

**TMS320F2810, TMS320F2811, TMS320F2812**  
**TMS320C2810, TMS320C2811, TMS320C2812**  
**DSP Silicon Errata**

*SPRZ193L*  
January 2003 – Revised March 2009



Copyright © 2009, Texas Instruments Incorporated

## Contents

<b>1</b>	<b>Introduction</b>	<b>4</b>
1.1	Device and Development Tool Support Nomenclature	4
1.2	Device Markings	5
<b>2</b>	<b>Known Design Marginality/Exceptions to Functional Specifications</b>	<b>6</b>
	Memory: Prefetching Beyond Valid Memory	6
	Memory: Program Reads From Flash/ROM Memory	6
	Memory: Flash and OTP Prefetch Buffer Overflow	7
	XINTF: XBANK Does Not Properly Extend an Access	8
	SCI: Incorrect Operation of SCI in Address Bit Mode	10
	SCI: Bootloader Does Not Clear the ABD Bit After Auto-Baud Lock	11
	SCI: Bootloader Does Not Clear the ABD Bit Before Auto-Baud Lock	11
	eCAN: Abort Acknowledge Bit Not Set	12
	WD: WDFLAG Bit Does Not Work as Intended	13
	ADC: EOS BUF1/2 Bits in ADCST Corrupted at the End of Conversion of Sequencer 1/2 When INT MOD SEQ1/2 is Enabled	13
	eCAN: CPU Access to the eCAN Registers May Fail If It Is in Conflict With an eCAN Access to the eCAN Registers	14
	SPI: Slave-Mode Operation	15
	McBSP: Receive FIFO Read Conflict	15
	McBSP: Read Operations Decrement the McBSP FIFO	16
	Clocking: Logic-High Level for XCLKIN Pin	16
	EV: QEP Circuit	17
	ADC: Reserved Bits in Autosequence Status Register (ADCASEQSR)	18
	ADC: Result Register Update Delay	18
	ADC: Sequencer Reset While Dual Sequencers Are Running	19
2.1	Advisory for TMS320F281x Revision D Only	19
	DEVICE-ID: Register of the Silicon Same for Rev C and D	19
2.2	Advisories for TMS320F281x Revisions 0 and A Only	20
	ADC: Device Has Higher Gain Error Than the Design Goal of 1% FSR on All of the B0–B7 Channels	20
	ADC: Device Has Higher Offset Error Than the Design Goal (0.5 to 1%) on Some Channels	21
	ADC – Device Has Higher Non-Linearity Than the Design Goal of 2 LSBs	21
	XINTF: XREADY Signal is not Sampled Properly When Using Asynchronous Sampling Mode	22
	Memory: Set Device Emulation Register Bits for On-Chip RAM Performance	22
	Memory: OTP Memory	23
	WD: A Low Output on GPIOF14 Can Disable the PLL and Watchdog if the Watchdog Fires a Reset	23

PLL: PLL x4 and x8 Multiplier Ratios ..... 23

Low-Power Modes – STANDBY Mode ..... 24

**3 Documentation Support ..... 25**

**4 Revision History ..... 26**

# 1 Introduction

This document describes the silicon updates to the functional specifications for the TMS320F2810, TMS320F2811, and TMS320F2812 digital signal processors. The updates are applicable to:

- TMS320C2812 and TMS320F2812 (179-ball MicroStar BGA™, GHH, and ZHH suffix)
- TMS320C2812 and TMS320F2812 (176-pin low-profile quad flatpack [LQFP], PGF suffix)
- TMS320C2810, TMS320C2811, TMS320C2811, and TMS320F2811 (128-pin LQFP, PBK suffix))

## 1.1 Device and Development Tool Support Nomenclature

To designate the stages in the product development cycle, TI assigns prefixes to the part numbers of all TMS320™ DSP devices and support tools. Each TMS320™ DSP commercial family member has one of three prefixes: TMX, TMP, or TMS. Texas Instruments recommends two of three possible prefix designators for support tools: TMDX and TMDS. These prefixes represent evolutionary stages of product development from engineering prototypes (TMDX) through fully qualified production devices/tools (TMDS).

Device development evolutionary flow:

<b>TMX</b>	Experimental device that is not necessarily representative of the final device's electrical specifications
<b>TMP</b>	Final silicon die that conforms to the device's electrical specifications but has not completed quality and reliability verification
<b>TMS</b>	Fully qualified production device

Support tool development evolutionary flow:

<b>TMDX</b>	Development-support product that has not yet completed Texas Instruments internal qualification testing.
<b>TMDS</b>	Fully qualified development-support product

TMX and TMP devices and TMDX development-support tools are shipped against the following disclaimer:

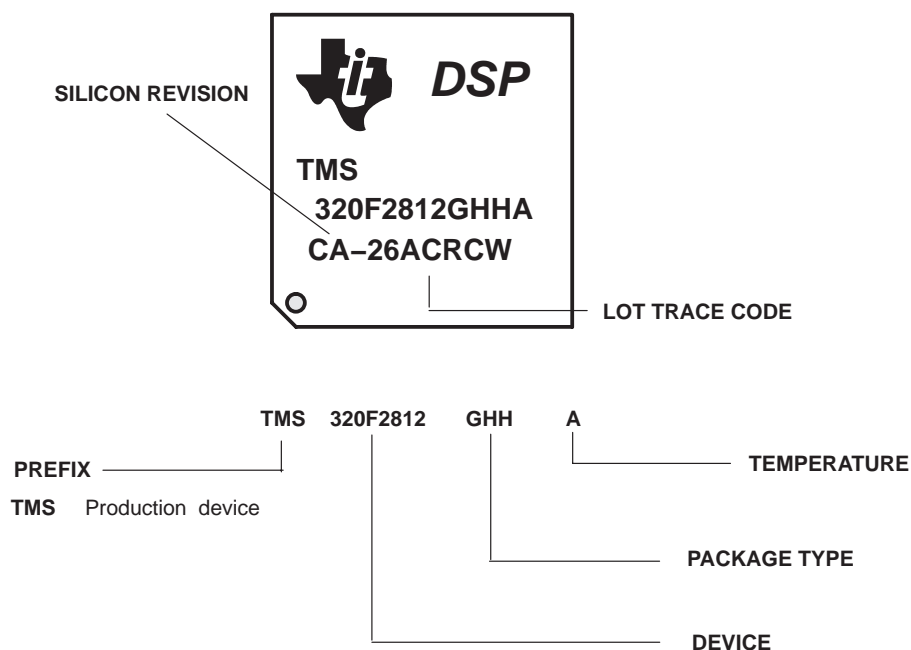
"Developmental product is intended for internal evaluation purposes."

TMS devices and TMDS development-support tools have been characterized fully, and the quality and reliability of the device have been demonstrated fully. TI's standard warranty applies.

Predictions show that prototype devices (TMX or TMP) have a greater failure rate than the standard production devices. Texas Instruments recommends that these devices not be used in any production system because their expected end-use failure rate still is undefined. Only qualified production devices are to be used.

## 1.2 Device Markings

Figure 1 provides an example of the TMS320F281x device markings and defines each of the markings. The device revision can be determined by the symbols marked on the top of the package as shown in Figure 1. Some prototype devices may have markings different from those illustrated.



**Figure 1. Example Markings for 281x Packages**

**Table 1. Determining Silicon Revision From Lot Trace Code (F281x)**

Second Letter in Prefix of Lot Trace Code	Silicon Revision	Revision ID (0x0883)	Comments
<b>Blank</b> (no second letter in prefix)	Indicates Revision 0	0x0000	This silicon revision is available as TMX only.
<b>A</b>	Indicates Revision A	0x0001	This silicon revision is available as TMX only.
<b>B</b>	Indicates Revision B	0x0002	Internal
<b>C</b>	Indicates Revision C	0x0003	TMP/TMX/TMS
<b>D</b>	Indicates Revision D	0x0003	Internal
<b>E</b>	Indicates Revision E	0x0005	Production device (TMS)
<b>F</b>	Indicates Revision F	0x0006	Internal
<b>G</b>	Indicates Revision G	0x0007	Production device (TMS)

## 2 Known Design Marginality/Exceptions to Functional Specifications

### Advisory

*Memory: Prefetching Beyond Valid Memory*

**Revision(s) Affected:** TMS320F281x: 0, A, B, C, D, E, F and G  
TMS320C281x: 0, A, B

**Details:** The C28x CPU prefetches instructions beyond those currently active in its pipeline. If the prefetch occurs past the end of valid memory, then the CPU may receive an invalid opcode.

**Workaround:** The prefetch queue is 8x16 words in depth. Therefore, code should not come within 8 words of the end of valid memory. This restriction applies to all memory regions and all memory types (Flash/ROM, OTP, SARAM, XINTF) on the device. Prefetching across the boundary between two valid memory blocks is ok.

Example 1: M1 ends at address 0x7FF and is not followed by another memory block. Code in M1 should be stored no farther than address 0x7F7. Addresses 0x7F8–0x7FF should not be used for code.

Example 2: M0 ends at address 0x3FF and valid memory (M1) follows it. Code in M0 can be stored up to and including address 0x3FF. Code can also cross into M1 up to and including address 0x7F7

### Advisory

*Memory: Program Reads From Flash/ROM Memory*

**Revision(s) Affected:** TMS320F281x: 0, A, B, C, D, E, F and G  
TMS320C281x: 0, A, B

**Details:** When an interrupt occurs while program code is executing instructions from the address range 0x3F7FF0 through 0x3F7FF7, it is possible that subsequent data reads from the Flash/ROM will return all zeros.

**Workaround:** Do not place program code within this address range. This range can be used for data variable storage.

**Advisory***Memory: Flash and OTP Prefetch Buffer Overflow*

**Revision(s) Affected:** TMS320F281x: 0, A, B, C, D, E, F and G  
TMS320C281x: 0, A, B

**Details:** This advisory applies to code executing from flash or OTP with the flash prefetch buffer enabled. On ROM devices this applies to the ROM that replaces flash and OTP.

The flash prefetch buffer may overflow if a SBF or BF instruction is within eight 16-bit words preceding an operation using indirect or direct program-memory addressing. The window where this can occur is shown in Example 1.

**Example 1.**

Address	Offset	
0x0000		BF LSW (32-bit opcode)
0x0001		BF MSW or SBF (16-bit opcode)
-----		
0x0002	SBF/BF + 1 word	↑
0x0003	SBF/BF + 2 words	
0x0004	SBF/BF + 3 words	If an instruction within this window uses program-memory, addressing it can cause the flash prefetch buffer to overflow.
0x0005	SBF/BF + 4 words	
0x0006	SBF/BF + 5 words	
0x0007	SBF/BF + 6 words	
0x0008	SBF/BF + 7 words	
0x0009	SBF/BF + 8 words	↓
-----		
0x0010	SBF/BF + 9 words	

Whether or not an overflow actually occurs depends on the instruction sequence, flash wait states and CPU pipeline stalls. If an overflow occurs it will result in execution of invalid opcodes. Instructions that use program-memory addressing include MAC/XMAC, DMAC/XMACD, QMACL, IMACL, PREAD/XPREAD, and PWRITE/XPWRITE.

- Workaround(s):**
1. Hand-coded assembly:  
Use the SB/B instructions instead of SBF/BF for code targeted to execute from flash or OTP. The SB/B instructions are more efficient in wait-stated memory so a performance improvement may also be seen.
  2. Compiler-generated assembly:  
Use the compiler switch `-me` to force the compiler to generate SB/B instructions instead of SBF/BF instructions. In heavily wait stated memory, the SB/B instructions are more efficient than SBF/BF. In SARAM the SBF/BF instructions are more efficient. Therefore this switch should be applied as follows:
    - Use the compiler switch `-me` on source code that runs from flash or OTP.
    - Do not use the compiler switch `-me` on source code that runs from SARAM.
    - Use `-me` if a file contains functions that run from flash as well as functions that run from SARAM.

The `-me` switch is available in the C28x compiler as of V4.1.4, V5.0.0 and above.

**Advisory***XINTF: XBANK Does Not Properly Extend an Access*

**Revision(s) Affected:** TMS320F281x: 0, A, B, C, D, E, F and G  
TMS320C281x: 0, A, B

**Details:** When XTIMCLK is not equal to SYSCLKOUT the XBANK logic may not properly delay a pending access. This occurs for some combinations of XINTF zone wait states and XBANK delay cycles. There are two cases when this occurs.

**Case 1: When  $XTIMCLK = 1/2 SYSCLKOUT$  and  $XCLKOUT = XTIMCLK$**

A pending access may not be delayed by the XBANK logic if either:

- $WLEAD + WACTIVE + WTRAIL \leq XBANK[BCYC]$  or
- $RLEAD + RACTIVE + RTRAIL \leq XBANK[BCYC]$

Where WLEAD, WACTIVE, WTRAIL, RLEAD, RACTIVE, RTRAIL are defined as shown in Table 2.

**Table 2. Pending Access Relationships**

	<b>X2TIMING = 0</b>	<b>X2TIMING = 1</b>
<b>WLEAD</b>	$XTIMING \times [XWRLEAD]$	$XTIMING \times [XWRLEAD] \times 2$
<b>WACTIVE</b>	$XTIMING \times [XWRACTIVE] + 1$	$XTIMING \times [XWRACTIVE] \times 2 + 1$
<b>WTRAIL</b>	$XTIMING \times [XWRTRAIL]$	$XTIMING \times [XWRTRAIL] \times 2$
<b>RLEAD</b>	$XTIMING \times [XRDLEAD]$	$XTIMING \times [XRDLEAD] \times 2$
<b>RACTIVE</b>	$XTIMING \times [XRDACTIVE] + 1$	$XTIMING \times [XRDACTIVE] \times 2 + 1$
<b>RTRAIL</b>	$XTIMING \times [XRDTRAIL]$	$XTIMING \times [XRDTRAIL] \times 2$

In Table 2, XTIMINGx refers to the XTIMING register for Zone x. When XBANK delay cycles are added between two accesses, Zone x refers to the first zone in the sequence. For example: if XBANK[BANK] = 7, then delay cycles will be added to any access into or out of Zone 7. This means:

- Access to Zone 0 followed by Zone 7: the timing of Zone 0 is critical.
- Access to Zone 1 followed by Zone 7: the timing of Zone 1 is critical.
- Access to Zone 7 followed by Zone 0: the timing of Zone 7 is critical.

Thus, the timing of any zone involved in bank switching must be considered.

**Case 2) When  $XTIMCKLK = 1/2 SYSCLKOUT$  and  $XCLKOUT = 1/2 XTIMCLK$ :**

A pending access may not be delayed properly by the XBANK logic if XBANK[BCYC] = 4 or XBANK[BCYC] = 6.



*XINTF: XBANK Does Not Properly Extend an Access (continued)***Workaround(s):**

- Case 1) If  $XTIMCLK = 1/2 \text{ SYSCLKOUT}$  and  $XCLKOUT = XTIMCLK$  then select:**

$$XBANK[BCYC] \leq WLEAD + WACTIVE + WTRAIL \quad \text{and}$$

$$XBANK[BCYC] \leq RLEAD + RACTIVE + RTRAIL$$

When XBANK delay cycles are added between two accesses, the timing restriction applies to the first zone accessed as described earlier. The timing of any zone involved in bank switching must be considered.

Table 3 shows examples of valid XBANK[BCYC] selections. This list is not exhaustive.

**Table 3. Examples of Valid XBANK Selections**

XWRLEAD XRDLEAD	WRACTIVE XRDACTIVE	XWRTRAIL XRDTRAIL	X2TIMING	WLEAD + WACTIVE + WTRAIL	Choose XBANK[BCYC]
1	2	1	0	5	< 5
1	3	1	0	6	< 6
2	3	1	0	7	< 7
1	0	1	1	3	< 3
1	1	0	1	5	< 5
1	1	1	1	7	< 7

- Case 2: If  $XTIMCLK = 1/2 \text{ SYSCLKOUT}$  and  $XCLKOUT = 1/2 \text{ XTIMCLK}$ , then select:**

$$XBANK[BCYC] \neq 4 \text{ and}$$

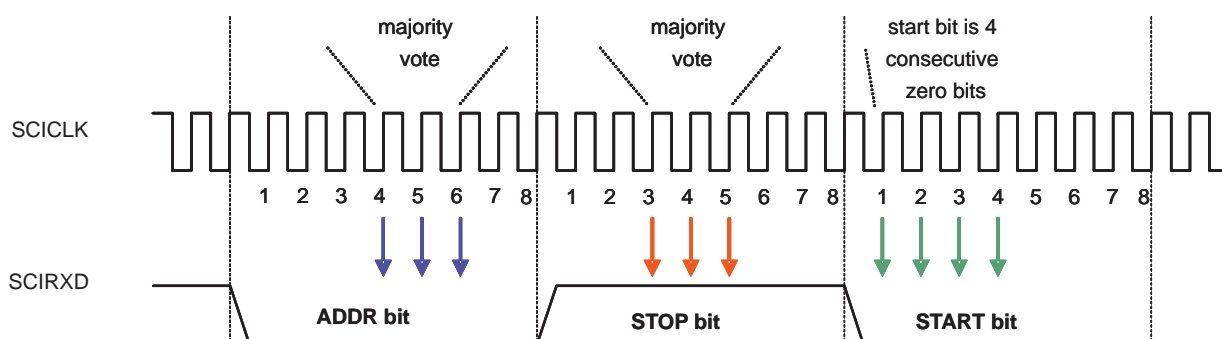
$$XBANK[BCYC] \neq 6$$

**Advisory***SCI: Incorrect Operation of SCI in Address Bit Mode*

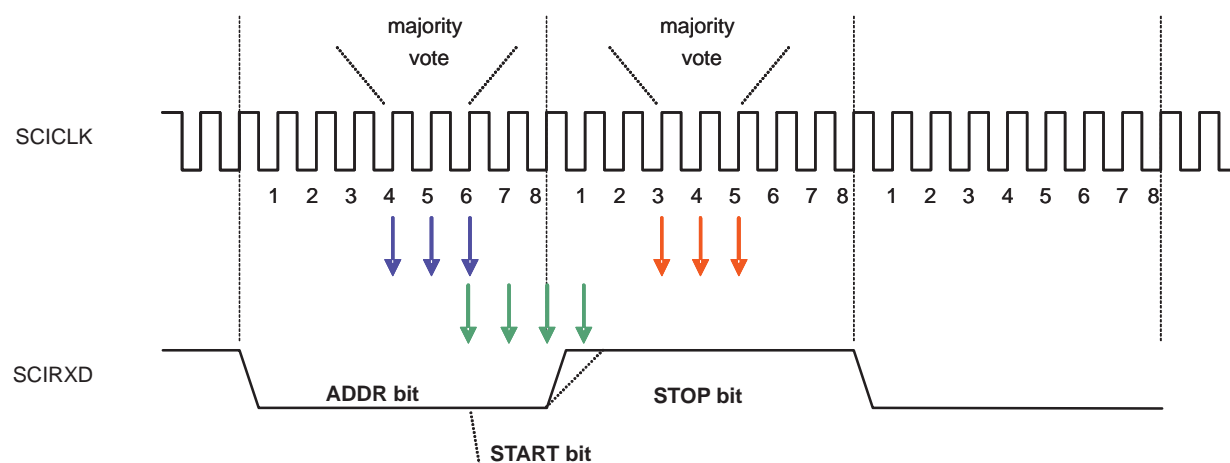
**Revision(s) Affected:** TMS320F281x: 0, A, B, C, D, E, F and G  
TMS320C281x: 0, A, B

**Details:** SCI does not look for STOP bit after the ADDR bit. Instead, SCI starts looking for the start bit beginning on sub-sample 6 of the ADDR bit. Slow rise-time from ADDR to STOP bit can cause false START bit to occur since 4th sub-sample for the start bit may be sensed low.

*Expected Operation:*



*Erroneous Operation:*



**Figure 2. Difference Between Expected and Erroneous Operation of START Bit**

**Workaround:** Program the baud rate of the SCI to be slightly slower than the actual. This will cause the 4th sub-sample of the false START bit to be delayed in time, and therefore occur more towards the middle of the STOP bit (away from the signal transition region). The amount of baud slowing needed depends on the rise-time of the signal in the system. Alternatively, IDLE mode of the SCI module may be used, if applicable.

**Advisory***SCI: Bootloader Does Not Clear the ABD Bit After Auto-Baud Lock*

**Revision(s) Affected:** TMS320F281x: 0, A, B, C, D, E, F and G  
TMS320C281x: 0, A, B

**Details:** The SCI ROM bootloader code does not clear the auto-baud detect (ABD) bit in the SCIFFCT register after the auto-baud process completes. If the SCI-A port is used after the bootloader is executed, transmit interrupts (SCITXINTA) will not be able to occur, nor will the auto-baud lock feature of SCI-A work correctly.

**Workaround:** If the SCI bootloader has been executed, the user's application code should clear the ABD bit by writing a 1 to ABD CLR (bit 14) in the SCIFFCT register before enabling the SCITXINTA interrupt, and before using the auto-baud feature.

**Advisory***SCI: Bootloader Does Not Clear the ABD Bit Before Auto-Baud Lock*

**Revision(s) Affected:** TMS320F281x: 0, A, B, C, D, E, F and G  
TMS320C281x: 0, A, B

**Details:** The SCI ROM bootloader code does not correctly clear the Auto-Baud Detect (ABD) bit in the SCIFFCT register before the auto-baud process begins. The bootloader code fragment is shown below:

```
// Prepare for autobaud detection
// Set the CDC bit to enable autobaud detection
// and clear the ABD bit
SCIAREgs.SCIFFCT.all = 0x2000;
```

The comments incorrectly state that the ABD bit is cleared. The ABD bit is cleared by writing a 1 to the ABD\_CLR bit (bit 14) of the SCIFFCT register. This situation does not hinder operation from power up or reset because the ABD bit is cleared by default after reset. If, however, the bootloader is invoked a second time from software, then the ABD bit will not be cleared and autobaud lock will not occur properly.

**Workaround:** If the bootloader is going to be re-invoked by software, the user's code must first clear the ABD bit before calling the bootloader. To do this, write a 1 to the ABD CLR bit (bit 14) in the SCIFFCT register.

**Advisory***eCAN: Abort Acknowledge Bit Not Set*

**Revision(s) Affected:** TMS320F281x: 0, A, B, C, D, E, F and G  
TMS320C281x: 0, A, B

**Details:** After setting a transmission request reset (TRR) register bit to abort a message, there are some rare instances where the TRRn and TRSn bits will clear without setting the abort acknowledge (AAn) bit. The transmission itself is correctly aborted, but no interrupt is asserted and there is no indication of a pending operation.

In order for this rare condition to occur, all of the following conditions must happen:

1. The previous message was not successful, either because of lost arbitration or because no node on the bus was able to acknowledge it or because an error frame resulted from the transmission. The previous message need not be from the same mailbox in which a transmit abort is currently being attempted.
2. The TRRn bit of the mailbox should be set in a CPU cycle immediately following the cycle in which the TRSn bit was set. The TRSn bit remaining set due to incompleteness of transmission satisfies this condition as well. i.e., the TRSn bit could have been set in the past, but the transmission remains incomplete.
3. The TRRn bit must be set in the exact SYSCLKOUT cycle where the CAN module is in idle state for one cycle. The CAN module is said to be in idle state when it is not in the process of receiving/transmitting data.

If these conditions occur, then the TRRn and TRSn bits for the mailbox will clear  $t_{clr}$  SYSCLKOUT cycles after the TRR bit is set where:

$$t_{clr} = ((\text{mailbox\_number}) * 2) + 3 \text{ SYSCLKOUT cycles}$$

The TAn and AAn bits will not be set if this condition occurs. Normally, either the TA or AA bit sets after the TRR bit goes to zero.

**Workaround:** When this problem occurs, the TRRn and TRSn bits will clear within  $t_{clr}$  SYSCLKOUT cycles. To check for this condition, first disable the interrupts. Check the TRRn bit  $t_{clr}$  SYSCLKOUT cycles after setting the TRRn bit to make sure it is still set. A set TRRn bit indicates that the problem did not occur.

If the TRRn bit is cleared, it could be because of the normal end of a message and the corresponding TAn or AAn bit is set. Check both the TAn and AAn bits. If either of the bits is set, then the problem did not occur. If they are both zero, then the problem did occur. Handle the condition like the interrupt service routine would except that the AAn bit does not need clearing now.

If the TAn or AAn bit is set, then the normal interrupt routine will happen when the interrupt is re-enabled.

**Advisory***WD: WDFLAG Bit Does Not Work as Intended*

**Revision(s) Affected:** TMS320F281x: 0, A, B, C, D, E, F and G  
TMS320C281x: 0, A, B

**Details:** The WDFLAG bit in F281x devices cannot be used to reliably distinguish a watchdog-initiated reset from a power-on (or warm) reset. This is because the device expects the  $\overline{\text{XRS}}$  pin to be pulled high (by the external reset circuit) within 4 SYSCLKOUT cycles (8 OSCCLK cycles at power up) after the end of the watchdog-initiated reset pulse, which is 512 OSCCLK cycles. Most of the external  $\overline{\text{XRS}}$  circuits cannot provide the fast rise-time requirement (due to the capacitance); therefore, the WDFLAG bit should not be used in applications .

**Workaround:** None. This bit should not be used.

**Advisory***ADC: EOS BUF1/2 Bits in ADCST Corrupted at the End of Conversion of Sequencer 1/2 When INT MOD SEQ1/2 is Enabled*

**Revision(s) Affected:** TMS320F281x: 0, A, B, C, D, E, F and G  
TMS320C281x: 0, A, B

**Details:** Setting the INT MOD SEQx bit in ADCTRL2 as per the user guide should result in the ADC wrapper generating INT SEQx at the end of every other conversion rather than at the end of every conversion. Also EOS BUFx will be set at the end of every conversion to track the status of the SEQx in use. However, a conversion on SEQ1 will cause EOS\_BUF2 to be set if INT\_MOD\_SEQ2 is enabled for that sequencer, even if INT\_MOD\_SEQ1 is not enabled.

For example, if INT\_MOD\_SEQ2 is set, a conversion on SEQ1 will cause the EOS\_BUF2 bit to be set incorrectly. This will cause INT SEQ2 to be set incorrectly after the next SEQ2 completion. If EOS\_BUF2 is already set (from previous SEQ conversion), a conversion on SEQ1 will cause EOS\_BUF2 to be cleared causing the interrupt to be missed. The above relationship is also true for SEQ2 affecting SEQ1. In all cases, the sequencers do work correctly, with the exception that EOS BUFx gets corrupted.

**Workaround:** Do not use the INT\_MOD\_SEQx feature if both SEQ1 and SEQ2 will be used before two completions of sequence have completed on the INT\_MOD\_SEQ selected sequencer.

**Advisory**

*eCAN: CPU Access to the eCAN Registers May Fail If It Is in Conflict  
With an eCAN Access to the eCAN Registers*

**Revision(s) Affected:** TMS320F281x: 0, A, B, C, D, E, F and G  
TMS320C281x: 0, A, B

**Details:** If contention exists between the CPU and the eCAN controller for access to certain eCAN register areas, a CPU read may erroneously read all zeros (0x00000000), and a CPU write may erroneously fail to execute. Specifically:

1) If the CPU reads the eCAN mailbox RAM area (MSGID, MSGCTRL, MDL, or MDH registers) at the same time that the eCAN controller is accessing (reading or writing) the LAM/MOTO/MOTS register area, the CPU may erroneously read all zeros (0x00000000).

**Workaround:** For all CPU reads from the eCAN mailbox RAM area, check to see if the read returns all zeros. If so, the CPU should perform a second read. If the second read returns zero as well, then the data is correctly zero. If the second read returns a non-zero value, then the second data is the correct value. Note that interrupts must be disabled during the consecutive CPU reads. See Note 5.

**Details:** 2) If the CPU writes to the eCAN mailbox RAM area (MSGID, MSGCTRL, MDL, or MDH register) at the same time that the eCAN controller is accessing (reading or writing) the LAM/MOTO/MOTS register area, the CPU write may fail to execute.

**Workaround:** For all CPU writes to the eCAN mailbox RAM area, the CPU should write the data twice. Note that interrupts must be disabled during the consecutive CPU writes. See Note 5.

**Details:** 3) If the CPU reads the LAM/MOTO/MOTS register area at the same that the eCAN controller is accessing (reading or writing) the eCAN mailbox RAM area (MSGID, MSGCTRL, MDL, or MDH registers), the CPU may erroneously read all zeros (0x00000000).

**Workaround:** For all CPU reads from the LAM/MOTO/MOTS register area, check to see if the read returns all zeros. If so, the CPU should perform a second read. If the second read returns zero as well, then the data is correctly zero. If the second read returns a non-zero value, then the second data is the correct value. Note that interrupts must be disabled during the consecutive CPU reads. See Note 5.

**Details:** 4) If the CPU writes to the LAM/MOTO/MOTS register area at the same that the eCAN controller is accessing (reading or writing) the eCAN mailbox RAM area (MSGID, MSGCTRL, MDL, or MDH registers), the CPU write may fail to execute.

**Workaround:** For all CPU writes to the LAM/MOTO/MOTS register area, the CPU should write the data twice with a minimum of 4 CPU cycles in between the writes. Note that interrupts must be disabled during the consecutive CPU writes. See Note 5.

- NOTES: 1. An example of the eCAN controller reading the LAM/MOTO/MOTS register area is a read of the LAMn register to check if a received message passes the acceptance mask filtering criterion. This happens during reception of a frame.
2. An example of the eCAN controller writing to the LAM/MOTO/MOTS register area is a write to the MOTSn register to update the time-stamp upon successful transmission of a frame.

*eCAN: CPU Access to the eCAN Registers May Fail If It Is in Conflict  
With an eCAN Access to the eCAN Registers (continued)*

3. An example for the eCAN controller attempting to read the mailbox RAM area (MSGID, MSGCTRL, MDL & MDH registers) is right before transmission.
4. An example for the eCAN controller attempting to write to the mailbox RAM area (MSGID, MSGCTRL, MDL & MDH registers) is right after reception.
5. A "C callable assembly" implementation of the workaround can be downloaded from the TI Website (literature number SPRC180).

**Advisory***SPI: Slave-Mode Operation*

**Revision(s) Affected:** TMS320F281x: 0, A, B, C, D, E, F and G  
TMS320C281x: 0, A, B

**Details:** When in slave mode, the SPI does not resynchronize received words based on SPISTE. A spurious SPICLK pulse could therefore throw the data stream out of sync.

**Workaround:** If the circuit board is not noisy enough to generate spurious SPICLK pulses, then this is not an issue. If noise is an issue, then the McBSP in SPI-slave mode may be used, since the McBSP resynchronizes on each new word.

**Advisory***McBSP: Receive FIFO Read Conflict*

**Revision(s) Affected:** TMS320F281x: 0, A, B, C, D, E, F and G  
TMS320C281x: 0, A, B

**Details:** The McBSP peripheral operates with or without FIFOs. The receive FIFO has interrupt generation logic that initiates interrupts based on the 5-bit FIFO status bits (12–8) and interrupt level bits (4–0) in the MFFRX register.

If the CPU reads the receive FIFO while the McBSP module writes new data into the FIFO, there is a potential conflict. The CPU read will not be stalled and read data will not be valid. The FIFO write gets the priority. The receive FIFO will be updated after every word is received in DRR2/1 registers. The DRR2/1 register update time will primarily depend on the word size and CLKR rate. For example for 8-bit word, it should be typically 8 times the CLKR cycle time. This conflict will be more pronounced if data transferred on the receive channel is back-to-back with no delays between words.

**Workaround:** The receive FIFOs should be read based on receive interrupts and within the next word receive time. To avoid the read conflict, additional checks could be used before initiating receive FIFO read. In most McBSP configurations, the FSR is a receiving sync pulse either active high or low (based on the FSR polarity bit) and will go inactive during word transfer time. These active and inactive phases can be detected by checking the FSR flag bit MCFFST (bit 3) register or checking the status of the FSR pin. See the FSR flag bit description for details.

**Advisory***McBSP: Read Operations Decrement the McBSP FIFO*

**Revision(s) Affected:** TMS320F281x: 0, A, B, C, D, E, F and G  
TMS320C281x: 0, A, B

**Details:** A read operation from any of the following locations will cause the McBSP receive FIFO contents to decrement by 1, as if the McBSP DRR1 register had been read:

- 0x7001 Reserved
- 0x7401 EV-A T1CNT
- 0x7C01 Reserved

The actual value read from the location is correct and is not affected by this issue.

**Workaround(s):**

1. Ensure that the McBSP receive FIFO is empty before performing any read operation from any of these addresses.
2. If McBSP traffic is common in the application and a timer count needs to be monitored, consider using a timer other than EV Timer1.

**Advisory***Clocking: Logic-High Level for XCLKIN Pin*

**Revision(s) Affected:** TMS320F281x: 0, A, B, C, D, E, F and G  
TMS320C281x: 0, A, B

**Details:** This advisory is applicable only when an external oscillator is used to clock the device. The X1/XCLKIN pin is referenced to the core power supply ( $V_{DD}$ ), rather than the 3.3-V I/O supply ( $V_{DDIO}$ ). Therefore, the logic-high level for the input clock should not exceed  $V_{DD}$ . This requirement remains the same for future silicon revisions as well.

**Workaround:** A clamping diode may be used to clamp a buffered clock signal to ensure that the logic-high level does not exceed  $V_{DD}$  (1.8 V or 1.9 V). Otherwise, 1.8-V oscillators may be used.



**Advisory**

EV: QEP Circuit

**Revision(s) Affected:** TMS320F281x: 0, A, B, C, D, E, F and G  
TMS320C281x: 0, A, B

**Details:** After a DSP reset, the QEP module fails to detect the first transition that occurs on QEP input pins. (This problem also manifests itself when an external clock is used for the EV timers.) Therefore, if the first transition occurs after a GP timer has been initialized and enabled as the QEP counter (i.e., to use QEP as source of clock), the first transition will not be counted by the GP timer. The result is an error of one count in the GP timer out of a total of 1024 counts for a 256-line encoder, or 4096 counts for a 1024-line encoder. However, the issue is not a concern under any of the following conditions:

1. *The first transition happens **before** the GP timer is initialized and enabled as QEP counter.* This ensures that all transitions are counted after initialization.
2. *After the first index pulse is received and if the index pulse is used to recalibrate the GP Timer (through capture interrupt).* The recalibration corrects the error in the GP timer; therefore, from the time the first index pulse is received, the QEP counter becomes accurate.

**Workaround(s):**

1. Make the first transition happen before the GP timer is initialized and enabled as QEP counter. This is usually the case because typically the rotor shaft is locked to a known position before the GP timer is initialized. Locking the rotor shaft will generate transitions on QEP input pins, unless the rotor shaft is exactly aligned to the known position (which is a rare case). Disturbing the rotor shaft on purpose takes care of the rare case.
2. Use the index pulse of the encoder to recalibrate the GP timer used as QEP counter.
3. The counter has to be forced to count before the application actually uses the QEP. During initialization, configure the internal clock (HSPCLK) to be the counter source. After the first count is done, the counter should be reconfigured for external signals (QEP/TCLKIN) and reset to 0. Now the counter will also count the first edge of the QEP.

**Advisory***ADC: Reserved Bits in Autosequence Status Register (ADCASEQSR)*

**Revision(s) Affected:** TMS320F281x: 0, A, B, C, D, E, F and G  
TMS320C281x: 0, A, B

**Details:** SEQ2 STATE2–0 and SEQ1 STATE3–0 bit fields (bits 6 through 0) are the pointers of SEQ2 and SEQ1, respectively. These bits are reserved for TI testing and should not be used in customer applications.

**Workaround:** None

**Advisory***ADC: Result Register Update Delay*

**Revision(s) Affected:** TMS320F281x: 0, A, B, C, D, E, F and G  
TMS320C281x: 0, A, B

**Details:** The ADC result status flags INT\_SEQ1 and INT\_SEQ2 bit fields (bits 0 and 1 respectively) in the ADC\_ST\_FLG register indicate the availability of new ADC results after conversions and initiation of the ADC interrupts.

The update of the ADC result register requires one extra ADC cycle to complete after the status flags INT\_SEQ1 and INT\_SEQ2 bit(s) are set. The result of reading the result register prior to this extra cycle will result in old data being read (reset value/previous conversion result).

If auto-sequencers are enabled with a non-zero value in the MAXCONV register, the last result register update takes an additional ADC cycle from the time the INT\_SEQ1 or INT\_SEQ2 flag is set.

**Workaround:** Delay the read of the ADC result register(s) by at least one ADC clock period. This delay can be implemented by using software delay loops.

If the ADC result register(s) are read using the ADC interrupt, rather than polling, the wait period introduced by the ISR (interrupt service routine) could minimize the delay needed in software. This ISR branching delay is generally greater than 8 SYSCLKOUT cycles.

The ratio of the ADC clock (ADCCLK) to the CPU clock (SYSCLKOUT) determines the size of the software delay. For example, if ADCCLK = 10 MHz the software delay should be at least 100 ns.

**Timing example to estimate the software delay:**

- 1) Get the HSPCLK prescaler value – HSPCP
- 2) Get the ADCCLK prescaler value – ADCCLKPS
- 3) Get the CPS (ADCCTRL1[7]) value – CPS
- 4) Software wait-period in CPU cycles (SYSCLKOUT) before the ADC result register read is defined as:

Software wait = (HSPCP \* 2) \* (ADCCLKPS \* 2) \* (CPS + 1) cycles

If HSPCP or ADCCLKPS is 0, then the respective terms should be (HSPCP + 1) or (ADCCLKPS + 1)

**Advisory***ADC: Sequencer Reset While Dual Sequencers Are Running*

**Revision(s) Affected:** TMS320F281x: 0, A, B, C, D, E, F and G  
TMS320C281x: 0, A, B

**Details:** In the TMS320F2812/TMS320F2811/TMS320F2810 on-chip ADC, there are two sequencers for performing ADC conversions; SEQ1 and SEQ2. If one of the sequencers is reset while the other sequencer is running, it will result in the running sequencer never completing its current sequence. The sequencer busy bit (bit 3/bit2 in ADCST register) for the sequencer will remain active and an “End-of-sequence (EOS)” interrupt for the running sequencer will never be generated. For example, if SEQ1 is reset while SEQ2 is performing a sequence, then SEQ2 will never complete.

**Workaround:** If dual sequencers are enabled, then the software handling the ADC module should make sure that SEQ1 BSY/ SEQ2 BSY bits are not set before performing a reset of either sequencer.

## 2.1 Advisory for TMS320F281x Revision D Only

**Advisory***DEVICE-ID: Register of the Silicon Same for Rev C and D*

**Revision(s) Affected:** TMS320F281x: D

**Details:** The DEVICE-ID register of the rev D silicon contains the same value (0x0003) as that of the rev C silicon.

**Workaround:** The next revision (Rev E) of the silicon has a DEVICE-ID value of 0x0005.

## 2.2 Advisories for TMS320F281x Revisions 0 and A Only

These revisions are not to be used in development or production.

### Advisory

*ADC: Device Has Higher Gain Error Than the Design Goal of 1% FSR on All of the B0–B7 Channels*

**Revision(s) Affected:** TMS320F281x: 0 and A

**Details:** The device has a higher gain error than the design goal of 1% FSR on all of the B0–B7 channels. The gain error varies across channels A0–A7 and B0–B7.

Based on the current data obtained on B group channels, all B group channels show a uniform gain error as high as 2 to 3%.

**Workaround:** The channel-to-channel gain error data across channels are listed in Table 4. This should help in calibrating in software or hardware. This was fixed in Revision B silicon.

**Table 4. Channel-to-Channel Offset Error Data Across Channels (176-Pin PGF)**

ADC Channels	A0	A1	A2	A3	A4	A5	A6	A7	B0	B1	B2	B3	B4	B5	B6	B7
Gain Error in %	0.20	0.18	0.52	0.53	0.53	0.55	0.54	0.54	2.92	2.92	2.92	2.93	2.93	2.93	2.93	2.97
Offset in LSB Counts	14.80	15.64	4.86	9.82	5.82	–14.57	–13.98	–31.78	20.94	21.98	22.48	23.39	22.39	23.14	23.64	24.91

### Note:

The data provided are typical values only. These values are obtained from bench characterization at room temperature on a few devices.

TMX samples are not fully screened for all ADC parameters. If there are devices that have worse performance than suggested issues/values, it is recommended that the part be replaced.

**Advisory**

*ADC: Device Has Higher Offset Error Than the Design Goal (0.5 to 1%) on Some Channels*

**Revision(s) Affected:** TMS320F281x: 0 and A

**Details:** Based on the current data obtained on all channels, some channels show an offset error as high as 1%.

**Workaround:** The channel-to-channel offset error data across channels are listed in Table 4. This should help in calibrating in software or hardware. This was fixed in Revision B silicon.

**Note:**

The data provided are typical values only. These values are obtained from bench characterization at room temperature on a few devices.

TMX samples are not fully screened for all ADC parameters. If there are devices that have worse performance than suggested issues/values, it is recommended that the part be replaced.

**Advisory**

*ADC – Device Has Higher Non-Linearity Than the Design Goal of 2 LSBs*

**Revision(s) Affected:** TMS320F281x: 0 and A

**Details:** Based on the current data obtained on all channels, some channels show non-linearity as high as 12 LSBs in the mid-scale range. That is, the mid-range conversions will be off by about 12 LSB counts.

**Workaround:** This issue is corrected in the next revision of the silicon. The following option could be used to correct for INL errors only for the TMX Revision A silicon.

The INL issue is across all channels. Use the ADC results only for 9-bit data accuracy on this revision of the silicon and ignore the rest of the bits. This will mitigate the INL effect in the application provided the algorithm can tolerate 9-bit accuracy.

**Note:**

The data provided are typical values only. These values are obtained from bench characterization at room temperature on a few devices.

TMX samples are not fully screened for all ADC parameters. If there are devices that have worse performance than suggested issues/values, it is recommended that the part be replaced.

**Advisory**

*XINTF: XREADY Signal is not Sampled Properly When Using Asynchronous Sampling Mode*

**Revision(s) Affected:** TMS320F281x: 0 and A

**Details:** In case of asynchronous ready mode, if the XREADY signal is high within the Lead period, then access will complete in the number of cycles programmed in LEAD + ACTIVE + TRAIL counters even if XREADY goes low before the start of the ACTIVE period. In this case, XREADY is not being used properly to extend the access.

**Workaround:** Try one of the following possible workarounds:

- Ensure that the XREADY signal is not low at the start of an access when using asynchronous sampling mode. If the XINTF sees the XREADY signal low from the start of an access, then the ACTIVE period will be extended as desired.
- Use the XTIMING register wait-state values to extend the access such that timings are met without using XREADY.
- Use the synchronous XREADY sampling mode. This problem is not observed in synchronous mode.

This issue is fixed in the next revision of the silicon.

**Advisory**

*Memory: Set Device Emulation Register Bits for On-Chip RAM Performance*

**Revision(s) Affected:** TMS320F281x: 0 and A

**Details:** To get the best performance of on-chip RAM blocks M0/M1/L0/L1/H0, the internal control register bits have to be enabled. The bits are in the Device Emulation Registers.

**Workaround:** All device initialization code should include the following register updates. These are EALLOW-protected registers.

Register Address	Value
0x950	0x0300
0x951	0x0300
0x952	0x0300
0x953	0x0300
0x954	0x0300

**Code Example:**

```

EALLOW
MOVL    XAR1, #0x0950
MOVL    XAR2, #0x0300
MOV     *XAR1++, AR2
MOV     *XAR1++, AR2
MOV     *XAR1++, AR2
MOV     *XAR1++, AR2
MOV     *XAR1++, AR2
EDIS

```

The Code Composer GEL init files will initialize these for emulation and debug environment. From the next silicon revision onward, this initialization is automatically done upon reset.

**Advisory***Memory: OTP Memory*

**Revision(s) Affected:** TMS320F281x: 0 and A

**Details:** The 1K-word OTP memory is not available.

**Workaround:** This is fixed in the next revision of the silicon.

**Advisory***WD: A Low Output on GPIOF14 Can Disable the PLL and Watchdog if the Watchdog Fires a Reset*

**Revision(s) Affected:** TMS320F281x: 0 and A

**Details:** If, during program execution, the XF\_ $\overline{\text{XPLLDIS}}$ /GPIOF14 signal is changed to either of the following:

- A general-purpose output and driven low
- The XF functionality and driven low

and a watchdog reset occurs, then the low output state of the XF\_ $\overline{\text{XPLLDIS}}$ /GPIOF14 pin will be latched into the  $\overline{\text{XPLLDIS}}$  signal. The result of this is that the PLL and the reset function of the watchdog will be disabled. The watchdog itself is not disabled.

**Workaround:** One of the following workarounds can be used:

- Do not toggle XF/GPIOF14 in user code. Instead, use another GPIO signal for status.
- Set the watchdog to fire an interrupt instead of reset.

This is fixed in the next revision of the silicon.

**Advisory***PLL: PLL x4 and x8 Multiplier Ratios*

**Revision(s) Affected:** TMS320F281x: 0 and A

**Details:** When the PLL multiplier is set to x4 or x8 (by writing 0004 or 0008, respectively, in the PLLCR register), the watchdog is re-enabled and resets the device upon a WD overflow. With noisy board conditions, this problem may be observed with other PLL multipliers as well.

**Workaround:** Do not use these multiplier values for these revisions.  
This is fixed in the next revision of the silicon.

**Advisory***Low-Power Modes – STANDBY Mode*

**Revision(s) Affected:** TMS320F281x: 0 and A

**Details:** When the device is put into STANDBY mode, the watchdog is re-enabled and resets the device upon a WD overflow.

**Workaround:** Do not use the STANDBY mode for these revisions.  
This is fixed in the next revision of the silicon.



### 3 Documentation Support

For device-specific data sheets and related documentation, visit the TI web site at: <http://www.ti.com>.

For further information regarding the TMS320F2810 and TMS320F2812, please see the following publication:

- *TMS320F2810, TMS320F2811, TMS320F2812, TMS320C2810, TMS320C2811, TMS320C2812 Digital Signal Processors* data manual (literature number SPRS174)

## 4 Revision History

This revision history highlights the technical changes made to the SPRZ193K to make it an SPRZ193L revision.

**Scope:** See table below.

PAGE(S) NO.	ADDITIONS/CHANGES/DELETIONS
Global	For all advisories that affect TMS320C281x: <ul style="list-style-type: none"><li>– Changed “Revision(s) Affected” from “TMS320C281x: 0, A” to “TMS320C281x: 0, A, <b>B</b>”</li></ul>
7	“Memory: Flash and OTP Prefetch Buffer Overflow” advisory: <ul style="list-style-type: none"><li>– Updated Workaround(s)</li></ul>

## 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 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. Reproduction of this information with alteration is an unfair and deceptive business practice. TI is not responsible or liable for such altered documentation. Information of third parties may be subject to additional restrictions.

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.

TI products are not authorized for use in safety-critical applications (such as life support) where a failure of the TI product would reasonably be expected to cause severe personal injury or death, unless officers of the parties have executed an agreement specifically governing such use. Buyers represent that they have all necessary expertise in the safety and regulatory ramifications of their applications, and acknowledge and agree that they are solely responsible for all legal, regulatory and safety-related requirements concerning their products and any use of TI products in such safety-critical applications, notwithstanding any applications-related information or support that may be provided by TI. Further, Buyers must fully indemnify TI and its representatives against any damages arising out of the use of TI products in such safety-critical applications.

TI products are neither designed nor intended for use in military/aerospace applications or environments unless the TI products are specifically designated by TI as military-grade or "enhanced plastic." Only products designated by TI as military-grade meet military specifications. Buyers acknowledge and agree that any such use of TI products which TI has not designated as military-grade is solely at the Buyer's risk, and that they are solely responsible for compliance with all legal and regulatory requirements in connection with such use.

TI products are neither designed nor intended for use in automotive applications or environments unless the specific TI products are designated by TI as compliant with ISO/TS 16949 requirements. Buyers acknowledge and agree that, if they use any non-designated products in automotive applications, TI will not be responsible for any failure to meet such requirements.

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

### Products

Amplifiers	<a href="http://amplifier.ti.com">amplifier.ti.com</a>
Data Converters	<a href="http://dataconverter.ti.com">dataconverter.ti.com</a>
DLP® Products	<a href="http://www.dlp.com">www.dlp.com</a>
DSP	<a href="http://dsp.ti.com">dsp.ti.com</a>
Clocks and Timers	<a href="http://www.ti.com/clocks">www.ti.com/clocks</a>
Interface	<a href="http://interface.ti.com">interface.ti.com</a>
Logic	<a href="http://logic.ti.com">logic.ti.com</a>
Power Mgmt	<a href="http://power.ti.com">power.ti.com</a>
Microcontrollers	<a href="http://microcontroller.ti.com">microcontroller.ti.com</a>
RFID	<a href="http://www.ti-rfid.com">www.ti-rfid.com</a>
RF/IF and ZigBee® Solutions	<a href="http://www.ti.com/lprf">www.ti.com/lprf</a>

### Applications

Audio	<a href="http://www.ti.com/audio">www.ti.com/audio</a>
Automotive	<a href="http://www.ti.com/automotive">www.ti.com/automotive</a>
Broadband	<a href="http://www.ti.com/broadband">www.ti.com/broadband</a>
Digital Control	<a href="http://www.ti.com/digitalcontrol">www.ti.com/digitalcontrol</a>
Medical	<a href="http://www.ti.com/medical">www.ti.com/medical</a>
Military	<a href="http://www.ti.com/military">www.ti.com/military</a>
Optical Networking	<a href="http://www.ti.com/opticalnetwork">www.ti.com/opticalnetwork</a>
Security	<a href="http://www.ti.com/security">www.ti.com/security</a>
Telephony	<a href="http://www.ti.com/telephony">www.ti.com/telephony</a>
Video & Imaging	<a href="http://www.ti.com/video">www.ti.com/video</a>
Wireless	<a href="http://www.ti.com/wireless">www.ti.com/wireless</a>

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265  
Copyright © 2009, Texas Instruments Incorporated