Application Report
SPRA507

Sequential Addressing of I/O Ports on the TMS320C54x DSP

Clay Turner *Digital Signal Processoring Solutions*

Abstract

On the Texas Instruments (TI™) TMS320C54x, the I/O port addresses are hard-coded in the opcodes for the PORTW (I/O port write) and PORTR (I/O port read) instructions. This document discusses how the tables of data can be transferred using sequential I/O port addresses.

The fact that the I/O port addresses in the PORTR and PORTW instruction are hard-coded makes reading or writing a table of data to sequential addresses in I/O space difficult. The simplest method would be to have a sequence of PORTx instructions with each address specified explicitly. Although this method will work, it consumes as many instructions as there are entries in the table (at 2 program memory words per PORTx instruction). An alternative method takes advantage of the ability to overlay the on-chip DARAM on the C54x. A code listing is included.

Contents

Examples

Design Problem

On the TMS320C54x, the I/O port addresses are hard-coded in the opcodes for the PORTW (I/O port write) and PORTR (I/O port read) instructions. How can tables of data be transferred using sequential I/O port addresses?

Solution

The fact that the I/O port addresses in the PORTR and PORTW instruction are hard-coded makes reading or writing a table of data to sequential addresses in I/O space difficult. The simplest method is to have a sequence of PORTx instructions with each address specified explicitly. Although this method will work, it consumes as many instructions as there are entries in the table (at 2 program memory words per PORTx instruction). The alternative method shown in Example 1 takes advantage of the ability to overlay the on-chip DARAM on the C54x.

When the on-chip DARAM block is “overlaid” (due to the OVLY bit being set to 1), this block of memory is accessible from both program space and data space and mapped to the same addresses in both memory spaces. So a given memory address in program memory is physically the same location as the address in data memory. Consequently, an access to data memory can be used to modify program memory. This capability is utilized to dynamically change the address coded in the PORTx instruction so that on each pass of a loop, the address can be different. An example of this implementation is shown below in Example 1.

Example 1. Sequential Addressing of a PORTW Instruction

```
start:
            ORM   #00020h,pmst    ;set OVLY=1
            STM   #01000h,ar2     ;pointer to data memory address
            STM   #02000h,ar3     ;pointer for I/O port address
            MVMD  ar3,(portloc+1) ;update PORTW instruction
                                  ; with new address
            STM   #table_length,BRC  ;initialize BRC
            RPTB  end_block-1
portloc:    PORTW *ar2+,0h        ;copy word from data space to
                                  ; I/O space and increment
                                  ; data memory address
            MAR   *ar3+           ; increment I/O memory address
            MVMD  ar3,(portloc+1) ;update PORTW instruction
                                  ;with new address
            NOP                   ;wait for MVMD pipeline latency
            NOP                   ;wait for MVMD pipeline latency
                                  ;(portloc+1) is now updated
end_block:
```

In this example, AR2 and AR3 are used as pointers to the tables in data space and I/O space, respectively. The source table is located at address 01000h in data space. The destination table is located at address 02000h in I/O space. The MVMD instruction is used to modify the port address in the PORTW instruction. The port address is the second word of the instruction and is indicated as one address higher than the location of the PORTW instruction by using the label (portloc+1).

On each pass of the loop, data is copied from data memory address (pointed by AR2) to the I/O port address that is currently loaded in the PORTW opcode. The MAR instruction increments AR3, which keeps track of the desired I/O port address. The MVMD copies that new address into the second word of the PORTW instruction. The result is a block of locations in data memory being copied to a block of locations in I/O space.

A two-cycle latency between the MVMD instruction modifies the port address and the PORTx instruction. This occurs because the MVMD instruction writes the change to (portloc+1) in the execute phase of the pipeline, but the PORTx instruction will read the port address during the second cycle of the fetch phase of the pipeline. At least two cycles must exist between the MVMD instruction and the PORTx instruction that follows to make sure that the address has been modified before it is fetched.

The two NOP instructions in the example serve this latency, but two useful one-cycle instructions or a single two-cycle instruction could replace them. The MAR instruction could even replace one of the NOP instructions if the starting address was corrected accordingly. The port address indicated in the PORTx instruction (0h in this example) is irrelevant because it gets modified anyway. An approach similar to Example 1 can be used with the PORTR instruction. The operands of the PORTR instruction will simply be reversed.

For this implementation, each pass of the loop will require a minimum of 7 cycles to execute (assuming external memory wait states are minimized). For this code to run successfully, the OVLY bit must be set and this code must be stored in on-chip DARAM. The addresses of the data and I/O tables are not limited.

INTERNET

www.ti.com

Register with TI&ME to build custom information pages and receive new product updates automatically via email.

TI Semiconductor Home Page
http://www.ti.com/sc

TI Distributors
http://www.ti.com/sc/docs/distmenu.htm

PRODUCT INFORMATION CENTERS

US TMS320

Hotline	(281) 274-2320
Fax	(281) 274-2324
BBS	(281) 274-2323
email	dsph@ti.com

Americas

Phone	+1(972) 644-5580
Fax	+1(972) 480-7800
Email	sc-infomaster@ti.com

Europe, Middle East, and Africa

Phone	
Deutsch	+49-(0) 8161 80 3311
English	+44-(0) 1604 66 3399
Francais	+33-(0) 1-30 70 11 64
Italiano	+33-(0) 1-30 70 11 67
Fax	+33-(0) 1-30-70 10 32
Email	epic@ti.com

Japan

Phone	
International	+81-3-3457-0972
Domestic	+0120-81-0026
Fax	
International	+81-3-3457-1259
Domestic	+0120-81-0036
Email	pic-japan@ti.com

Asia

Phone	
International	+886-2-3786800
Domestic	
Australia	1-800-881-011

Asia (continued)

TI Number	-800-800-1450
China	10811
TI Number	-800-800-1450
Hong Kong	800-96-1111
TI Number	-800-800-1450
India	000-117
TI Number	-800-800-1450
Indonesia	001-801-10
TI Number	-800-800-1450
Korea	080-551-2804
Malaysia	1-800-800-011
TI Number	-800-800-1450
New Zealand	+000-911
TI Number	-800-800-1450
Philippines	105-11
TI Number	-800-800-1450
Singapore	800-0111-111
TI Number	-800-800-1450
Taiwan	080-006800
Thailand	0019-991-1111
TI Number	-800-800-1450

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 and complete. 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.