INTRODUCTION

This application note discusses how to retrieve data from program memory to data memory and write data from data memory to program memory.

RETLW K Instruction

As in all PIC17CXXX family parts, the simplest method used to retrieve data from program memory to data memory is to use the RETLW K instruction. For example:

```assembly
; simple program to transfer table values to PortB
Main
    movlw 5,W ;load offset
    call SimpleTableRead
    movwf PortB ;output to PortB

SimpleTableRead
    addwf PC ;add offset to PC
    retlw 0 ;return a known table value based on the OFFSET.
```

In the example above, OFFSET is loaded with the required offset to the Table and the subroutine SimpleTableRead is called. The table value is returned in the W register. In this manner program memory can be transferred to data memory.

Table Read Instruction

The PIC17C42 has an expanded instruction set which includes the TABLRD and TLRD instructions. These instructions are specifically constructed to transfer data from program memory to data memory.

If the instruction syntax is: TABLRD t,i,f, the sequence in which this instruction is executed is as follows:

a) if t = 1 then the high byte of the table latch (TBLATH) is loaded in the file register f.
b) else (if t = 0) the low byte of the table latch (TBLATL) is loaded in the file register f.
c) next, the 16 bit data pointed to by the table pointer (TBLPTR) is loaded into the table latch.
d) lastly, if i = 1 the table pointer (TBLPTR) is incremented.

Note: The first time this instruction is executed in a sequence, the table latch will not be initialized, hence an unknown value will be loaded in the file register. This is not a problem if the user overwrites the same f register in the next subsequent instruction.

If the instruction syntax is: TLRD t,f, the sequence in which this instruction is executed is as follows:

a) if t = 1 then the high byte of the table latch (TBLATH) is loaded in the file register f.
b) else (if t = 0) the low byte of the table latch (TBLATL) is loaded in the file register f.

FIGURE 1: TABLE READ
Read In-Line

A simple method of transferring data from program memory to data memory is to use the `TABLRD` and `TLRD` instructions in sequence as shown in the example below:

;transfer 6 bytes of data from program memory at 0x500, to data memory at 0x80

ReadInLine

```
movlw   05        ;load table pointer
; with 0x500
movwf   TBLPTRH   ;      /
clrf    TBLPTRL   ;      /
tablrd  0,1,0x80  ;get 16 bit value in table latch.
tablrd  1,1,0x81  ;high byte (2nd) @ 81
tlrd  0,0x82     ;3rd byte @ 82
tablrd  1,1,0x83  ;4th byte @ 83
tlrd  0,0x84     ;5th byte @ 84
tablrd  1,1,0x85  ;6th byte @ 85
```

Reading a Block of Data

In instances where a block of N bytes needs to be transferred from program memory to data memory, the `TABLRD` and `TLRD` instruction need to be included in a loop which checks for N transfers.

;transfer 'COUNT' bytes (even values only) of data at program memory 'MESSAGE' to data memory at: 'RAM_BUFFER'

ReadBlock

```
movlw   high MESSAGE  ;load table pointer
movpf   W,TBLPTRH     ;      /
movlw   low MESSAGE   ;      /
movpf   W,TBLPTRL     ;      /
bcf     ALUSTA,5      ;enable post auto increment of FSR0
movlw   RAM_BUFFER    ;initialize FSR0 to RAM_BUFFER
movfp   W,FSR0        ;initialize COUNT to FSR0
tablrd  1,1,RAM_BUFFER ;initialize table latch
ReadBlockLoop
	tlrd    1,0x00        ;do indirect read of high byte
	tablrd  0,1,0x00      ;do indirect read of low byte
	decfsz  W              ;check if count=0

go to   ReadBlockLoop  ;no then do next
return                           ;else end of transfer.
```

<table>
<thead>
<tr>
<th>Program</th>
<th>Code Size</th>
<th>Transfer Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Simple Table Read (using <code>RETLW</code>)</td>
<td>N + 3</td>
<td>6 cycles/byte</td>
</tr>
<tr>
<td>Read In-Line</td>
<td>4 + N + N/2</td>
<td>1.5 cycles/byte</td>
</tr>
<tr>
<td>Read Block (using loop)</td>
<td>14 + N/2</td>
<td>3 cycles/byte</td>
</tr>
</tbody>
</table>

N = Number of bytes to transfer

**Code Size**

<table>
<thead>
<tr>
<th>Code Size</th>
<th>Simple Table Read</th>
<th>Read In-Line</th>
<th>Read Block</th>
</tr>
</thead>
<tbody>
<tr>
<td>N = 10</td>
<td>13</td>
<td>19</td>
<td>19</td>
</tr>
<tr>
<td>N = 20</td>
<td>23</td>
<td>34</td>
<td>24</td>
</tr>
</tbody>
</table>

Concluding:

In cases where the number of bytes to be transferred is small, the Read In-Line offers small code size for fast transfer rate. However, as the number of bytes to be transferred increases, the Read Block offers optimum code size for a decent transfer rate.
Table Write Instruction

The PIC17C42 has a TABLWT and a TLWT instruction which transfer data from data memory to program memory. Note in cases where the table pointer points to internal EPROM, the table write instruction will try to program the EPROM, hence the programming voltage must be present on the VPP line to successfully program the part.

The instruction syntax is: TABLWT t, i, f.

The sequence in which this instruction is executed is as follows:

a) if t = 1 then the file register f is loaded to the high byte of the table latch (TBLATH).

b) else (if t = 0) the file register f is loaded to the low byte of the table latch (TBLATL).

c) next, the 16-bit data in the table latch is transferred to the program memory pointed to by the table pointer (TBLPTR).

d) lastly, if i = 1 the table pointer (TBLPTR) is incremented.

The instruction syntax is: TLWT t, f

The sequence in which this instruction is executed is as follows:

a) if t = 1 then the file register f is loaded to the high byte of the table latch (TBLATH).

b) else (if t = 0) the file register f is loaded to the low byte of the table latch (TBLATL).

Write In-Line

A simple method of transferring data from data memory to program memory is to use the TABLWT and TLWT instruction in sequence as shown in the example below:

; transfer 6 bytes of data in data memory at 0x80 to program memory at 0x5000:

ReadInLine

movlw 50 ;load table pointer
movwf TBLPTRH ;with 0x5000
clr TBLPTRL ;
tlwt 1,0x80 ;high byte @ table latch.
tablwt 0,1,0x81 ;low byte @ table latch
            ;latch @ prog. mem.
tlwt 1,0x82 ;3rd and 4th byte
            ;@ prog. mem.
tablwt 0,1,0x83 ;
tlwt 1,0x84 ;5th and 6th byte
            ;@ prog. mem.
tablwt 0,1,0x85 ;

Writing a Block of Data

In instances where a block of N bytes needs to be transferred from data memory to program memory, the TABLWT and TLWT instructions need to be included in a loop which checks for N transfers.

; transfer 'COUNT' bytes (even values only) of data at program memory at 'RAM_BUFFER' to program memory at 'MESSAGE'

WriteBlock

movlw high MESSAGE ;load table pointer
movpf W,TBLPTRH ;
movlw low MESSAGE ;
movpf W,TBLPTRL ;
bcf ALUSTA,5 ;enable post auto
            ;increment of FSR0
movlw RAM_BUFFER ;initialize FSR0
            ;to RAM_BUFFER
movfp W,FSR0 ;
movlw COUNT/2 ;initialize count
WriteBlockLoop

      tlwt 1,0x00 ;high byte
      ;@ table latch
tablwt 0,1,0x00 ;low byte @ table
            ;latch;
            ;table latch
            ;@ prog. mem.

      decfsz W ;check if count = 0
goto WriteBlockLoop ;no then do next
return ;else end of transfer

FIGURE 2: TABLE WRITE

![Diagram of Table Write Instruction](image-url)
Note the following details of the code protection feature on PICmicro® MCUs.

- The PICmicro family meets the specifications contained in the Microchip Data Sheet.
- Microchip believes that its family of PICmicro microcontrollers is one of the most secure products of its kind on the market today, when used in the intended manner and under normal conditions.
- There are dishonest and possibly illegal methods used to breach the code protection feature. All of these methods, to our knowledge, require using the PICmicro microcontroller in a manner outside the operating specifications contained in the data sheet. The person doing so may be engaged in theft of intellectual property.
- Microchip is willing to work with the customer who is concerned about the integrity of their code.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of their code. Code protection does not mean that we are guaranteeing the product as "unbreakable".
- Code protection is constantly evolving. We at Microchip are committed to continuously improving the code protection features of our product.

If you have any further questions about this matter, please contact the local sales office nearest to you.

Information contained in this publication regarding device applications and the like is intended through suggestion only and may be superseded by updates. It is your responsibility to ensure that your application meets with your specifications. No representation or warranty is given and no liability is assumed by Microchip Technology Incorporated with respect to the accuracy or use of such information, or infringement of patents or other intellectual property rights arising from such use or otherwise. Use of Microchip's products as critical components in life support systems is not authorized except with express written approval by Microchip. No licenses are conveyed, implicitly or otherwise, under any intellectual property rights.

© 2002 Microchip Technology Inc.
<table>
<thead>
<tr>
<th>Country</th>
<th>Address</th>
<th>Phone</th>
<th>Fax</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AMERICAS</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Corporate Office</td>
<td>2355 West Chandler Blvd. Chandler, AZ 85224-6199</td>
<td>480-792-7200</td>
<td>480-792-7277</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td><strong>Web Address:</strong> <a href="http://www.microchip.com">http://www.microchip.com</a></td>
</tr>
<tr>
<td>Rocky Mountain</td>
<td>2355 West Chandler Blvd. Chandler, AZ 85224-6199</td>
<td>480-792-7966</td>
<td>480-792-7456</td>
</tr>
<tr>
<td>Atlanta</td>
<td>500 Sugar Mill Road, Suite 2008 Atlanta, GA 30350</td>
<td>770-640-0034</td>
<td>770-640-0037</td>
</tr>
<tr>
<td>Boston</td>
<td>2 Lan Drive, Suite 120 Westford, MA 01886</td>
<td>978-692-3848</td>
<td>978-692-3821</td>
</tr>
<tr>
<td>Chicago</td>
<td>333 Pierce Road, Suite 180 Itasca, IL 60143</td>
<td>630-285-0071</td>
<td>630-285-0075</td>
</tr>
<tr>
<td>Dallas</td>
<td>4570 Westgrove Drive, Suite 160 Addison, TX 75001</td>
<td>972-818-7943</td>
<td>972-818-2924</td>
</tr>
<tr>
<td>Kokomo</td>
<td>2767 S. Albright Road Kokomo, Indiana 46902</td>
<td>765-864-8360</td>
<td>765-864-8387</td>
</tr>
<tr>
<td>Los Angeles</td>
<td>1201 Von Karman, Suite 1090 Irvine, CA 92612</td>
<td>949-263-1888</td>
<td>949-263-1338</td>
</tr>
<tr>
<td>San Jose</td>
<td>Microchip Technology Inc. 2107 North First Street, Suite 590 San Jose, CA 95131</td>
<td>408-436-7950</td>
<td>408-436-7955</td>
</tr>
<tr>
<td>Toronto</td>
<td>6285 Northame Drive, Suite 108 Mississauga, Ontario L4V 1X5, Canada</td>
<td>905-673-0699</td>
<td>905-673-6509</td>
</tr>
<tr>
<td><strong>ASIA/PACIFIC</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Australia</td>
<td>Microchip Technology Australia Pty Ltd Suite 22, 41 Rawson Street Epping 2121, NSW Australia</td>
<td>61-2-9868-6733</td>
<td>61-2-9868-6755</td>
</tr>
<tr>
<td>China - Beijing</td>
<td>Microchip Technology Consulting (Shanghai) Co., Ltd., Beijing Liaison Office Unit 915 Bei Hai Wan Tai Bldg. No. 6 Chaoyangmen Beidajie Beijing, 100027, No. China</td>
<td>86-10-85282100</td>
<td>86-10-85282104</td>
</tr>
<tr>
<td>China - Chengdu</td>
<td>Microchip Technology Consulting (Shanghai) Co., Ltd., Chengdu Liaison Office Rm. 2401, 24th Floor, Ming Xing Financial Tower No. 88 TIDU Street Chengdu 610016, China</td>
<td>86-28-67662000</td>
<td>86-28-6766599</td>
</tr>
<tr>
<td>China - Fuzhou</td>
<td>Microchip Technology Consulting (Shanghai) Co., Ltd., Fuzhou Liaison Office Unit 26F, World Trade Plaza No. 71 Wusi Road Fuzhou 350001, China</td>
<td>86-591-79350500</td>
<td>86-591-7935020</td>
</tr>
<tr>
<td>China - Shanghai</td>
<td>Microchip Technology Consulting (Shanghai) Co., Ltd., Room 701, Bldg. B Far East International Plaza No. 317 Xian Xian Road Shanghai, 2000051</td>
<td>86-21-6275-5700</td>
<td>86-21-6275-5060</td>
</tr>
<tr>
<td>China - Shenzhen</td>
<td>Microchip Technology Consulting (Shanghai) Co., Ltd., Shenzhen Liaison Office Rm. 1315, 13/F, Shenzhen Kerry Centre, Renminnu Lu Shenzhen 518001, China</td>
<td>86-755-2350361</td>
<td>86-755-2366086</td>
</tr>
<tr>
<td>Hong Kong</td>
<td>Microchip Technology Hong Kong Ltd. Unit 901-6, Tower 2, Metroplaza 223 Hong Fong Road Kwai Fong, N.T., Hong Kong</td>
<td>852-2401-1200</td>
<td>852-2401-3431</td>
</tr>
<tr>
<td>India</td>
<td>Microchip Technology Inc. India Liaison Office Divyasree Chambers 1 Floor, Wing A (A3/A4) No. 11, O’Shaugnessy Road Bangalore, 560 025, India</td>
<td>91-80-2290061</td>
<td>91-80-2290062</td>
</tr>
<tr>
<td>Korea</td>
<td>Microchip Technology Korea 168-1, Youngbo Bldg. 3 Floor Samsung-Dong, Kangnam-Ku Seoul, Korea 135-882</td>
<td>82-2-554-7200</td>
<td>82-2-558-5934</td>
</tr>
<tr>
<td>Singapore</td>
<td>Microchip Technology Singapore Pte Ltd. 200 Middle Road #07-02 Prime Centre Singapore, 188980</td>
<td>65-334-8870</td>
<td>65-334-8850</td>
</tr>
<tr>
<td>Taiwan</td>
<td>Microchip Technology Taiwan 11F-3, No. 207 Tung Hua North Road Taipei, 105, Taiwan</td>
<td>886-2-2717-7175</td>
<td>886-2-2545-0139</td>
</tr>
<tr>
<td><strong>EUROPE</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Denmark</td>
<td>Microchip Technology Nordic ApS Regus Business Centre Lautrup høj 1-3 Ballerup DK-2750 Denmark</td>
<td>45 4420 9859</td>
<td>45 4420 9910</td>
</tr>
<tr>
<td>France</td>
<td>Microchip Technology SARL Parc d’Activite du Moulin de Massy 43 Rue du Saule Trapu Batiment A – 1er Etage 91300 Massy, France</td>
<td>33-1-69-53-63-20</td>
<td>33-1-69-30-90-79</td>
</tr>
<tr>
<td>Germany</td>
<td>Microchip Technology GmbH Gustav-Heinemann Ring 125 D-81739 Munich, Germany</td>
<td>49-89-6275-5000</td>
<td>49-89-6275-5060</td>
</tr>
<tr>
<td>Italy</td>
<td>Microchip Technology SRL Centro Direzionale Colleoni Palazzo Taurus 1 V. Le Colleoni 1 20041 Agrate Brianza Milan, Italy</td>
<td>39-039-65791-1</td>
<td>39-039-6899883</td>
</tr>
<tr>
<td>United Kingdom</td>
<td>Arizona Microchip Technology Ltd. 505 Eskdale Road Winnersh Triangle Wokingham Berkshire, England RG41 5TU</td>
<td>44 118 921 5869</td>
<td>44 118 921 5820</td>
</tr>
</tbody>
</table>

© 2002 Microchip Technology Inc.