INTRODUCTION

The MCP25050 I/O Expander is an effective device that is used in a Controller Area Network (CAN) which operates without the use of a microcontroller. It supports CAN V2.0B with bit rates up to 1Mb/s. Since the I/O Expander is a stand-alone device, it can be configured to user defaults using a software template. These defaults are stored in non-volatile EPROM. A network protocol must be chosen that supports a Master Node. The Master Node is required for peer to peer communications between I/O Expander Nodes and therefore handles communication to and from all I/O Expander Nodes. For this design we have chosen the CAN-NET protocol which provides a generic framework for communication that natively supports I/O Expander Nodes. The CAN-NET framework allows users to develop a proprietary protocol for use by their own products.

This Application Note describes a control system for a scissors lift that handles all of the operations and movements using one Master Node and three I/O Expander Nodes. The nodes are distributed throughout the vehicle and are connected together utilizing the 2-wire CAN interface. The master Node consists of a PIC16F874 working with an MCP2510 CAN controller.

With the substantial I/O capability of the expanders, all of the scissors lift control signals are able to seamlessly communicate with each other. The MCP25050 has many peripherals such as digital I/O, four 10-bit A/D channels, and two PWM outputs with up to 10-bits of resolution. Utilizing the I/O Expanders reduces the size of each node along with having the ability to control a large system with a few wires rather than using complex wiring harnesses.

SYSTEM OVERVIEW

A scissors lift is essentially a mobile work platform enabling the user to reach relatively high places. The concept behind this vehicle is to have versatile maneuverability along with the ability to control the height. The basic block diagram is shown in Figure 1. All the actuators in the system are hydraulically based including the traction motors. A single DC motor drives a hydraulic pump and electro-hydraulic valves route the fluid to the appropriate actuator. The operator has complete control of the system from an operator panel located on the work platform. A single axis joystick controls forward and reverse motion while left and right steering is activated by a thumb controlled rocker switch on the top of the joystick. Raising and lowering of the platform is accomplished with UP and DOWN push buttons. A battery indicator and horn button are also located on the panel.
The system uses CAN to bring all of the controls together utilizing the CAN bus shown in Figure 2. The CAN bus replaces large wiring harness and the controls are combined into a node. Each node handles the inputs and outputs along with transmitting and receiving information utilizing the bus. The bus consists of four wires. Two power wires and two CAN wires. The master controller supplies the main power and the nodes accept this power from the bus. Each node is regulated at 5V.
FIGURE 2: SYSTEM DIAGRAM

LEGEND
AI = Analog Input
AO = Analog Output (PWM)
DI = Digital Input
DO = Digital Output

NODE NAMES
Node 0, Master Controller
Node 10, Power Node at Battery
Node 11, Operator Node
Node 12, Valve At Manifold

FUNCTIONS
Up DO Follows Up DI
Down DO Follows Down DI
Steer L DO Follows Steer L DI
Steer R DO Follows Steer R DI
Horn DO Follows Horn DI
Battery AO Follows Battery AI
Key DO Follows Key DI
DC Drive AO Follows Fwd/Rev AI
or Runs at 50% with Up or Down

Power Node #10
MCP25050
DC Drive, AO
Key, DI
Battery, AI

Operator Node #11
MCP25050
Up, DI
Down, DI
Horn, DI
Battery, AO
Steer Left, DI
Steer Right, DI
Forward, AI
Reverse, AI

Valve Node #12
MCP25050
Steer Left, DO
Steer Right, DO
Up, DO
Down, DO
Horn, DO
Forward, DO
Reverse, DO
POWER NODE
The operation of Power Node is shown in Table 1. The battery input is scaled down from 12V and applied to one of the analog inputs on the MCP25050. The DC Drive is controlled by one of the PWM outputs of the MCP25050. The output signal is actually a PWM signal which is filtered into a DC voltage and run into the speed input of the DC Drive. The forward and reverse movement of the joystick determines the duty cycle while moving. When one of the up or down buttons are pressed the duty cycle will run at 50%. A key-switch in the base unit is connected to a digital input on the MCP25050.

<table>
<thead>
<tr>
<th>Operation</th>
<th>Type</th>
<th>Direction</th>
<th>Port Pin Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Battery</td>
<td>Analog</td>
<td>Input</td>
<td>AN0</td>
</tr>
<tr>
<td>Key</td>
<td>Digital</td>
<td>Input</td>
<td>GP1</td>
</tr>
<tr>
<td>DC Drive</td>
<td>PWM</td>
<td>Output</td>
<td>PWM1</td>
</tr>
</tbody>
</table>

TABLE 1: OPERATION OF POWER NODE

OPERATOR NODE
The operation of Operator Control Node is shown in Table 2. Operator Control Node controls all operations of the system from the work platform. The up and down momentary buttons are digital inputs which control their corresponding hydraulic valves and run the DC Drive at 50% speed. The joystick has a thumb operated momentary rocker switch for left and right steering. Forward and reverse motion of the lift is controlled by 2 potentiometers in the joystick, which are applied to two of the analog inputs on the MCP25050. These operations also control their corresponding hydraulic valves. The horn is a momentary button that is connected to a digital input and controls the horn relay. The battery voltage is displayed on an analog panel meter which is driven from one of the PWM outputs on the MCP25050.

<table>
<thead>
<tr>
<th>Operation</th>
<th>Type</th>
<th>Direction</th>
<th>Port Pin Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Forward</td>
<td>Analog</td>
<td>Input</td>
<td>AN0</td>
</tr>
<tr>
<td>Reverse</td>
<td>Analog</td>
<td>Input</td>
<td>AN1</td>
</tr>
<tr>
<td>Battery</td>
<td>PWM</td>
<td>Output</td>
<td>PWM1</td>
</tr>
<tr>
<td>Horn</td>
<td>Digital</td>
<td>Input</td>
<td>GP3</td>
</tr>
<tr>
<td>Steer Left</td>
<td>Digital</td>
<td>Input</td>
<td>GP4</td>
</tr>
<tr>
<td>Steer Right</td>
<td>Digital</td>
<td>Input</td>
<td>GP5</td>
</tr>
<tr>
<td>Up</td>
<td>Digital</td>
<td>Input</td>
<td>GP6</td>
</tr>
<tr>
<td>Down</td>
<td>Digital</td>
<td>Input</td>
<td>GP7</td>
</tr>
</tbody>
</table>

TABLE 2: OPERATION OF OPERATOR CONTROL NODE

VALVE NODE
The operation of Valve Control Node is shown in Table 3. Valve Control Node controls the hydraulic valves located at the manifold. All signals come from digital outputs on the MCP25050.

<table>
<thead>
<tr>
<th>Operation</th>
<th>Type</th>
<th>Direction</th>
<th>Port Pin Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Up</td>
<td>Digital</td>
<td>Output</td>
<td>GP0</td>
</tr>
<tr>
<td>Down</td>
<td>Digital</td>
<td>Output</td>
<td>GP1</td>
</tr>
<tr>
<td>Horn</td>
<td>Digital</td>
<td>Output</td>
<td>GP2</td>
</tr>
<tr>
<td>Steer Left</td>
<td>Digital</td>
<td>Output</td>
<td>GP3</td>
</tr>
<tr>
<td>Steer Right</td>
<td>Digital</td>
<td>Output</td>
<td>GP4</td>
</tr>
<tr>
<td>Forward</td>
<td>Digital</td>
<td>Output</td>
<td>GP5</td>
</tr>
<tr>
<td>Reverse</td>
<td>Digital</td>
<td>Output</td>
<td>GP6</td>
</tr>
</tbody>
</table>

HARDWARE OVERVIEW
This reference design was implemented using CAN-NET development boards from Diversified Engineering. The CAN-NET Education board was used for the master control node and the CAN-NET I/O Expander Node was used for all satellite nodes. The CAN-NET I/O Expander Node is a versatile development platform for the MCP25050. Any combination of inputs and outputs can be realized by selecting the proper connections on the I/O header. Schematics for these boards are included in the appendix.

The CAN data rate selected for this system is 125 kbps.

CAN-NET GENERAL PURPOSE PROTOCOL

General Structure
The Application Note uses a flexible, general-purpose protocol structure that is designed to provide a basic framework for development of specialized proprietary protocols. The goal is simplicity rather than sophistication. We first present the general structure and then customize it to the reference design problem.

The general structure divides the 29 bit extended message identifier into two types of messages: Broadcast and Directed. Broadcast messages have no specific destination. Directed messages are sent with one or more specific destinations intended. Most of the fields of the message identifier are the same for both message types.

The general structure is designed for systems with a maximum of 128 nodes with each node having a unique address. This restriction can be relaxed by rear-
ranging the number of bits allocated to each field or by adjusting the meaning of the Source and Dest/Subclass fields.

**FIGURE 3: I/O PROTOCOL**

The message identifier is structured by breaking it into six fields as indicated in the following table:

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Sym</th>
<th># Bits</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Priority</td>
<td>I</td>
<td>3</td>
<td>Priority: a 0 has priority over a 1</td>
</tr>
<tr>
<td>Class</td>
<td>C</td>
<td>8</td>
<td>Type of information</td>
</tr>
<tr>
<td>Broadcast</td>
<td>B</td>
<td>1</td>
<td>0-&gt;Directed, 1-&gt;Broadcast</td>
</tr>
<tr>
<td>Destination/Subclass</td>
<td>D</td>
<td>7</td>
<td>Destination or Class dependent modifier.</td>
</tr>
<tr>
<td>Source Address</td>
<td>S</td>
<td>7</td>
<td>Source address</td>
</tr>
<tr>
<td>Cmd</td>
<td>C</td>
<td>3</td>
<td>Reserved for hardware restrictions of node</td>
</tr>
</tbody>
</table>

These fields are mapped onto the 29-bit message ID in the Microchip parts through the use of four one-byte registers. This mapping is common for the 2510 parts and the I/O Expander parts.

Priority - The Priority bits are the upper three bits in the identifier and are used to resolve priority conflicts if two nodes want to transmit at the same time. A 0 has priority over a 1.

Class - The Class categorizes the type of information carried by the message. Eight bits support 256 classes or types of information. As will be discussed below, Broadcast type messages have a Subclass field that further expands the number of categories supported.

Broadcast - The Broadcast bit is a flag that identifies the message as a Directed message (0) or as a Broadcast message (1).

Dest/Subclass - The Dest/Subclass field is a seven-bit field. Its definition depends on the preceding Broadcast flag.

If the message is a Directed message then this is a Destination field and contains the address of the node or nodes to which the message is directed. A maximum of 128 node addresses is allowed.

If the message is a Broadcast message then this is a Subclass field that further categorizes the Class of information carried by the message. The meaning of the Subclass field depends on the specific Class.

Source - The Source field identifies the node that produced the message. A maximum of 128 node addresses are allowed.
**Cmd** - This three-bit field is set aside as an additional extension to the Class field to further identify the contents of the message. It is suggested that it be used to distinguish between multiple message types contained within the same node. This is how the I/O Expander devices use it and it is the only part of the message ID that is hardware determined (in the case of I/O Expander devices) and not adjustable by the user.
CAN-NET PROTOCOL IMPLEMENTATION FOR REFERENCE DESIGN

Class Definition

The CAN communication for the reference design consists only of messages being exchanged between the Controller board and I/O Expander nodes, (i.e I/O Expanders cannot generate messages that can be decoded by other I/O Expanders).

All of the classes have associated data that is formatted in a specific manner that is fixed for an I/O Expander. Detailed information is available in the I/O Expander data sheet.

Class 1

These messages are generated by the I/O Expander for consumption by the Controller board.

The associated data is an eight-byte data group that contains all the measured data values measured by the MCP25050:

- The eight bits of IOINTFL indicate which inputs have changed since the last message.
- The eight bits of GPIO give the state of each of the inputs.
- The four bytes, AN0H, AN1H, AN2H, AN3H, give the upper 8 bits of the 10 bit A/D measurement from each activated A/D input.
- The two bytes, AN10L and AN32L, give the lower 2 bits of the 10-bit A/D measurement from each activated A/D input. The bits are left justified in the four nibbles that make up the two bytes as follows:
  - AN10L = (AN1:1,AN1:0,0,0 AN0:1,AN0:0,0,0)
  - AN32L = (AN3:1, AN3:0,0,0, AN2:1,AN2:0,0,0)

Class 2 and 3

These messages are generated by the Controller board for consumption by the MCP25050 devices.

The data group used by these Classes is the "Write Register" command of the MCP25050 devices that allows the writing of a data value directly into a register of the 2505x. In Class 2 the PWM registers are addressed and for Class 3 it is the GPLAT register that sets the digital output levels.

Node Addresses

Each node in the system is assigned a unique node address for use in the Source and Dest/Subclass fields.

<table>
<thead>
<tr>
<th>Address</th>
<th>Node</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>MasterController Node</td>
</tr>
<tr>
<td>10</td>
<td>Power Node</td>
</tr>
<tr>
<td>11</td>
<td>Operator Control Node</td>
</tr>
<tr>
<td>12</td>
<td>Valve Control Node</td>
</tr>
</tbody>
</table>

Master Controller Node

<table>
<thead>
<tr>
<th>Direction</th>
<th>Send</th>
<th>Receive</th>
</tr>
</thead>
<tbody>
<tr>
<td>Priority</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Class</td>
<td>2 or 3</td>
<td>1</td>
</tr>
<tr>
<td>Broadcast</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Dest/Subclass</td>
<td>10,11,12</td>
<td>0</td>
</tr>
<tr>
<td>Source</td>
<td>0</td>
<td>10,11,12</td>
</tr>
<tr>
<td>Cmd</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Power Node

<table>
<thead>
<tr>
<th>Direction</th>
<th>Send</th>
<th>Receive</th>
</tr>
</thead>
<tbody>
<tr>
<td>Priority</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Class</td>
<td>1,4</td>
<td>2,3</td>
</tr>
<tr>
<td>Broadcast</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Dest/Subclass</td>
<td>0</td>
<td>10</td>
</tr>
<tr>
<td>Source</td>
<td>10</td>
<td>0</td>
</tr>
<tr>
<td>Cmd</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Operator Control Node

<table>
<thead>
<tr>
<th>Direction</th>
<th>Send</th>
<th>Receive</th>
</tr>
</thead>
<tbody>
<tr>
<td>Priority</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Class</td>
<td>1,4</td>
<td>2,3</td>
</tr>
<tr>
<td>Broadcast</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Dest/Subclass</td>
<td>0</td>
<td>11</td>
</tr>
</tbody>
</table>
SOFTWARE OVERVIEW

Configuring the MCP25050 devices consists of providing arguments to a set of macros that generate a data table for MASM. In this sense, I/O Expander devices are configured rather than programmed.

The Controller board is programmed in the normal fashion but the specific details of the programming are not particularly important for the reference design since the primary purpose of the Controller board is to receive messages from the I/O Expanders and repackage the data to be sent to the other I/O Expanders.

I/O EXPANDER CONFIGURATION

For I/O Expander configurations that are static, (i.e. the configuration is not changed dynamically over the network), the important configuration parameters fall into two categories: network related items and I/O functions.

Choosing the network values other than the message ID's consist primarily in calculating the networking parameters determined by clock frequency and other physical characteristics of the network. The message ID's for transmitting and receiving messages are determined by the network protocol selected. For the reference design the message ID selection is described in detail above.

The MCP25050 can be configured to perform up to eight I/O functions. To chose from there are 8 digital inputs, 7 digital outputs, four 10-bit A/D channels, and two PWM outputs with up to 10-bits of resolution. Available with each of the I/O types are associated support functions such as message transmission triggered by a change in input. Scheduled message transmission can be used in addition to on-change messaging to insure the network is routinely informed of the current state of the inputs even if none of them have changed.

A combination of scheduled and on-change messages often is the best solution to routine updates with rapid response to change.

The data selections for the reference design are in files:

<table>
<thead>
<tr>
<th>File</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OPERNODE.ASM</td>
<td>Operator board</td>
</tr>
<tr>
<td>MAINNODE.ASM</td>
<td>Main board</td>
</tr>
<tr>
<td>VALVNODE.ASM</td>
<td>Hydraulic valve board</td>
</tr>
</tbody>
</table>

CONTROLLER BOARD SOFTWARE

The Controller board software is written to run on the Diversified Engineering CAN-NET Education board. The code for the CAN NET board is written in the PIC instruction set to be assembled using Microchip's MPLAB environment. There is significant use of macros to make the code more readable and less error prone. In addition to the macros defined at the top of the individual files a large number of macros can be found in the MACROS16.INC file. If in reading the code you come across an unfamiliar instruction it probably is a macro. We use the macros in MACROS16.INC extensively in writing code for PICs and have found that they increase readability and greatly reduce programming errors.

To simplify the code, the code that handled the LCD display and keypad input was removed. What remains is the initialization code that sets up the ports and initializes the MCP2510 CAN controller and a main loop that checks for CAN messages from the I/O Expander nodes and sends messages to the I/O Expander nodes.

The technique used by the program is to maintain a local set of variables that fully represent the state of the system. The variables are updated by messages received from the I/O Expander nodes and the new values are sent to the appropriate I/O Expanders.

Nine digital flags that contain the current state of the associated buttons or outputs represent the binary values.

<table>
<thead>
<tr>
<th>tbFlgUp</th>
<th>Up</th>
</tr>
</thead>
<tbody>
<tr>
<td>tbFlgDown</td>
<td>Down</td>
</tr>
<tr>
<td>tbFlgHorn</td>
<td>Horn</td>
</tr>
<tr>
<td>tbFlgLeft</td>
<td>Left</td>
</tr>
<tr>
<td>tbFlgRight</td>
<td>Right</td>
</tr>
<tr>
<td>tbFlgForward</td>
<td>Forward</td>
</tr>
<tr>
<td>tbFlgReverse</td>
<td>Reverse</td>
</tr>
<tr>
<td>tbFlgBattery</td>
<td>Battery LED</td>
</tr>
<tr>
<td>tbFlgKeyLED</td>
<td>Key LED</td>
</tr>
<tr>
<td>tbFlgKey</td>
<td>Key</td>
</tr>
</tbody>
</table>
The four analog variables are represented by four 1-byte quantities.

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>bBatteryLevel</td>
<td>Battery level 0 -&gt; 255</td>
</tr>
<tr>
<td>bDCDrive</td>
<td>DC Drive control level: 0 -&gt; 255</td>
</tr>
<tr>
<td>bForward</td>
<td>Joy stick level: 0 -&gt; 255</td>
</tr>
<tr>
<td>bReverse</td>
<td>Joy stick level: 0 -&gt; 255</td>
</tr>
</tbody>
</table>

Each time a message is received from a MCP25050, the received data is used to update the local binary and analog variables that maintain the state of the system. If a binary or analog value is received that should be sent to another of the I/O Expanders in the system a flag is set indicating that a message should be sent to that I/O Expander.

Each time around the main loop incoming messages are parsed and messages are generated for I/O Expanders.

The only calculations done by the controller board software are for the operation of the DC drive motor. The two analog values from the forward and reverse joy-stick inputs on the Operator Control Board are converted to a single PWM for the DC Drive and binary forward or reverse valve positions. A dead band is imposed so that the exact center of the joystick need not be known. Other than these calculations, the input data is simply sent back out to the appropriate node.

The controller software is contained in the files:

<table>
<thead>
<tr>
<th>File</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MCP2510.inc</td>
<td>Definitions and macros for 2510 support</td>
</tr>
<tr>
<td>Macros16.inc</td>
<td>Support macros</td>
</tr>
<tr>
<td>MainExp.asm</td>
<td>Main program</td>
</tr>
<tr>
<td>RefCode.asm</td>
<td>Code specific to the Reference Design</td>
</tr>
<tr>
<td>Ref.asm</td>
<td>I/O Expander Reference Design</td>
</tr>
<tr>
<td>Canlib.asm</td>
<td>2510 support functions</td>
</tr>
<tr>
<td>Lcd4bit.asm</td>
<td>LCD Handler</td>
</tr>
<tr>
<td>OperExp.asm</td>
<td>Operator Control Board</td>
</tr>
<tr>
<td>ValveExp.asm</td>
<td>Valve Control Board</td>
</tr>
</tbody>
</table>

CONCLUSION

The MCP25050 CAN I/O Expanders are an excellent and effective solution for new systems or simply adding them to an existing system. The advantage of the MCP25050 CAN I/O Expander is that an extra controller is not needed per node in order to utilize the CAN engine. Another advantage is that several Expanders can work off of the same CAN bus rather than using large and complicated wiring harnesses. This design demonstrates a useful way to integrate the expanders in a system using different types of inputs and outputs and also by providing a stepping stone to quickly start similar projects. From this example, several uses can be implemented simply by using the basic techniques from this design.

CONTACTING DIVERSIFIED ENGINEERING

Additional information and CAN related products may be obtained from Diversified Engineering by calling:
(203) 799-7875

or by visiting their web site:
www.DiversifiedEngineering.net

SOURCE CODE

Because of its overall size and number of files for the controller software, a complete source file is not provided. The complete source code is available as a single WinZip archive file, which may be downloaded from the Microchip corporate Web site at:

www.microchip.com
APPENDIX A: CAN-NET BOARD SCHEMATICS

FIGURE A-1: Main CAN-NET Board Schematic
FIGURE A-3: CAN I/O Expander Schematic (2 of 2)
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.

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.

Trademarks

The Microchip name and logo, the Microchip logo, FilterLab, KEELOQ, MPLAB, PIC, PICmicro, PICMASTER, PICSTART, PRO MATE, SEEVAL and The Embedded Control Solutions Company are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

dsPIC, ECONOMONITOR, FanSense, FlexROM, fuzzyLAB, In-Circuit Serial Programming, ICSP, ICEPIC, microID, microPort, Migratable Memory, MPASM, MPLIB, MPLINK, MPSIM, MXDEV, PICC, PICDEM, PICDEM.net, rfPIC, Select Mode and Total Endurance are trademarks of Microchip Technology Incorporated in the U.S.A.

Serialized Quick Term Programming (SQTP) is a service mark of Microchip Technology Incorporated in the U.S.A.

All other trademarks mentioned herein are property of their respective companies.

© 2002, Microchip Technology Incorporated, Printed in the U.S.A., All Rights Reserved.

Printed on recycled paper.
AMERICAS

Corporate Office
2335 West Chandler Blvd.
Chandler, AZ  85224-6199
Tel:  480-792-7200  Fax:  480-792-7277
Technical Support: 480-792-7627
Web Address: http://www.microchip.com

Rocky Mountain
2335 West Chandler Blvd.
Chandler, AZ  85224-6199
Tel:  480-792-7966  Fax:  480-792-7456

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

Boston
2 Lan Drive, Suite 120
Westford, MA  01886
Tel:  978-692-3848  Fax:  978-692-3821

Chicago
333 Pierce Road, Suite 180
Itasca, IL  60143
Tel:  630-285-0070  Fax:  630-285-0075

Dallas
4570 Westgrove Drive, Suite 160
Addison, TX  75001
Tel:  972-818-7943  Fax:  972-818-2924

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

Kokomo
2767 S. Albright Road
Kokomo, Indiana  46902
Tel:  765-864-8360  Fax:  765-864-8387

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

New York
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:  408-438-7950  Fax:  408-436-7955

Toronto
6285 Northam Drive, Suite 108
Mississauga, Ontario L4V 1X5, Canada
Tel:  905-673-0699  Fax:  905-673-6509

ASIA/PACIFIC

Australia
Microchip Technology Australia Pty Ltd
Suite 22, 41 Rawson Street
Epping 2121, NSW
Australia
Tel:  61-2-9868-6733  Fax:  61-2-9868-6755

China - Beijing
Microchip Technology Consulting (Shanghai) Co., Ltd., Beijing Liaison Office
Unit 915
Bei Hai Wan Tai Bldg.
No. 6 Chaoyangmen Beidajie
Beijing, 100027, No. China
Tel:  86-10-85282100  Fax:  86-10-85282104

China - Chengdu
Microchip Technology Consulting (Shanghai) Co., Ltd., Chengdu Liaison Office
Rm. 2401, 24th Floor,
Ming Xing Financial Tower
No. 88 TIDU Street
Chengdu 610016, China
Tel:  86-28-6766200  Fax:  86-28-6766599

China - Fuzhou
Microchip Technology Consulting (Shanghai) Co., Ltd., Fuzhou Liaison Office
Unit 71 Wusi Road
Fuzhou 350001, China
Tel:  86-591-7503506  Fax:  86-591-7503521

Japan
Microchip Technology Japan K.K.
Benex S-1 6F
3-18-20, Shin-yokohama
Kokoku-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, Gangnam-Ku
Seoul, Korea 135-882
Tel:  82-2-554-7200  Fax:  82-2-558-5934

Singapore
Microchip Technology Singapore Pte Ltd.
200 Middle Road
#07-02 Prime Centre
Singapore, 189980
Tel:  65-334-8870  Fax:  65-334-8850

Taiwan
Microchip Technology Taiwan
11F-3, No. 207
Tung Hua North Road
Taipei, 105, Taiwan
Tel:  886-2-2717-7175  Fax:  886-2-2545-0139

EUROPE

Denmark
Microchip Technology Nordic ApS
Rebus Business Centre
Lastrup høj 1-3
Ballrup DK-2750 Denmark
Tel:  45 4420 9895  Fax:  45 4420 9910

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

Germany
Microchip Technology GmbH
Gustav-Heinemann Ring 125
D-81739 Munich, Germany
Tel:  49-89-627-5700  Fax:  49-89-627-5060

Italy
Microchip Technology SRL
Centro Direzionale Colleoni
Palazzo Taurus 1 V. Le Colleoni 1
20041 Agrate Brianza
Milan, Italy
Tel:  39-039-673591-1  Fax:  39-039-6899883

United Kingdom
Arizona Microchip Technology Ltd.
505 Eskdale Road
Winnersh Triangle
Wokingham
Berkshire, England RG41 5TU
Tel:  44 118 921 5869  Fax:  44-118 921-5820

01/18/02