TMS320 DSP
DESIGNER'S NOTEBOOK

Monitoring TMS320C240 Peripheral Registers in the Debugger Software

APPLICATION BRIEF: SPRA276

Jeff Crankshaw

Digital Signal Processor Solutions
May 1997

TEXAS INSTRUMENTS

IMPORTANT NOTICE

Texas Instruments (TI) reserves the right to make changes to its products or to discontinue any semiconductor product or service without notice, and advises its customers to obtain the latest version of relevant information to verify, before placing orders, that the information being relied on is current.

TI warrants performance of its semiconductor products and related software to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are utilized to the extent TI deems necessary to support this warranty. Specific testing of all parameters of each device is not necessarily performed, except those mandated by government requirements.

Certain application using semiconductor products may involve potential risks of death, personal injury, or severe property or environmental damage ("Critical Applications").

TI SEMICONDUCTOR PRODUCTS ARE NOT DESIGNED, INTENDED, AUTHORIZED, OR WARRANTED TO BE SUITABLE FOR USE IN LIFE-SUPPORT APPLICATIONS, DEVICES OR SYSTEMS OR OTHER CRITICAL APPLICATIONS.

Inclusion of TI products in such applications is understood to be fully at the risk of the customer. Use of TI products in such applications requires the written approval of an appropriate TI officer. Questions concerning potential risk applications should be directed to TI through a local SC sales office.

In order to minimize risks associated with the customer's applications, adequate design and operating safeguards should be provided by the customer to minimize inherent or procedural hazards.

TI assumes no liability for applications assistance, customer product design, software performance, or infringement of patents or services described herein. Nor does TI 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 of TI covering or relating to any combination, machine, or process in which such semiconductor products or services might be or are used.

TRADEMARKS

TI is a trademark of Texas Instruments Incorporated.

Other brands and names are the property of their respective owners.

CONTACT INFORMATION

US TMS320 HOTLINE	(281) 274-2320
US TMS320 FAX	(281) 274-2324
US TMS320 BBS	(281) 274-2323
US TMS320 email	dsph@ti.com

Contents

Abstract 7
Design Problem 8
Solution 8

Examples

Example 1. INIT File 8

Monitoring TMS320C240 Peripheral Registers in the Debugger Software

Abstract

This document discusses how to view the peripheral registers on a ‘C240 or ‘F240 device while running the emulator. It includes a lengthy code example.

Design Problem

How do I view the peripheral registers on a ‘C240 or ‘F240 device while I am running the emulator?

Solution

The ‘C24x DSP controllers are designed around the ‘C2xLP core processor and utilize the same software development tools as the other devices in the ‘C2xx generation. However, the ‘C24x devices integrate many peripherals that are unique to the DSP family, such the Event Manager and the dual 10-bit ADCs. Also, all of the peripheral registers are mapped into data space.

The ‘C2xx C Source debugger software provides the capability to customize the debugger display. By using the wa command, you may display any data memory location and give it a meaningful name. Typing the following on the command line of the debugger will bring up the Watch window with the contents of data memory location 701Ah and label that data as SYSSR:

```
wa *0x0701A, SYSSR
```

This is the location of the System Status register for ‘C240 and ‘F240 devices.

Details of the *wa* command and Watch window may be found in Chapter 8 and Chapter 13 of the TMS320C2xx C Source Debugger User’s Guide (SPRU151).

A command file customized for the ‘C240 that includes memory map definitions and *wa* commands for all registers is included below. This file is also available on the TI Web site and the TMS320 BBS.

Example 1. INIT File

```
echo C240init.CMD for C240
mr
;DATA MEMORY
ma 0x00000,1,0x005F,ram  ;MMRs
ma 0x00060,1,0x0020,ram  ;On-Chip RAM B2
ma 0x00100,1,0x0100,ram  ;On-Chip RAM B0 if CNF=0
ma 0x00300,1,0x0100,ram  ;On-Chip RAM B1
ma 0x07010,1,0x0010,ioport  ;Peripheral – System Config & Control
ma 0x07020,1,0x0010,ioport  ;Peripheral – WDT / RTI
ma 0x07030,1,0x0010,ioport  ;Peripheral - ADC
ma 0x07040,1,0x0010,ioport  ;Peripheral - SPI
ma 0x07050,1,0x0010,ioport  ;Peripheral - SCI
ma 0x07070,1,0x0010,ioport  ;Peripheral – Ext Ints
ma 0x07090,1,0x0010,ioport  ;Peripheral – Digital I/O
ma 0x07400,1,0x000D,ioport  ;Peripheral – Event Mgr GPT
```

```
ma 0x07411,1,0x000C,ioport  ;Peripheral - Event Mgr CMP,PWM
ma 0x07420,1,0x0007,ioport  ;Peripheral - Event Mgr CAP,QEP
ma 0x0742C,1,0x0009,ioport  ;Peripheral - Event Mgr Int Cntl

;PROGRAM MEMORY
ma 0x00000,0,0x03FFF ,rom  ;Internal Prog mem (Flash or ROM)
                           ;Available if MPNMC=0.
ma 0x0fe00,0,0x100, ram    ;Available if CNF=1 i.e. B0

;System Module Registers
;~~~~~~~~~~~~~~~~~~~~~~
;wa *0x07018, SYSCR     ;System Module Control Register
;wa *0x0701A, SYSSR     ;System Module Status Register
;wa *0x0701E, SYSIVR    ;System Interrupt Vector Register

;Interrupt Registers
;~~~~~~~~~~~~~~~~~~
;wa *0x07070, XINT1_CR  ;Int1 (type A) Control reg
;wa *0x07072, NMI_CR    ;Non maskable Int (type A) Control reg
;wa *0x07078, XINT2_CR  ;Int2 (type C) Control reg
;wa *0x0707A, XINT3_CR  ;Int3 (type C) Control reg
;wa *0x0742C, PDPINT_CR ;Power Drive Protection Int cntl reg

;Digital I/O
;~~~~~~~~~~
;wa *0x07090, OPCRA     ;Output Control Reg A
;wa *0x07092, OPCRB     ;Output Control Reg B
;wa *0x07094, IPSRA     ;Input Status Reg A
;wa *0x07096, IPSRB     ;Input Status Reg B
;wa *0x07098, IOPA_DDR  ;I/O port A Data & Direction reg.
;wa *0x0709A, IOPB_DDR  ;I/O port B Data & Direction reg.
;wa *0x0709C, IOPC_DDR  ;I/O port C Data & Direction reg.
;wa *0x0709E, IOPD_DDR  ;I/O port D Data & Direction reg.

;WatchDog(WD)/Real Time Int(RTI)/Phase Locked Loop(PLL) Registers
;~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
;wa *0x07021, RTI_CNT   ;RTI Counter reg
;wa *0x07023, WD_CNT    ;WD Counter reg
;wa *0x07025, WD_KEY    ;WD Key reg
;wa *0x07027, RTI_CR    ;RTI Control reg
;wa *0x07029, WD_CR     ;WD Control reg
;wa *0x0702B, PLL_CR1   ;PLL control reg 1
;wa *0x0702D, PLL_CR2   ;PLL control reg 2

;Analog-to-Digital Converter(ADC) registers
;~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
;wa *0x07031, ADC_CR_SR     ;ADC Control & Status reg
;wa *0x07033, ADC_CFG       ;ADC Configuration reg
;wa *0x07035, ADC_CH_SEL    ;ADC Channel Select reg
;wa *0x07037, ADC_CH0_DATA  ;ADC Channel 0 Result Data
;wa *0x07039, ADC_CH1_DATA  ;ADC Channel 1 Result Data
```

```
;Serial Peripheral Interface (SPI) Registers
;~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
;wa *0x07040, SPI_CR1       ;SPI Config Control Reg 1
;wa *0x07041, SPI_CR2       ;SPI Operation Control Reg 2
;wa *0x07042, SPI_SR        ;SPI Status Reg
;wa *0x07044, SPI_BAUD      ;SPI Baud rate control reg
;wa *0x07046, SPI_EMU       ;SPI Emulation buffer reg
;wa *0x07047, SPI_BUF       ;SPI Serial Input buffer reg
;wa *0x07049, SPI_DAT       ;SPI Serial Data reg
;wa *0x0704D, SPI_PORT_CR1  ;SPI Port control reg1
;wa *0x0704E, SPI_PORT_CR2  ;SPI Port control reg2
;wa *0x0704F, SPI_PRI_CR    ;SPI Priority control reg

;Serial Communications Interface (SCI) Registers
;~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
;wa *0x07050, SCI_COM_CR    ;SCI Comms Control Reg
;wa *0x07051, SCI_CR1       ;SCI Control Reg 1
;wa *0x07052, SCI_HBAUD     ;SCI Baud rate control
;wa *0x07053, SCI_LBAUD     ;SCI Baud rate control
;wa *0x07054, SCI_CR2       ;SCI Control Reg 2
;wa *0x07055, SCI_RX_STAT   ;SCI Receive status reg
;wa *0x07056, SCI_RX_EMU    ;SCI EMU data buffer
;wa *0x07057, SCI_RX_BUF    ;SCI Receive data buffer
;wa *0x07059, SCI_TX_BUF    ;SCI Transmit data buffer
;wa *0x0705D, SCI_PORT_CR1  ;SCI Port control reg1
;wa *0x0705E, SCI_PORT_CR2  ;SCI Port control reg2
;wa *0x0705F, SCI_PRI_CR    ;SCI Priority control reg

;Event Manager (EV)
;~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
;wa *0x07400, GPTCON    ;General Timer Controls
;wa *0x07401, T1CNT     ;T1 Counter Register
;wa *0x07402, T1CMP     ;T1 Compare Register
;wa *0x07403, T1PER     ;T1 Period Register
;wa *0x07404, T1CON     ;T1 Control Register
;wa *0x07405, T2CNT     ;T2 Counter Register
;wa *0x07406, T2CMP     ;T2 Compare Register
;wa *0x07407, T2PER     ;T2 Period Register
;wa *0x07408, T2CON     ;T2 Control Register
;wa *0x07409, T3CNT     ;T3 Counter Register
;wa *0x0740a, T3CMP     ;T3 Compare Register
;wa *0x0740b, T3PER     ;T3 Period Register
;wa *0x0740c, T3CON     ;T3 Control Register
;wa *0x07411, COMCON    ;Compare Unit Control Register
;wa *0x07413, ACTR      ;Full Compare Output Action Ctl. Reg.
;wa *0x07414, SACTR     ;Simple Compare Output Action Ctl. Reg.

;wa *0x07415, DBTCON    ;Dead Band Timer Control
;wa *0x07417, CMPR1     ;Full Compare Channel 1 Threshold
;wa *0x07418, CMPR2     ;Full Compare Channel 2 Threshold
```

```
;wa *0x07419, CMPR3     ;Full Compare Channel 3 Threshold
;wa *0x0741a, SCMPR1    ;Simple Comp Channel 1 Threshold
;wa *0x0741b, SCMPR2    ;Simple Comp Channel 2 Threshold
;wa *0x0741c, SCMPR3    ;Simple Comp Channel 3 Threshold
;wa *0x07420, CAPCON    ;Capture Unit Control
;wa *0x07422, CAPFIFO   ;FIFO1-4 Status Register
;wa *0x07423, FIFO1     ;Capture Channel 1 FIFO Top
;wa *0x07424, FIFO2     ;Capture Channel 2 FIFO Top
;wa *0x07425, FIFO3     ;Capture Channel 3 FIFO Top
;wa *0x07426, FIFO4     ;Capture Channel 4 FIFO Top
;wa *0x0742c, IMRA      ;Group A Interrupt Mask Register
;wa *0x0742d, IMRB      ;Group B Interrupt Mask Register
;wa *0x0742e, IMRC      ;Group C Interrupt Mask Register
;wa *0x0742f, IFRA      ;Group A Interrupt Flag Register
;wa *0x07430, IFRB      ;Group B Interrupt Flag Register
;wa *0x07431, IFRC      ;Group C Interrupt Flag Register
;wa *0x07432, IVRA      ;Group A Int. Vector Offset Register
;wa *0x07433, IVRB      ;Group B Int. Vector Offset Register
;wa *0x07434, IVRC      ;Group C Int. Vector Offset Register

echo   C240init.CMD HAS BEEN LOADED
```