TMS320C6204

Digital Signal Processor

Silicon Errata

SPRZ180D
January 2002

TEXAS INSTRUMENTS

Contents

1 Introduction . . . 3
1.1 Quality and Reliability Conditions . . . 3
TMX Definition . . . 3
TMP Definition . . . 3
TMS Definition . . . 3
1.2 Revision Identification . . . 4
2 Silicon Revision 1.2 Known Design Exceptions to Functional Specifications . . . 5
Advisory 1.2.1 Internal data RAM: Corruption After STB . . . 5
3 Silicon Revision 1.0 Known Design Exceptions to Functional Specifications . . . 6
Advisory 1.0.1 Data Corruption of STB on 0–1 or 2–3 Byte-Boundary . . . 6
Advisory 1.0.2 XBUS: XBUS and DMA Hang When Using XRDY During ASYNC I/O Read . . . 7
4 Documentation Support . . . 7

1 Introduction

This document describes the silicon updates to the functional specifications for the TMS320C6204, silicon release 1.3.

1.1 Quality and Reliability Conditions

TMX Definition

Texas Instruments (TI) does not warranty either (1) electrical performance to specification, or (2) product reliability for products classified as “TMX.” By definition, the product has not completed data sheet verification or reliability performance qualification according to TI Quality Systems Specifications.

The mere fact that a “TMX” device was tested over a particular temperature and voltage ranges should not, in any way, be construed as a warranty of performance.

TMP Definition

TI does not warranty product reliability for products classified as “TMP.” By definition, the product has not completed reliability performance qualification according to TI Quality Systems Specifications; however, products are tested to a published electrical and mechanical specification.

TMS Definition

Fully-qualified production device

1.2 Revision Identification

The device revision can be determined by the lot trace code marked on the top of the package. The location for the lot trace codes are shown in Figure 1 and Table 1.

Figure 1. Example, Lot Trace Code for TMS320C6204

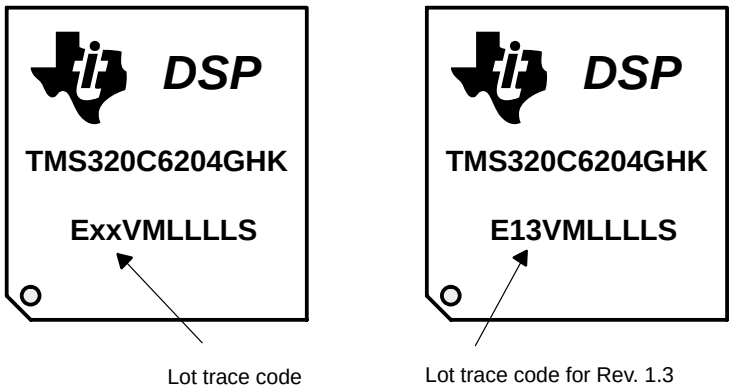


Table 1. Lot Trace Codes

Lot Trace Code	Silicon Revision	Comments
13	1.3	Revision 1.3 has fixed the problem described in Advisory 1.2.1.
12	1.2	This revision is functionally the same as 1.1.
11	1.1	
10	1.0	

2 Silicon Revision 1.2 Known Design Exceptions to Functional Specifications

Advisory 1.2.1 *Internal data RAM: Corruption After STB*

Revision(s) Affected: Rev 1.2, 1.1, and 1.0

Details: When an STB accesses byte 0 from any memory bank (address 8000 xxxy, where y = 0,4,8,C, etc.), data can get corrupted. This occurs when on the very next cycle any load or store access to the memory bank is made as shown in Table 2. The next cycle access must be one that accesses byte 1 within the same bank of RAM.

Table 2. Fail Instruction Sequence Examples

Example Number	Possible Sequence †	
	First	Second
1	STB Reg, dst1[0]	LDB dst2[1], Reg
2	STB Reg, dst1[0]	LDH dst2[0], Reg
3	STB Reg, dst1[0]	LDW dst2[0], Reg
4	STB Reg, dst1[0]	STB Reg, dst2[1]
5	STB Reg, dst1[0]	STH Reg, dst2[0]
6	STB Reg, dst1[0]	STW Reg, dst2[0]

Reg = any general-purpose register
dst = Any data memory address (80000000–8000FFFF)
† * General Fail conditions:

1. The accesses must be within the same bank.
2. The first cycle must be a STB Reg, dst1[0] instruction.
3. The second cycle can be any instruction that accesses byte 1 of the same bank as the previous STB instruction.

Workaround: Use half-word and word stores (STH, STW rather than STB). This can be done from C language by using long, int, and short rather than char.
In Assembly language, add a NOP after an STB instruction or move any following accesses to byte 1 of the same bank one cycle away.

This problem has been fixed on Revision 1.3.

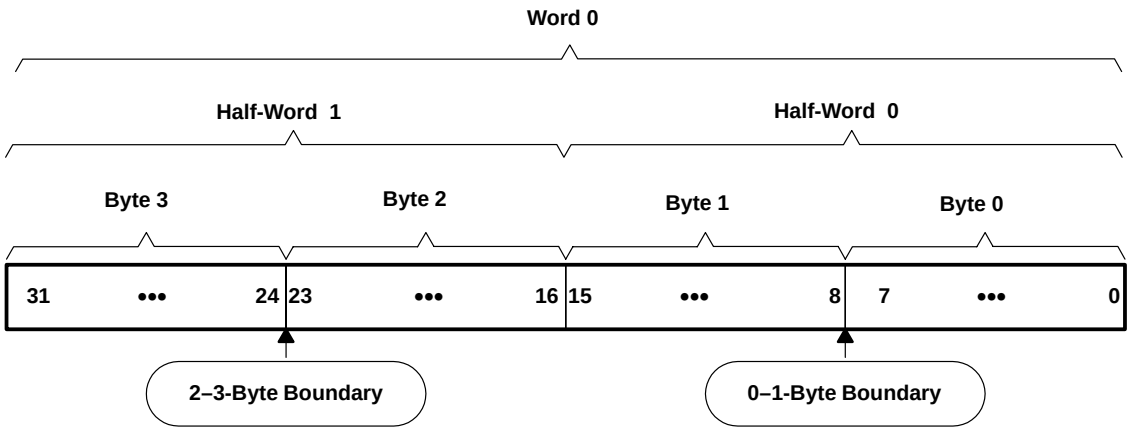
3 Silicon Revision 1.0 Known Design Exceptions to Functional Specifications

Advisory 1.0.1	*Data Corruption of STB on 0–1 or 2–3 Byte-Boundary*

Revision(s) Affected: Rev 1.0

Details: Data corruption occurs when a write to data memory is followed by an instruction that modifies the byte–enables on a 0–1 or a 2–3 byte boundary, as shown in Figure 2. Essentially, an STB instruction acts as an STH instruction. Data corruption occurs at the address referenced by the first instruction. Reads from data memory, as well as both read and write accesses to program memory and external memories do not exhibit this issue.

Figure 2. Diagram showing the 0–1 and 2–3 byte–boundary from a word in data memory



Examples: The following examples show some general coding where the bug occurs.

Table 3. Example Execute Packets

Example Number	Execute Packet	Possible Instructions
1	First	STB reg, byte0, or STH reg, half_word0, or STW reg, word0
	Second	STB reg, byte1, or LDB reg, byte1
2	First	STB reg, byte1, or STH reg, half_word0, or STW reg, word0
	Second	STB reg, byte0, or LDB reg, byte0

Example Number	Execute Packet	Possible Instructions
3	First	STB reg, byte3, or STH reg, half_word1, or STW reg, word0
	Second	STB reg, byte2, or LDB reg, byte2

In these examples, the second execute packet contains an instruction that modifies a byte on either a 0–1-byte-boundary (examples 1 & 2) or a 2–3-byte-boundary (example 3).

Workaround: Utilize half-word and word accesses instead of byte-wide accesses.

Avoid use of the char type from C.

Advisory 1.0.2 *XBUS: XBUS and DMA Hang When Using XRDY During ASYNC I/O Read*

Revision(s) Affected: 1.0

Details: An XRDY synchronization problem causes incorrect operation on the expansion bus. When using the expansion bus asynchronous I/O interface with XRDY, the last word in a data transfer may not be captured. The XBUS interface will seem to act correctly but the last word will not be stored in memory and the direct memory access (DMA) will not complete. The XBUS and the DMA are stalled after the last access, not allowing any subsequent DMA or XBUS transfers to complete. This problem has not been reported internally or externally to date but it could affect the XBUS operation if the recommendations in the workaround are not followed.

Workaround: Do not use XRDY to stall the expansion bus. Instead lengthen the access parameters (setup, strobe, hold) to the maximum and ensure that the external device is ready in that time frame or only make XBUS requests after the external device is ready.

4 Documentation Support

For device-specific datasheets and related documentation, visit the TI web site at: **http://www.ti.com**

To access documentation on the web site:

1. Go to **http://www.ti.com**
2. Open the "**Products & Services**" dialog box and choose "**DSP**"
3. Scroll to the "**TMS320C6000™**" and click on "**Product List**"
4. Click on a device name and then click on the documentation type you prefer.

TMS320C6000, TMS320C6x, and TMS320 are trademarks of Texas Instruments.

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 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.

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.

Mailing Address:

Texas Instruments
Post Office Box 655303
Dallas, Texas 75265