Application Report
SPNA092–November 2005

TMS470R1x DMA Restart Proceedure

Zhaohong Zhang *AEC Automotive*

ABSTRACT

In many TMS470R1x applications, direct memory access (DMA) is used to move data between the system memory and peripherals to communicate with external devices. When the external device detects a data error, a request (normally, an interrupt) will be sent to the TMS470 to restart the unfinished DMA block transfer. This application note presents a generic software approach to restarting the DMA channel that was in use from the beginning when the ongoing DMA block transfer is not completed.

Contents

1 Introduction

In many TMS470 applications, DMA is used to move data between the system memory and peripherals for TMS470 to communicate to external devices. Very often, the DMA transfer is triggered by the external devices. A data error could be detected by the external device in the middle of the DMA block transfer even though no error is detected within TMS470. When this happens, the external device will send a request (normally an interrupt request signal) to the TMS470 to restart the unfinished DMA block transfer.

In TMS470R1X devices, the DMA channel in use can only be restarted when the ongoing DMA block transfer is completed. This application note presents a software approach to force the unfinished DMA block transfer to end quickly so that the DMA channel can be restarted.

2 Details of the Software Approach

The software approach to restart the DMA channel in use when the ongoing DMA block transfer is not completed consists of the following steps:

1. Disable the TMS470 peripheral in use. In this way, any further access to this peripheral will not cause false communication between TMS470 and external device.
2. Disable the hardware request bit in the DMACC0 or DMACC1 register for this particular DMA channel. After this, the unfinished portion of the ongoing DMA block transfer will behave as a software-triggered block transfer and it will go to the end quickly without any external hardware request.
3. Wait until the transfer count goes to zero (block transfer is complete).
4. Re-enable the TMS470 peripheral in use.
5. Configure the DMA channel and restart the transfer.

Steps 1 and 4 are peripheral-dependent and may not be needed for some peripherals. Steps 2 and 3 can be implemented in two lines of C-code as follows.

```
*(volatile ULONG *) 0xFFFFFE80 &= ~0x00000002;   //disable h/w request
 while((*(volatile ULONG *) 0xFFFFF80c));       // wait till the transfer count go to zero
```

In the example shown under Step 5, DMA channel 0 is controlled by DMA control packet zero.

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.

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