MICROCHIP

AN579

Using the 8-Bit Parallel Slave Port

Author: Scott Fink
Microchip Technology Inc.

INTRODUCTION

PIC16C64/74 microcontrollers from Microchip Technology Inc. can be interfaced with ease into a multi-microprocessor environment using its built-in Parallel Slave Port (PSP). With their very high operating speeds (cycle times as low as 200 ns with a clock rate of 20 MHz), and an array of on-chip peripherals, these microcontrollers make ideal smart interfaces to the real world.

IMPLEMENTATION

PORTD operates as an 8-bit wide Parallel Slave Port, with PORTE providing the control signals. In parallel slave mode, PORTD is asynchronously readable and writable by the external world through the chip select (RE2/$\overline{\text{CS}}$), Read (RE0/$\overline{\text{RD}}$), and Write (RE1/$\overline{\text{WR}}$) control inputs.

In order to use the Parallel Slave Port, the data direction bits in the TRISE register corresponding to $\overline{\text{RD}}$, $\overline{\text{WR}}$, and $\overline{\text{CS}}$ (TRISE<2:0>) must be configured as inputs (set = 1) and control bit PSPMODE (TRISE<4>) must be set.

The port pins are connected to two 8-bit latches, one for data output (from the PIC16CXXX) and one for data input. The PIC16CXXX sends data by writing to the output latch, and receives data by reading the input latch (note that the input and output latches are at the same address). In PSP mode the TRISD register is ignored, since the external device connected to the slave port controls the direction of data flow.

When the external device performs either a read or a write operation to the PIC16CXXX, interrupt flag, PSPIF (PIR1<7>), will be set and the processor interrupted if bit PSPIE (PIE1<7>) is set and interrupts are enabled (enable bits GIE and PEIE, (INTCON<7:6>) set). When the interrupt is serviced, bit PSPIF must be cleared by software.

The read-only status flag bit IBF, Input Buffer Full (TRISE<7>), is set if a received word is waiting to be read. Bit IBF is cleared upon read of the input buffer latch. If another word is received prior to the first being read, status flag bit IBOV (TRISE<5>) is set. Bit IBOV can be cleared by software.

The Output Buffer Full status bit, OBF (TRISE<6>), is set if a word written to PORTD latch is waiting to be read by the external bus.

When not in Parallel Slave Port mode the IBF and OBF bits are cleared. If flag bit IBOV was previously set, however, it must be cleared by software.

Note that the following registers are for a PIC16C74 and not all peripherals are available on the PIC16C64.

TABLE 1: SUMMARY OF PARALLEL SLAVE PORT REGISTERS

Register Name	Function	Address	Power-on Reset Value
PORTD	Parallel slave port Read/Write Data	08h	`xxxx xxxx`
TRISD	PORTD data direction register	88h	`1111 1111`
PORTE	Read/Write/Chip Select signals	09h	`---- -xxx`
TRISE	Control bits for PORTD slave port	89h	`0000 -111`
INTCON	peripheral and global interrupt enable bits	0Bh	`0000 000x`
PIR1	Interrupt register (PSPIF bit)	0Ch	`0000 0000`
PIE1	Interrupt Enable register (PSPIE bit)	8Ch	`0000 0000`

TABLE 2: PORTE FUNCTIONS

Name	Bit#	Buffer Type	Function
RE0/$\overline{\mathbf{RD}}$/AN5	bit0	ST/TTL[(1)]	Input/output port pin or read control input in parallel slave port mode or analog input: $\overline{\text{RD}}$ 1 = Not a read operation 0 = Read operation. Reads PORTD register (if chip selected)
RE1/$\overline{\mathbf{WR}}$/AN6	bit1	ST/TTL[(1)]	Input/output port pin or write control input in parallel slave port mode or analog input: $\overline{\text{WR}}$ 1 = Not a write operation 0 = Write operation. Writes PORTD register (if chip selected)
RE2/$\overline{\mathbf{CS}}$/AN7	bit2	ST/TTL[(1)]	Input/output port pin or chip select control input in parallel slave port mode or analog input: $\overline{\text{CS}}$ 1 = Device is not selected 0 = Device is selected

Legend: ST = Schmitt Trigger input TTL = TTL input

Note 1: Input buffers are Schmitt Triggers when in I/O mode and TTL buffers when in Parallel Slave Port Mode.

FIGURE 1: TRISE REGISTER

R-0	R-0	R/W-0	R/W-0	U-0	R/W-1	R/W-1	R/W-1
IBF	OBF	IBOV	PSPMODE	—	Bit2	Bit1	Bit0
bit7							bit0

R = Readable bit
W = Writable bit
U = Unimplemented bit, read as ‘0’
- n = Value at POR reset

bit 7: **IBF:** Input Buffer Full Status bit
1 = A word has been received and waiting to be read by the CPU
0 = No word has been received

bit 6: **OBF**: Output Buffer Full Status bit
1 = The output buffer still holds a previously written word
0 = The output buffer has been read

bit 5: **IBOV**: Input Buffer Overflow Detect bit (in microprocessor mode)
1 = A write occurred when a previously input word has not been read (must be cleared in software)
0 = No overflow occurred

bit 4: **PSPMODE**: Parallel Slave Port Mode Select bit
1 = Parallel slave port mode
0 = General purpose I/O mode

bit 3: **Unimplemented**: Read as '0'

bit 2: **Bit2**: Direction control bit for pin RE2/$\overline{\text{CS}}$/AN7
1 = Input
0 = Output

bit 1: **Bit1**: Direction control bit for pin RE1/$\overline{\text{WR}}$/AN6
1 = Input
0 = Output

bit 0: **Bit0**: Direction control bit for pin RE0/$\overline{\text{RD}}$/AN5
1 = Input
0 = Output

FIGURE 2: PIE1 REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PSPIE	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE
bit7							bit0

R = Readable bit
W = Writable bit
U = Unimplemented bit,
read as ‘0’
- n = Value at POR reset

bit 7: **PSPIE:** Parallel Slave Port Read/Write Interrupt Enable bit
1 = Enables the PSP read/write interrupt
0 = Disables the PSP read/write interrupt

bit 6: **ADIE**: A/D Converter Interrupt Enable bit
1 = Enables the A/D interrupt
0 = Disables the A/D interrupt

bit 5: **RCIE**: USART Receive Interrupt Enable bit
1 = Enables the USART receive interrupt
0 = Disables the USART receive interrupt

bit 4: **TXIE**: USART Transmit Interrupt Enable bit
1 = Enables the USART transmit interrupt
0 = Disables the USART transmit interrupt

bit 3: **SSPIE**: Synchronous Serial Port Interrupt Enable bit
1 = Enables the SSP interrupt
0 = Disables the SSP interrupt

bit 2: **CCP1IE**: CCP1 Interrupt Enable bit
1 = Enables the CCP1 interrupt
0 = Disables the CCP1 interrupt

bit 1: **TMR2IE**: TMR2 to PR2 Match Interrupt Enable bit
1 = Enables the TMR2 to PR2 match interrupt
0 = Disables the TMR2 to PR2 match interrupt

bit 0: **TMR1IE**: TMR1 Overflow Interrupt Enable bit
1 = Enables the TMR1 overflow interrupt
0 = Disables the TMR1 overflow interrupt

FIGURE 3: PIR1 REGISTER

R/W-0	R/W-0	R-0	R-0	R/W-0	R/W-0	R/W-0	R/W-0
PSPIF(1)	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF
bit7							bit0

R = Readable bit
W = Writable bit
U = Unimplemented bit, read as ‘0’
- n = Value at POR reset

bit 7: **PSPIF(1):** Parallel Slave Port Read/Write Interrupt Flag bit
1 = A read or a write operation has taken place (must be cleared in software)
0 = No read or write has occurred

bit 6: **ADIF**: A/D Converter Interrupt Flag bit
1 = An A/D conversion completed (must be cleared in software)
0 = The A/D conversion is not complete

bit 5: **RCIF**: USART Receive Interrupt Flag bit
1 = The USART receive buffer is full (cleared by reading RCREG)
0 = The USART receive buffer is empty

bit 4: **TXIF**: USART Transmit Interrupt Flag bit
1 = The USART transmit buffer is empty (cleared by writing to TXREG)
0 = The USART transmit buffer is full

bit 3: **SSPIF**: Synchronous Serial Port Interrupt Flag bit
1 = The transmission/reception is complete (must be cleared in software)
0 = Waiting to transmit/receive

bit 2: **CCP1IF**: CCP1 Interrupt Flag bit
Capture Mode
1 = A TMR1 register capture occurred (must be cleared in software)
0 = No TMR1 register capture occurred
Compare Mode
1 = A TMR1 register compare match occurred (must be cleared in software)
0 = No TMR1 register compare match occurred
PWM Mode
Unused in this mode

bit 1: **TMR2IF**: TMR2 to PR2 Match Interrupt Flag bit
1 = TMR2 to PR2 match occurred (must be cleared in software)
0 = No TMR2 to PR2 match occurred

bit 0: **TMR1IF**: TMR1 Overflow Interrupt Flag bit
1 = TMR1 register overflowed (must be cleared in software)
0 = TMR1 register did not overflow

Note 1: PIC16C73/73A/76 devices do not have a Parallel Slave Port implemented, this bit location is reserved on these devices, always maintain this bit clear.

Interrupt flag bits get set when an interrupt condition occurs regardless of the state of its corresponding enable bit or the global enable bit, GIE (INTCON<7>). User software should ensure the appropriate interrupt flag bits are clear prior to enabling an interrupt.

TABLE 3: INTCON REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-x
GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF
bit7							bit0

R = Readable bit
W = Writable bit
U = Unimplemented bit, read as ‘0’
- n = Value at POR reset

bit 7: **GIE:** Global Interrupt Enable bit
1 = Enables all un-masked interrupts
0 = Disables all interrupts

bit 6: **PEIE**: Peripheral Interrupt Enable bit
1 = Enables all un-masked peripheral interrupts
0 = Disables all peripheral interrupts

bit 5: **T0IE**: TMR0 Overflow Interrupt Enable bit
1 = Enables the TMR0 interrupt
0 = Disables the TMR0 interrupt

bit 4: **INTE**: RB0/INT External Interrupt Enable bit
1 = Enables the RB0/INT external interrupt
0 = Disables the RB0/INT external interrupt

bit 3: **RBIE**: RB Port Change Interrupt Enable bit
1 = Enables the RB port change interrupt
0 = Disables the RB port change interrupt

bit 2: **T0IF**: TMR0 Overflow Interrupt Flag bit
1 = TMR0 register has overflowed (must be cleared in software)
0 = TMR0 register did not overflow

bit 1: **INTF**: RB0/INT External Interrupt Flag bit
1 = The RB0/INT external interrupt occurred (must be cleared in software)
0 = The RB0/INT external interrupt did not occur

bit 0: **RBIF**: RB Port Change Interrupt Flag bit
1 = At least one of the RB7:RB4 pins changed state (must be cleared in software)
0 = None of the RB7:RB4 pins have changed state

Please check the Microchip BBS for the latest version of the source code. Microchip’s Worldwide Web Address: www.microchip.com; Bulletin Board Support: MCHIPBBS using CompuServe® (CompuServe membership not required).

APPENDIX A: PIC16C64/74 PARALLEL SLAVE PORT

```
MPASM 01.40 Released            PSP64.ASM   1-16-1997  17:03:44         PAGE  1


LOC  OBJECT CODE     LINE SOURCE TEXT
  VALUE

                    00001 ;*********************************************************************
                    00002 ;* 16C64/74 Parallel Slave port
                    00003 ;*  This program demonstrates the Parallel Slave Port function of
                    00004 ;*  the PIC16C64/74.  The program is interrupt driven, when the PIC
                    00005 ;*  is either read from or written to, an interrupt is generated. If
                    00006 ;*  the interrupt was caused by a read, a register is incremented, and
                    00007 ;*  the new count is placed in an output queue.  If the interrupt was
                    00008 ;*  caused by a write, the data is put on the Port B pins
                    00009 ;
                    00010 ;     Program:          PSP64.ASM
                    00011 ;     Revision Date:
                    00012 ;                       1-15-97      Compatibility with MPASMWIN 1.40
                    00013 ;
                    00014 ;;********************************************************************
                    00015         list p=16c64
                    00016         ERRORLEVEL  -302
                    00017 ;
                    00018         include "p16c64.inc"
                    00001         LIST
                    00002 ; P16C64.INC Standard Header File, Ver. 1.01 Microchip Technology, Inc.
                    00238         LIST
                    00019
                    00020 ;Register definitions
  00000020          00021 FLAGREG equ     20h             ;Flag bit register
  00000021          00022 OUTDATA equ     21h             ;Output data
  00000022          00023 INDATA  equ     22h             ;Input data
  00000023          00024 COUNT   equ     23h             ;Count of times output register read
                    00025
                    00026 ;Bit definitions for flag register
  00000000          00027 err     equ     00h             ;Error flag bit
  00000001          00028 OUTRDY  equ     01h             ;Output data ready flag
  00000002          00029 INFULL  equ     02h             ;Input data received flag
                    00030
0000                00031         org     0000h           ;Reset Vector
0000 2806           00032         goto    Start
                    00033
0005                00034         org     0005h           ;Interrupt Vector
0005 2820           00035         goto    Service_Int
                    00036
0006                00037 Start
0006 01A1           00038         clrf    OUTDATA         ;Clear data registers
0007 01A2           00039         clrf    INDATA
0008 1683           00040         bsf     STATUS,RP0      ;Select register Bank1
0009 3017           00041         movlw   b'00010111'     ;Set RD, WR, and CS as
000A 0089           00042         movwf   TRISE           ; inputs, Enable Parallel Slave port
000B 30FF           00043         movlw   0FFh
000C 0086           00044         movwf   TRISB           ;Set Port_B to all outputs
000D 3080           00045         movlw   b'10000000'     ;
000E 008C           00046         movwf   PIE1            ;Enable Parallel Slave Port interrupt
000F 1283           00047         bcf     STATUS,RP0      ;Select register Bank0
                    00048
0010 0821           00049         movf    OUTDATA,W       ;Set output Data in PORTD
0011 0088           00050         movwf   PORTD
0012 30C0           00051         movlw   b'11000000'     ;Set GIE, PEIE (enable interrupts)
```

```
0013 008B           00052         movwf   INTCON
                    00053
0014                00054 Loop
0014 1920           00055         btfsc   FLAGREG,INFULL  ;Check if input data received
0015 2819           00056         goto    Checkout        ;No data ready, check output
0016 1120           00057         bcf     FLAGREG,INFULL  ;Clear input data ready flag
0017 0822           00058         movf    INDATA,W        ;Get Input data
0018 0086           00059         movwf   PORTB           ;Output input data to Port_B
0019                00060 Checkout
0019 18A0           00061         btfsc   FLAGREG,OUTRDY  ;Check if data output already
001A 2814           00062         goto    Loop            ;Not output yet, loop
001B 0AA3           00063         incf    COUNT, F        ;Increment output data
001C 0823           00064         movf    COUNT,W         ;Get output data
001D 00A1           00065         movwf   OUTDATA         ;Put data in output queue
001E 14A0           00066         bsf     FLAGREG,OUTRDY  ;Set flag for interrupt routine
001F 2814           00067         goto    Loop
                    00068
                    00069 ;*********************************************************************
                    00070 ;*Interrupt Service Routine
                    00071 ;*      Inputs: FLAGREG - Flag register to/from the main routine:
                    00072 ;*                        Bit 1: OUTRDY - To Service_Int, indicates
                    00073 ;*                                        data ready in output queue
                    00074 ;*              OUTDATA - Output data queue
                    00075 ;*              PIR1    - Interrupt flag register
                    00076 ;*              TRISE   - Parallel slave port flag register
                    00077 ;*              PORTD   - Input data from slave port
                    00078 ;*
                    00079 ;*      Outputs:
                    00080 ;*              PORTD   - Output data to slave port
                    00081 ;*              INDATA  - Input data queue
                    00082 ;*              FLAGREG - Flag register to/from the main routine:
                    00083 ;*                        Bit 0: ERROR -  From Service_Int, indicates
                    00084 ;*                                        input buffer overflow
                    00085 ;*                        Bit 2: INFULL - From Service_Int, indicates
                    00086 ;*                                        data received and in INDATA
                    00087 ;*********************************************************************
                    00088
0020                00089 Service_Int
0020 1F8C           00090         btfss   PIR1,PSPIF      ;Test for Peripheral interrupt
0021 2832           00091         goto    Intout          ;Not a Peripheral interrupt, exit
0022 138C           00092         bcf     PIR1,PSPIF      ;Clear Peripheral interrupt
0023 1683           00093         bsf     STATUS,RP0      ;Select Bank1
0024 1F89           00094         btfss   TRISE,IBF       ;Check if input data ready
0025 282A           00095         goto    Notinput        ;No input, check output
0026 1283           00096         bcf     STATUS,RP0      ;Input ready, select Bank0
0027 1520           00097         bsf     FLAGREG,INFULL  ;Set flag for main routine
0028 0808           00098         movf    PORTD,W         ;Get input data
0029 00A2           00099         movwf   INDATA          ;Put byte in input queue
002A                00100 Notinput
002A 1B09           00101         btfsc   TRISE,OBF       ;Check if output data read
002B 2832           00102         goto    Intout          ;Not read, exit
002C 1283           00103         bcf     STATUS,RP0      ;Select Bank0
002D 1CA0           00104         btfss   FLAGREG,OUTRDY  ;Check if data in output queue
002E 2832           00105         goto    Intout          ;Output not read, exit
002F 0821           00106         movf    OUTDATA,W       ;Get data from queue
0030 0888           00107         movf    PORTD, F        ;Put data in output buffer
0031 10A0           00108         bcf     FLAGREG,OUTRDY  ;Clear flag for main routine
0032                00109 Intout
0032 1683           00110         bsf     STATUS,RP0      ;Select Bank1
0033 1A89           00111         btfsc   TRISE,IBOV      ;Check input buffer overflow flag
0034 2837           00112         goto    Interror        ;If not clear, error
0035 1283           00113         bcf     STATUS,RP0      ;Select Bank0
0036 0009           00114         retfie                  ;Re-enable GIE and return
0037                00115 Interror
0037 1283           00116         bcf     STATUS,RP0      ;Select Bank0
0038 1420           00117         bsf     FLAGREG,err     ;Set error flag for main routine
```

```
0039 0009                retfie                  ;Re-enable GIE and return

                         end
MEMORY USAGE MAP ('X' = Used,  '-' = Unused)


0000 : X----XXXXXXXXXXX XXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXX XXXXXXXXXX------

All other memory blocks unused.

Program Memory Words Used:    54
Program Memory Words Free:  1994


Errors   :     0
Warnings :     0 reported,     0 suppressed
Messages :     0 reported,     6 suppressed
```

MICROCHIP

WORLDWIDE SALES AND SERVICE

AMERICAS

Corporate Office
Microchip Technology Inc.
2355 West Chandler Blvd.
Chandler, AZ 85224-6199
Tel: 480-786-7200 Fax: 480-786-7277
Technical Support: 480-786-7627
Web Address: http://www.microchip.com

Atlanta
Microchip Technology Inc.
500 Sugar Mill Road, Suite 200B
Atlanta, GA 30350
Tel: 770-640-0034 Fax: 770-640-0307

Boston
Microchip Technology Inc.
5 Mount Royal Avenue
Marlborough, MA 01752
Tel: 508-480-9990 Fax: 508-480-8575

Chicago
Microchip Technology Inc.
333 Pierce Road, Suite 180
Itasca, IL 60143
Tel: 630-285-0071 Fax: 630-285-0075

Dallas
Microchip Technology Inc.
4570 Westgrove Drive, Suite 160
Addison, TX 75248
Tel: 972-818-7423 Fax: 972-818-2924

Dayton
Microchip Technology Inc.
Two Prestige Place, Suite 150
Miamisburg, OH 45342
Tel: 937-291-1654 Fax: 937-291-9175

Detroit
Microchip Technology Inc.
Tri-Atria Office Building
32255 Northwestern Highway, Suite 190
Farmington Hills, MI 48334
Tel: 248-538-2250 Fax: 248-538-2260

Los Angeles
Microchip Technology Inc.
18201 Von Karman, Suite 1090
Irvine, CA 92612
Tel: 949-263-1888 Fax: 949-263-1338

New York
Microchip Technology Inc.
150 Motor Parkway, Suite 202
Hauppauge, NY 11788
Tel: 631-273-5305 Fax: 631-273-5335

San Jose
Microchip Technology Inc.
2107 North First Street, Suite 590
San Jose, CA 95131
Tel: 408-436-7950 Fax: 408-436-7955

AMERICAS (continued)

Toronto
Microchip Technology Inc.
5925 Airport Road, Suite 200
Mississauga, Ontario L4V 1W1, Canada
Tel: 905-405-6279 Fax: 905-405-6253

ASIA/PACIFIC

Hong Kong
Microchip Asia Pacific
Unit 2101, Tower 2
Metroplaza
223 Hing Fong Road
Kwai Fong, N.T., Hong Kong
Tel: 852-2-401-1200 Fax: 852-2-401-3431

Beijing
Microchip Technology, Beijing
Unit 915, 6 Chaoyangmen Bei Dajie
Dong Erhuan Road, Dongcheng District
New China Hong Kong Manhattan Building
Beijing 100027 PRC
Tel: 86-10-85282100 Fax: 86-10-85282104

India
Microchip Technology Inc.
India Liaison Office
No. 6, Legacy, Convent Road
Bangalore 560 025, India
Tel: 91-80-229-0061 Fax: 91-80-229-0062

Japan
Microchip Technology Intl. Inc.
Benex S-1 6F
3-18-20, Shinyokohama
Kohoku-Ku, Yokohama-shi
Kanagawa 222-0033 Japan
Tel: 81-45-471- 6166 Fax: 81-45-471-6122

Korea
Microchip Technology Korea
168-1, Youngbo Bldg. 3 Floor
Samsung-Dong, Kangnam-Ku
Seoul, Korea
Tel: 82-2-554-7200 Fax: 82-2-558-5934

Shanghai
Microchip Technology
RM 406 Shanghai Golden Bridge Bldg.
2077 Yan'an Road West, Hong Qiao District
Shanghai, PRC 200335
Tel: 86-21-6275-5700 Fax: 86 21-6275-5060

ASIA/PACIFIC (continued)

Singapore
Microchip Technology Singapore Pte Ltd.
200 Middle Road
#07-02 Prime Centre
Singapore 188980
Tel: 65-334-8870 Fax: 65-334-8850

Taiwan, R.O.C
Microchip Technology Taiwan
10F-1C 207
Tung Hua North Road
Taipei, Taiwan, ROC
Tel: 886-2-2717-7175 Fax: 886-2-2545-0139

EUROPE

United Kingdom
Arizona Microchip Technology Ltd.
505 Eskdale Road
Winnersh Triangle
Wokingham
Berkshire, England RG41 5TU
Tel: 44 118 921 5858 Fax: 44-118 921-5835

Denmark
Microchip Technology Denmark ApS
Regus Business Centre
Lautrup hoj 1-3
Ballerup DK-2750 Denmark
Tel: 45 4420 9895 Fax: 45 4420 9910

France
Arizona Microchip Technology SARL
Parc d'Activite du Moulin de Massy
43 Rue du Saule Trapu
Batiment A - ler Etage
91300 Massy, France
Tel: 33-1-69-53-63-20 Fax: 33-1-69-30-90-79

Germany
Arizona Microchip Technology GmbH
Gustav-Heinemann-Ring 125
D-81739 München, Germany
Tel: 49-89-627-144 0 Fax: 49-89-627-144-44

Italy
Arizona Microchip Technology SRL
Centro Direzionale Colleoni
Palazzo Taurus 1 V. Le Colleoni 1
20041 Agrate Brianza
Milan, Italy
Tel: 39-039-65791-1 Fax: 39-039-6899883

11/15/99

DNV Certification, Inc.
USA

ANSI • RAB
QMS
ACCREDITED

DNV MSC
The Netherlands
Accredited by the RvA

DNV

ISO 9001 / QS-9000
REGISTERED FIRM

Microchip received QS-9000 quality system certification for its worldwide headquarters, design and wafer fabrication facilities in Chandler and Tempe, Arizona in July 1999. The Company's quality system processes and procedures are QS-9000 compliant for its PICmicro® 8-bit MCUs, KEELOQ® code hopping devices, Serial EEPROMs and microperipheral products. In addition, Microchip's quality system for the design and manufacture of development systems is ISO 9001 certified.

Information contained in this publication regarding device applications and the like is intended for suggestion only and may be superseded by updates. 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. The Microchip logo and name are registered trademarks of Microchip Technology Inc. in the U.S.A. and other countries. All rights reserved. All other trademarks mentioned herein are the property of their respective companies.