb state and executing the code sequence STR, STMIA or PUSH followed by a PC relative load, and the STR, STMIA or PUSH is aborted, the PC is saved to the abort link register.

**Problem:** In this situation the PC is saved to the abort link register in word resolution, instead of half-word resolution.

**Conditions:**

The processor must be in Thumb state, and the following sequence must occur:

<any instruction>

<STR, STMIA, PUSH> <---- data abort on this instruction

LDR rn, [pc,#offset]

In this case the PC is saved to the link register R14\_abt in only word resolution, not half-word resolution. The effect is that the link register holds an address that could be #2 less than it should be, so any abort handler could return to one instruction earlier than intended.

**Work around:** In a system that does not use Thumb state, there will be no problem.

In a system that uses Thumb state but does not use data aborts, or does not try to use data aborts in a recoverable manner, there will be no problem.

Otherwise the workaround is to ensure that a STR, STMIA or PUSH cannot precede a PC-relative load. One method for this is to add a NOP before any PC-relative load instruction. However this is would have to be done manually.

### CAN.1: Data Overrun condition can lock the CAN controller

**Introduction:** Each CAN controller provides a double Receive Buffer (RBX) per CAN channel to store incoming messages until they are processed by the CPU. Software task should read and save received data as soon as a message reception is signaled.

In cases, where both receive buffers are filled and the contents are not read before the third message comes in, a CAN Data Overrun situation is signaled. This condition is signaled via the Status register and the Data Overrun Interrupt (if enabled).

**Problem:** In a Data Overrun condition, the CAN controller is locked from further message reception.

**Workaround:**

1. Recovering from this situation is only possible with a soft reset to the CAN controller.
2. If software cannot read all messages in time before a third message comes in, it is recommend to change the acceptance filtering by adding further acceptance filter group(s) for messages, which are normally rejected. With this approach, the third incoming message is accepted and the Data Overrun condition is avoided. These additional messages are received with the corresponding group index number can be easily identified and rejected by software.

## Errata Notes

**Note.1:** The CAN controller on the LPC2194/01 is a new certified CAN controller. This controller has an improved interrupt behavior in Full-CAN mode. Care should be taken while using the global CAN filter look-up table (LUT). In the old CAN block (as found in the LPC2194 and LPC2194/00), the numbering of the CAN interfaces in the LUT is 1 to n.- In the new CAN block, the numbering of the CAN interfaces in the LUT is 0 to n-1 (n being the number of implemented CAN interfaces (2 or 4)).