

CC2520

ERRATA NOTES

TABLE OF CONTENTS

1	KNOWN BUGS	2
1.1	BUG 1: COMMAND STROBES CAN CORRUPT RX FIFO CONTENTS	2
1.2	BUG 2: GLITCHES ON THE FIFOP SIGNAL	3
1.3	BUG 3: SFLUSHRX CAUSES RX RECALIBRATION	4
1.4	BUG 4: GLITCHES ON THE RX_ACTIVE AND TX_ACTIVE SIGNALS	4
1.5	BUG 5: REGWR DOES NOT OUTPUT REPLACED VALUE	4
2	DOCUMENT HISTORY	5

1 Known bugs

Table 1: Summary of known bugs

Bug ID	Title
1	Command strobes can corrupt RX FIFO contents
2	Glitches on the FIFOP signal
3	SFLUSHRX causes RX recalibration
4	Glitches on the RX_ACTIVE and TX_ACTIVE signals
5	REGWR does not output replaced value

1.1 Bug 1: Command Strobes Can Corrupt RX FIFO Contents

This bug may be triggered by the SRXON, SRFOFF and STXON command strobes when frame filtering is enabled. It may also be triggered by the SFLUSHRX command strobe.

1.1.1 Bug Description

The bug is triggered when the command strobe is executed at the same time as a byte is received and written to the RX FIFO. The result is that the last received byte is always written to the RX FIFO. There are two situations where this extra byte is undesired and will cause problems:

- If the SFLUSHRX strobe is executed during reception of a frame, one would expect the FIFO to be empty afterwards. However, because of the bug, there could still be one byte in the FIFO, which, if not dealt with, would be interpreted as the length byte of the next received frame.
- If the SRXON, SRFOFF or STXON strobes are issued during reception of a frame, after the frame has been rejected, one extra byte could appear in the RX FIFO with no warning (before rejection or for accepted frames, there will be an RX_FRM_ABORTED exception). The consequences are the same as for SFLUSHRX.

1.1.2 Suggested Workaround

The proposed workaround consists of several small fixes that are simple to implement and have few negative side-effects.

Make sure to disable interrupts while the workaround is running if there is any risk of:

- Long interrupts (that may allow the bug to occur again before the workaround is completed)
- Interrupts that may interfere with the workarounds

SFLUSHRX

The workaround consists of flushing the RX FIFO twice instead of once. The second flush operation removes the unwanted byte from the RX FIFO:

```
DISABLE_INTERRUPTS();  
SFLUSHRX();  
SFLUSHRX();  
ENABLED_INTERRUPTS();
```

SRXON, SRFOFF and STXON

The workaround consists of flushing the RX FIFO after the strobe has been executed. The following procedure reduces the risk of false bug detection:

```
DISABLE_INTERRUPTS();

// Record the number of bytes in the RX FIFO before the strobe
rxfifoCntBefore = CC2520_REGRD(RXFIFOCNT);
SRFOFF(); // or SRXON or STXON

// If RXFIFOCNT changed during the last operation, it is very
// likely that the bug has occurred
if (RXFIFOCNT != rxfifoCntBefore) {
    SFLUSHRX();
    // Software clean-up
    ... Reset software for frame reception ...
}

ENABLED_INTERRUPTS();
```

Note: When flushing the RX FIFO, received frames still to be read out will be lost. There is a chance that the lost frames already have been acknowledged. The effect is that the transmitter gets a false acknowledgment in return, but this is likely to happen sooner or later anyway (the only thing that separates one acknowledgment frame from another is the 8-bit sequence number).

1.2 Bug 2: Glitches on the FIFOP Signal

This bug is triggered when receiving frames with length 0 or 1 that are rejected by frame filtering.

1.2.1 Bug Description

If frame filtering is enabled and a frame with length byte 0x00/0x80 or 0x01/0x81 is received, a short pulse (1-2 32 MHz clock cycles) will be generated on the FIFOP signal output. Also, due to the positive edge on the FIFOP signal, a FIFOP exception will be generated.

1.2.2 Suggested Workaround

Make sure to disable interrupts while the workaround is running if there is any risk of:

- Long interrupts (that may allow the bug to occur again before the workaround is completed)
- Interrupts that may interfere with the workarounds

When reading the FIFOP signal status (from the GPIO output or the FSMSTAT1 register), always read twice, and AND the read values, thereby masking out the glitch.

```
DISABLE_INTERRUPTS();
status = GET_FIFOP_STATUS();
status = status & GET_FIFOP_STATUS();
ENABLED_INTERRUPTS();
```

If the FIFOP interrupt is implemented, apply the following check at the beginning of the interrupt service routine:

```
// Disable interrupts unless this is done automatically by the MCU
DISABLE_INTERRUPTS();
if (!GET_FIFOP_STATUS()) return;
if (!GET_FIFOP_STATUS()) return;
ENABLED_INTERRUPTS();
```

Apply a similar check when using the FIFOP exception

1.3 Bug 3: SFLUSHRX Causes RX Recalibration

1.3.1 Bug Description

The SFLUSHRX command strobe causes receiver recalibration when issued while the receiver is active. This will prevent new frames from being received during the short period after the flush operation. It will also reset RSSI recording, and thereby prevent CCA checking for 320 us (the result will always be false).

1.3.2 Suggested Workaround

There is no fix for this bug, other than minimizing usage of the SFLUSHRX strobe.

1.4 Bug 4: Glitches on the RX_ACTIVE and TX_ACTIVE Signals

1.4.1 Bug Description

Very short glitches (~ ns) can occur on the RX_ACTIVE and TX_ACTIVE signals when output on one of the GPIO pins. This may prevent the signals from being read reliably by a microcontroller.

1.4.2 Suggested Workaround

Fetch the values from the SPI status byte, or by reading the FSMSTAT1 register.

It is still possible to use the signals on GPIO outputs for manual observation on an oscilloscope, logic analyzer or LEDs indicating RX or TX.

1.5 Bug 5: REGWR Does Not Output Replaced Value

1.5.1 Bug Description

The REGWR instruction does not return the data it is about to replace. It always returns 0x00.

1.5.2 Suggested Workaround

Use the MEMWR instruction instead, or do a REGRD before REGWR.

2 Document history

Revision	Date	Description/changes
SWRZ024	2007-12-20	Initial release

IMPORTANT NOTICE

Texas Instruments Incorporated and its subsidiaries (TI) reserve the right to make corrections, modifications, enhancements, improvements, and other changes to its products and services at any time and to discontinue any product or service without notice. Customers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All products are sold subject to TI's terms and conditions of sale supplied at the time of order acknowledgment.

TI warrants performance of its hardware products to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are used to the extent TI deems necessary to support this warranty. Except where mandated by government requirements, testing of all parameters of each product is not necessarily performed.

TI assumes no liability for applications assistance or customer product design. Customers are responsible for their products and applications using TI components. To minimize the risks associated with customer products and applications, customers should provide adequate design and operating safeguards.

TI does not warrant or represent that any license, either express or implied, is granted under any TI patent right, copyright, mask work right, or other TI intellectual property right relating to any combination, machine, or process in which TI products or services are used. Information published by TI regarding third-party products or services does not constitute a license from TI to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property of the third party, or a license from TI under the patents or other intellectual property of TI.

Reproduction of information in TI data books or datasheets is permissible only if reproduction is without alteration and is accompanied by all associated warranties, conditions, limitations, and notices. Reproduction of this information with alteration is an unfair and deceptive business practice. TI is not responsible or liable for such altered documentation.

Resale of TI products or services with statements different from or beyond the parameters stated by TI for that product or service voids all express and any implied warranties for the associated TI product or service and is an unfair and deceptive business practice. TI is not responsible or liable for any such statements.

Following are URLs where you can obtain information on other Texas Instruments products and application solutions:

Products		Applications	
Amplifiers	amplifier.ti.com	Audio	www.ti.com/audio
Data Converters	dataconverter.ti.com	Automotive	www.ti.com/automotive
DSP	dsp.ti.com	Broadband	www.ti.com/broadband
Interface	interface.ti.com	Digital Control	www.ti.com/digitalcontrol
Logic	logic.ti.com	Military	www.ti.com/military
Power Mgmt	power.ti.com	Optical Networking	www.ti.com/opticalnetwork
Microcontrollers	microcontroller.ti.com	Security	www.ti.com/security
		Telephony	www.ti.com/telephony
		Video & Imaging	www.ti.com/video
		Wireless	www.ti.com/wireless

Mailing Address: Texas Instruments Post Office Box 655303 Dallas, Texas 75265

Copyright © 2007, Texas Instruments Incorporated