ERRATA NOTE

CC1110Fx/CC1111Fx

Table Of Contents

1 Part May Hang in Power Mode

The following applies only to power mode 2 and 3.

1.1 Description of the Problem

When waking up from power mode 2 and 3 there is a small chance that the `SLEEP.MODE` bits are faulty set to a value other than zero before the `PCON.IDLE` bit is cleared by the CPU. This causes the chip to re-enter power mode immediately. Since an enabled interrupt is pending at this point, the chip will wake up and re-enter power mode continuously and appear to hang.

Once the device hang, only a system reset will get the chip back to normal operation.

1.2 Suggested Work-around

By ensuring that the `SLEEP.MODE` bits are written to zero at the instant the chip wakes from power mode, the chip will never re-enter power mode unintentionally.

If the following conditions are met, this can be done by setting up a DMA transfer to the `SLEEP` register that is triggered right before writing the `PCON.IDLE` bit.

- The chip is running at the HS RC oscillator at the highest possible clock speed setting
- The high speed crystal oscillator is powered down
- Flash Cache is disabled

Please note that the requirements stated in the following chapters of the data sheet still applies: *Power Management Control* and *SleepTimer and Power Modes*.

Note: The following code assumes the chip is already running at the HS RC oscillator with the highest clock speed setting possible and this has to be handled by the application. The code marked blue below is timing critical and should be done in the order as shown here with no intervening code.

```
// Initialization of source buffers and DMA descriptor for the DMA transfer
unsigned char __xdata PM2_BUF[7] = {0x06,0x06,0x06,0x06,0x06,0x06,0x04};
unsigned char __xdata PM3_BUF[7] = {0x07,0x07,0x07,0x07,0x07,0x07,0x04};
unsigned char __xdata dmaDesc[8] = {0x00,0x00,0xDF,0xBE,0x00,0x07,0x20,0x42};

// Store current DMA channel 0 descriptor and abort any ongoing transfers if //
the channel is in use
unsigned char storedDescHigh = DMA0CFGH;
unsigned char storedDescLow = DMA0CFGL;
DMAARM |= 0x81;

// Update descriptor with correct source
// NB! Replace &PM2_BUF with &PM3_BUF if powermode 3 is chosen instead
dmaDesc[0] = (unsigned int)& PM2_BUF >> 8;
dmaDesc[1] = (unsigned int)& PM2_BUF;

// Associate the descriptor with DMA channel 0 and arm the DMA channel
DMA0CFGH = (unsigned int)&dmaDesc >> 8;
DMA0CFGL = (unsigned int)&dmaDesc;
DMAARM = 0x01;

// NOTE! At this point, make sure all interrupts that will not be used to
// wake from PM are disabled as described in chapter 13.1.3 of the datasheet.

// Align with positive 32 kHz clock edge as described in chapter 13.8.2
// of the datasheet.
char temp = WORTIME0;
while( temp == WORTIME0);

// Make sure XOSC is powered down when entering PM2/3 and that the flash
// cache is disabled
// NB! Replace 0x06 with 0x07 if power mode 3 is chosen instead
MEMCTR |= 0x02;
SLEEP = 0x06;

// Enter power mode as described in chapter 13.1.3 in the datasheet.
// Make sure DMA channel 0 is triggered just before setting PCON.IDLE
asm("NOP");
asm("NOP");
asm("NOP");
if( SLEEP & 0x03 ) {
    asm("MOV 0xD7,#0x01");     // DMAREQ = 0x01;
    asm("NOP");                // Needed to perfectly align the DMA transfer
    asm("ORL 0x87,#0x01");     // PCON |= 0x01;
    asm("NOP");
}

// Enable Flash Cache
MEMCTR &= ~0x02;

// Update DMA channel 0 with original descriptor and arm channel if it was in //
use before PM was entered
DMA0CFGH = storedDescHigh;
DMA0CFGL = storedDescLow;
DMAARM = 0x01;
```

2 RX_OVERFLOW Issue

2.1 Description of the Problem

In addition to the RFD register, the ***CC1110Fx/CC1111Fx*** has several internal buffers for status registers, CRC bytes, and buffers used when FEC is enabled. If there is a byte in the RFD register and more bytes are written to this register by the radio, the radio will enter RX_OVERFLOW state. There are however some cases where the radio will be stuck in RX state instead of entering RX_OVERFLOW state, as it should. Below is a table showing the register settings that will cause this problem. APPEND_STATUS is found in the PKTCTRL1 register, CRC_EN is found in the PKTCTRL0 register, and FEC_EN is in the MDMCFG1 register. In the table below, x is the number of bytes that will be written to the RFD register by the radio (including the status bytes if APPEND_STATUS = 1). Assume that the radio is configured to enter IDLE state after a packet has been received.

When the radio is stuck in RX state like this, it will draw current as in RX state, but it will not be able to receive any more data. Neither RFIF.IRQ_DONE nor RFIF.IRQ_RX_OVF will be asserted. The only way to proceed is by issuing an SIDLE strobe command (RFST = 0x04).

Register Settings	# of Bytes Read from the RFD Register	MARCSTATE	Comment
APPEND_STATUS = 1 CRC_EN = 0 FEC_EN = 0	x – 1	RX_OVERFLOW	Ok
	x	IDLE	Ok
APPEND_STATUS = 1 CRC_EN = 0 FEC_EN = 1	x – 6	RX_OVERFLOW	Ok
	x – 5	RX	Not ok. Stuck in RX
	x – 4	RX	Not ok. Stuck in RX
	x – 3	RX	Not ok. Stuck in RX
	x – 2	RX	Not ok. Stuck in RX
	x – 1	RX	Not ok. Stuck in RX
	x	IDLE	Ok
APPEND_STATUS = 1 CRC_EN = 1 FEC_EN = 0	x – 3	RX_OVERFLOW	Ok
	x – 2	RX	Not ok. Stuck in RX
	x – 1	RX	Not ok. Stuck in RX
	x	IDLE	Ok
APPEND_STATUS = 1 CRC_EN = 1 FEC_EN = 1	x – 5	RX_OVERFLOW	Ok
	x – 4	RX	Not ok. Stuck in RX
	x – 3	RX	Not ok. Stuck in RX
	x – 2	RX	Not ok. Stuck in RX
	x – 1	RX	Not ok. Stuck in RX
	x	IDLE	Ok
APPEND_STATUS = 0 CRC_EN = 0 FEC_EN = 0	x – 1	RX_OVERFLOW	Ok
	x	IDLE	Ok

APPEND_STATUS = 0 CRC_EN = 0 FEC_EN = 1	x – 4	RX_OVERFLOW	Ok
	x – 3	RX	Not ok. Stuck in RX
	x – 2	RX	Not ok. Stuck in RX
	x – 1	RX	Not ok. Stuck in RX
	x	IDLE	Ok
APPEND_STATUS = 0 CRC_EN = 1 FEC_EN = 0	x – 1	RX_OVERFLOW	Ok
	x	IDLE	Ok
APPEND_STATUS = 0 CRC_EN = 1 FEC_EN = 1	x – 3	RX_OVERFLOW	Ok
	x – 2	RX	Not ok. Stuck in RX
	x – 1	RX	Not ok. Stuck in RX
	x	IDLE	Ok

2.2 Suggested Work-around

In applications where the DMA is used to read the RFD register, it is important to configure the DMA in accordance with the chosen radio configuration. Please see DN107 for more details on how this should be done. If the RFD register is read manually, it is important that the register is read when the RFTXRXIF flag in the TCON register has been asserted. If the RFTXRX interrupt is used, it is important that this interrupt has a high priority. If a polling scheme is used, one needs to make sure that interrupts that are enabled will not prevent the RFD register to be read before a new byte is received.

3 Premature XOSC_STB Assertion

The XOSC_STB signal (crystal oscillator stable signal) is derived from an internal ripple counter, which counts clock pulses during oscillator start up. Due to the very small oscillation amplitude during the initial crystal start up, the ripple counter can in some cases trigger on power noise instead of the actual crystal swing, and there is a finite possibility that the ripple counter used to set XOSC_STB can assert XOSC_STB prematurely.

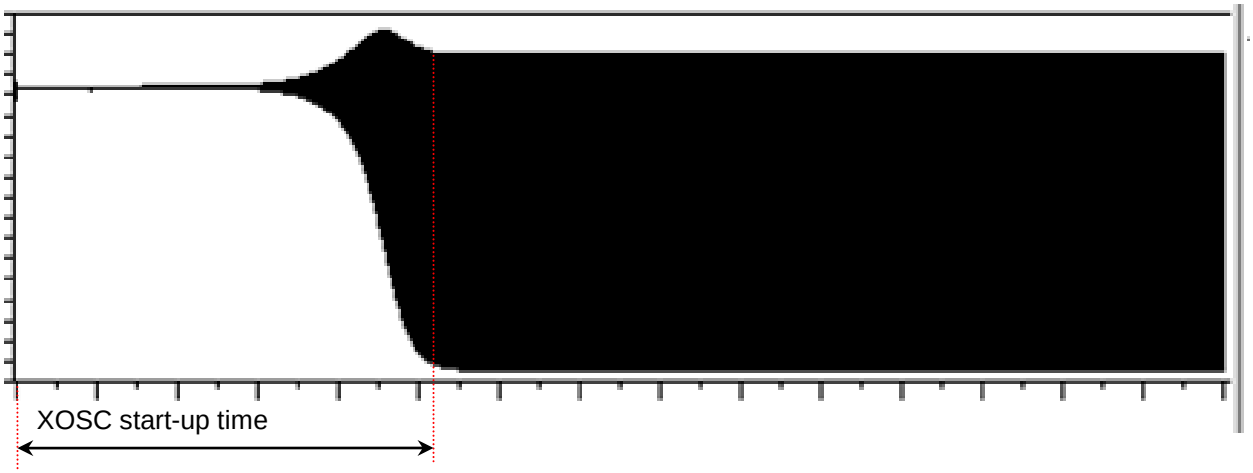
If the HS XOSC is turned on by selecting it as source for the system clock (CLKCON.OSC = 0), CLKCON.OSC will remain 1 (and keep the HS RCOSC as source) until the SLEEP.XOSC_STB bit is asserted. In the cases where XOSC_STB is asserted prematurely the system clock will change clock source prematurely and end up running on an unstable clock causing unpredictable behavior.

3.1 Suggested Work-around

The HS XOSC must be turned on by setting SLEEP.OSC_PD = 0. A SW delay should then be implemented to make sure the crystal is stable before selecting the HS XOSC as system clock (CLKCON.OSC = 0). The SW delay has to be greater than the maximum start-up time of the crystal.

```
SLEEP &= ~0x04;    // Power up both oscillators (HS XOSC and HS RCOSC)
delay();           // Delay longer than max crystal start-up time
CLKCON &= ~0x40;   // Set high speed crystal oscillator as system clock
```

The figure below shows the XOSC_Q1 pin as the crystal is turned on. When measuring on XOSC_Q1 one needs to use a high impedance probe to avoid loading the crystal.



When finding the max start-up time one should measure on several crystals over temperature. A 100% margin should be added to the measured start-up time to account for process variations.

4 Document History

Revision	Date	Description/Changes
SWRZ022C	2012-11-22	Updated with issue related to the XOSC_STB signal.
SWRZ022B	2007-12-21	Updated with issue related to RX_OVERFLOW state. Removed “Batches Affected” since there is only one revision available of this product.
SWRZ022A	2007-09-18	Updated with clarified conditions for the fix and update the code to remove instability.
SWRZ022	2007-09-06	Released for RTM

TEXAS INSTRUMENTS NORWAY

Tel. +47-22958544

Fax +47-22958546

www.ti.com

IMPORTANT NOTICE

Texas Instruments Incorporated and its subsidiaries (TI) reserve the right to make corrections, enhancements, improvements and other changes to its semiconductor products and services per JESD46, latest issue, and to discontinue any product or service per JESD48, latest issue. Buyers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All semiconductor products (also referred to herein as “components”) are sold subject to TI’s terms and conditions of sale supplied at the time of order acknowledgment.

TI warrants performance of its components to the specifications applicable at the time of sale, in accordance with the warranty in TI’s terms and conditions of sale of semiconductor products. Testing and other quality control techniques are used to the extent TI deems necessary to support this warranty. Except where mandated by applicable law, testing of all parameters of each component is not necessarily performed.

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

TI does not warrant or represent that any license, either express or implied, is granted under any patent right, copyright, mask work right, or other intellectual property right relating to any combination, machine, or process in which TI components or services are used. Information published by TI regarding third-party products or services does not constitute a license 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 significant portions of TI information in TI data books or data sheets is permissible only if reproduction is without alteration and is accompanied by all associated warranties, conditions, limitations, and notices. TI is not responsible or liable for such altered documentation. Information of third parties may be subject to additional restrictions.

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

Buyer acknowledges and agrees that it is solely responsible for compliance with all legal, regulatory and safety-related requirements concerning its products, and any use of TI components in its applications, notwithstanding any applications-related information or support that may be provided by TI. Buyer represents and agrees that it has all the necessary expertise to create and implement safeguards which anticipate dangerous consequences of failures, monitor failures and their consequences, lessen the likelihood of failures that might cause harm and take appropriate remedial actions. Buyer will fully indemnify TI and its representatives against any damages arising out of the use of any TI components in safety-critical applications.

In some cases, TI components may be promoted specifically to facilitate safety-related applications. With such components, TI’s goal is to help enable customers to design and create their own end-product solutions that meet applicable functional safety standards and requirements. Nonetheless, such components are subject to these terms.

No TI components are authorized for use in FDA Class III (or similar life-critical medical equipment) unless authorized officers of the parties have executed a special agreement specifically governing such use.

Only those TI components which TI has specifically designated as military grade or “enhanced plastic” are designed and intended for use in military/aerospace applications or environments. Buyer acknowledges and agrees that any military or aerospace use of TI components which have ***not*** been so designated is solely at the Buyer's risk, and that Buyer is solely responsible for compliance with all legal and regulatory requirements in connection with such use.

TI has specifically designated certain components as meeting ISO/TS16949 requirements, mainly for automotive use. In any case of use of non-designated products, TI will not be responsible for any failure to meet ISO/TS16949.

Products		**Applications**	
Audio	www.ti.com/audio	Automotive and Transportation	www.ti.com/automotive
Amplifiers	amplifier.ti.com	Communications and Telecom	www.ti.com/communications
Data Converters	dataconverter.ti.com	Computers and Peripherals	www.ti.com/computers
DLP® Products	www.dlp.com	Consumer Electronics	www.ti.com/consumer-apps
DSP	dsp.ti.com	Energy and Lighting	www.ti.com/energy
Clocks and Timers	www.ti.com/clocks	Industrial	www.ti.com/industrial
Interface	interface.ti.com	Medical	www.ti.com/medical
Logic	logic.ti.com	Security	www.ti.com/security
Power Mgmt	power.ti.com	Space, Avionics and Defense	www.ti.com/space-avionics-defense
Microcontrollers	microcontroller.ti.com	Video and Imaging	www.ti.com/video
RFID	www.ti-rfid.com		
OMAP Applications Processors	www.ti.com/omap	**TI E2E Community**	e2e.ti.com
Wireless Connectivity	www.ti.com/wirelessconnectivity		

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