19-0647; Rev 1; 3/07

ERRATA SHEET

MAX3421E
USB Peripheral/Host Controller with SPI Interface

www.maxim-ic.com

REVISION 1 ERRATA

The errata listed below describe situations where the MAX3421E components perform differently than expected or differently than described in the data sheet. Maxim Integrated Products, Inc. intends to address these errata in subsequent die revisions.

Note: The version of the MAX3421E can be determined electrically (by firmware) by reading the Die Revision register (R18); e.g., 0x01 is version 01, 0x12 is version 02. The die revision 2 may also be discovered by reading the top mark as described below. If no die revision digit is present, then the part is version 01.

CHIP REVISION IDENTIFICATION

Top Mark:

MAX3421EEHJ+ (32-pin TQFP package):

MAX3421EE

yyww2

^

yww is the date code

where y = last digit of year

ww = work week (01 to 52)

2 denotes die revision 2

MAX3421EETJ+ (32-pin TQFN package):

3421EE

TJyww < date code

XXXX2

yww is the date code

where y = last digit of year

ww = work week (01 to 52)

2 denotes die revision 2

ERRATA ITEMS	DIE REVISIONS AFFECTED	FIX STATUS
HIGH I_{LSUS}	0x01	Fixed in rev 0x02
Double buffering does not work properly on EP1-OUT IN USB peripheral mode	0x01	Fixed in rev 0x02
Random corruption of the first byte in the SNDFIFO	(0x01)(0x12)	Will be fixed in rev 3 (0x13)

1. HIGH I_{LSUS}

Description:

When operating as a peripheral, the USB-specified suspend current cannot be guaranteed over all operating conditions. The MAX3421E draws excess current when in suspend mode; this impacts bus-powered designs (peripherals that draw power from V_{BUS}) from meeting USB suspend current requirements.

Work Around:

High I_{LSUS} does not impact self-powered peripheral designs. There is no work around for bus-powered designs. Bus-powered peripherals designed using the MAX3421E will function normally but will not pass USB-compliance testing for suspend current.

Status:

Fixed in Die Revision 0x02.

2. DOUBLE BUFFERING DOES NOT WORK PROPERLY ON EP1-OUT, WHEN IN USB PERIPHERAL MODE.

Description:

If both FIFO buffers of endpoint 1-OUT contain USB packets, and the SPI master clears the OUT1DAVIRQ flag while a third packet is transmitted by the host over the bus, the FIFO data is corrupted. This situation is most often encountered when bulk-out data transfers of > 64 bytes are attempted.

Work Around:

Some USB applications (e.g., HID) do not require an OUT endpoint and are unaffected by this problem. There is no work around for applications that use EP1-OUT.

Status:

Will be fixed in Die Revision 0x02.

3. RANDOM CORRUPTION OF THE FIRST BYTE IN THE SNDFIFO.

Description:

When the MAX3421E sends an OUT packet that is NAK'd by a USB peripheral, the MAX3421E resends the packet by reloading the HXFR register. In revisions 1 and 2, the first byte in the SNDFIFO can be corrupted when this is done. The corruption occurs on a random basis.

Work Around:

The firmware that controls the SPI master resends a NAK'd OUT packet using the following four steps:

1. Write the SNDBC register with any value to switch the SNDFIFO back to the microcontroller.
2. Rewrite the first SNDFIFO byte (ONLY) with the previously written first byte.
3. Rewrite the SNDBC register with the previously written byte count. This switches the SNDFIFO back to USB.
4. Launch another OUT transfer by writing the HXFR register with the same value used to launch the original transfer.

Status:

Will be fixed in Revision 3. When fixed, only step 4 will be necessary to relaunch a NAK'd OUT packet.

App Note:

AN4000 (http://www.maxim-ic.com/appnotes.cfm/appnote_number/4000) describes the issue in detail and contains an example function to implement a firmware workaround.