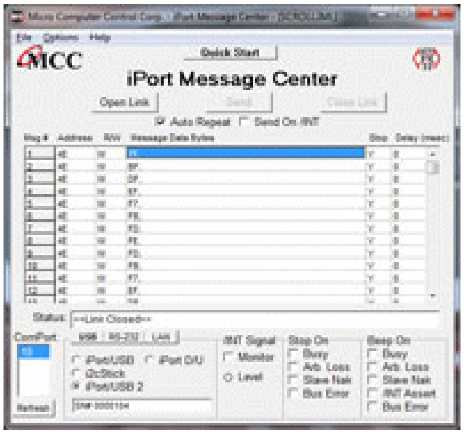
User’s Guide

iPort Utility Pack

for

Windows



iPort Message Center

☑ Auto Repeat ☐ Send On /INT

Msg #	Address	R/W	Message Data Bytes	Stop	Delay (msec)
1	4E	W	[illegible]	Y	0
2	4E	W	BF.	Y	0
3	4E	W	DF.	Y	0
4	4E	W	EF.	Y	0
5	4E	W	F7.	Y	0
6	4E	W	FB.	Y	0
7	4E	W	FD.	Y	0
8	4E	W	FE.	Y	0
9	4E	W	FD.	Y	0
10	4E	W	FB.	Y	0
11	4E	W	F7.	Y	0
12	4E	W	EF.	Y	0

Status: <<Link Closed>>

ComPort

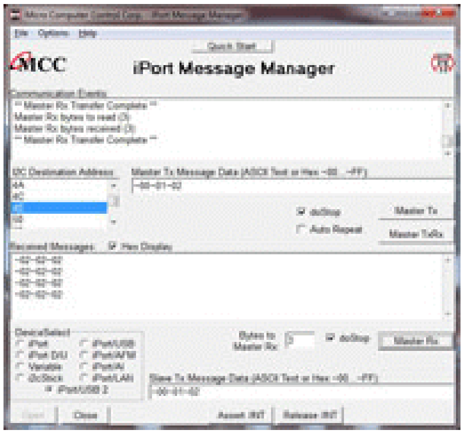
○ iPort/USB ○ iPort D/U
○ i2cStick
● iPort/USB 2

SN# 0000104

/INT Signal
☐ Monitor
○ Level

Stop On
☐ Busy
☐ Arb. Loss
☐ Slave Nak
☐ Bus Error

Beep On
☐ Busy
☐ Arb. Loss
☐ Slave Nak
☐ /INT Assert
☐ Bus Error



iPort Message Manager

Communication Events

** Master Rx Transfer Complete **
Master Rx bytes to read (3)
Master Rx bytes received (3)
** Master Rx Transfer Complete **

I2C Destination Address

4A
4C
4E
50

Master Tx Message Data (ASCII Text or Hex ~00...~FF)

~00~01~02

☑ doStop
☐ Auto Repeat

Received Messages ☑ Hex Display

~02~02~02
~02~02~02
~02~02~02
~02~02~02

DeviceSelect
○ iPort ○ iPort/USB
○ iPort D/U ○ iPort/AFM
○ Variable ○ iPort/AI
○ i2cStick ○ iPort/LAN
● iPort/USB 2

Bytes to Master Rx: 3 ☑ doStop

Slave Tx Message Data (ASCII Text or Hex ~00...~FF)

~00~01~02

Micro Computer Control
Small Area Network Specialists

www.mcc-us.com

DISCLAIMER: Micro Computer Control Corporation makes no representations or warranties with respect to the contents hereof and specifically disclaims any implied warranties of merchantability or fitness for any particular purpose. Further, Micro Computer Control Corporation reserves the right to revise the product described in this publication and to make changes from time to time in the content hereof without the obligation to notify any person of such revisions or changes.

WARNING - Life Support Applications: MCC products are not designed for use in life support appliances, devices, or systems where the malfunction of the product can reasonably be expected to result in a personal injury.

WARNING - Radio Frequency Emissions: This equipment has been tested and found to comply with the limits for a Class A digital device, pursuant to part 15 of the FCC rules, and the CE General Emissions Standard EN55032 and General Immunity Standard EN55035. These limits are designed to provide reasonable protection against interference when the equipment is operated in a ***commercial environment***. This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instruction manual, may cause interference to radio communications. Operation of this equipment in a ***residential area*** is likely to cause interference, in which case the user will be required to correct the interference at his own expense.

WARNING - Electrostatic Discharge (ESD) Precautions: Any damage caused by Electrostatic Discharge (ESD) through inadequate earth grounding is NOT covered under the warranty of this product. See the “Electrostatic (ESD) Precautions” section of this guide for more information.

Electronic Waste Notice - This product must NOT be thrown into general waste, but should be collected separately and properly recycled under local regulations.

Created in the United States of America
13-OCT-2018

Table of Contents

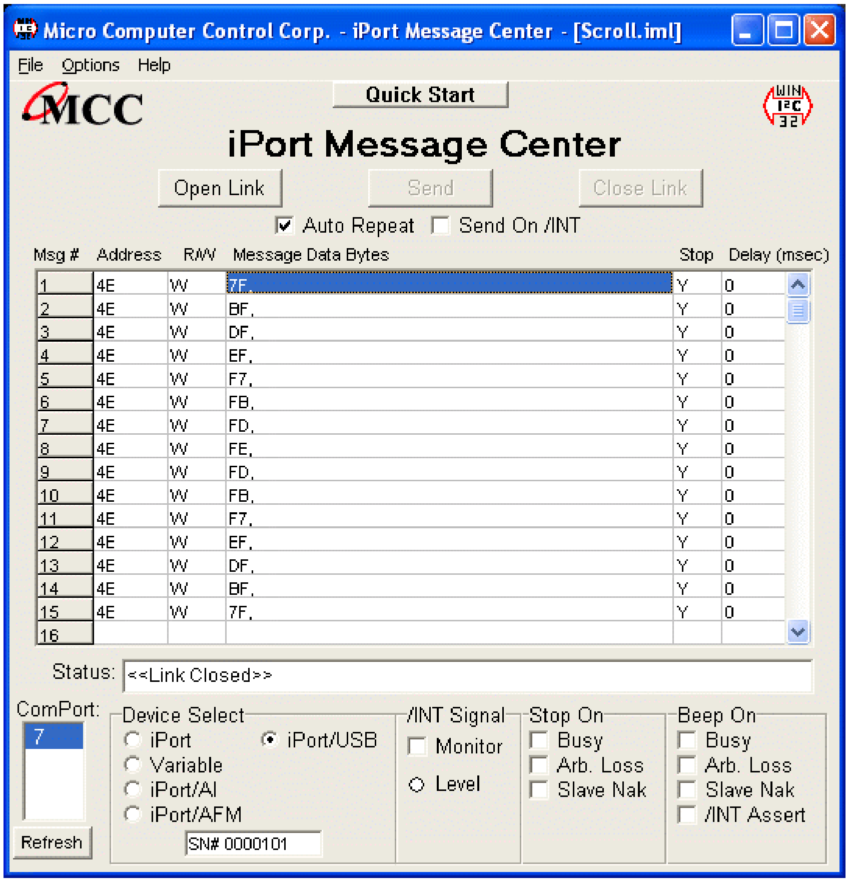
Introduction

The iPort Utility Pack for Windows is your express lane to I^{2}C Bus communications. The Utility Pack includes two (2) Windows-based applications (Message Center and Message Manager) that will help you get started sending and receiving I^{2}C Bus messages quickly and easily.

1.1 iPort Message Center

The iPort Message Center, our most popular application, operates with all versions of our I^{2}C Bus Host Adapters. With the Message Center, you can create, save, and automatically execute scripts of I^{2}C Bus messages. I^{2}C Bus message activity includes:

- Master Transmit
- Master Receive



☑ Auto Repeat ☐ Send On /INT

Msg #	Address	R/W	Message Data Bytes	Stop	Delay (msec)
1	4E	W	7F,	Y	0
2	4E	W	BF,	Y	0
3	4E	W	DF,	Y	0
4	4E	W	EF,	Y	0
5	4E	W	F7,	Y	0
6	4E	W	FB,	Y	0
7	4E	W	FD,	Y	0
8	4E	W	FE,	Y	0
9	4E	W	FD,	Y	0
10	4E	W	FB,	Y	0
11	4E	W	F7,	Y	0
12	4E	W	EF,	Y	0
13	4E	W	DF,	Y	0
14	4E	W	BF,	Y	0
15	4E	W	7F,	Y	0
16					

Status: <<Link Closed>>

ComPort: 7

Device Select: ○ iPort ● iPort/USB ○ Variable ○ iPort/AI ○ iPort/AFM SN# 0000101

/INT Signal: ☐ Monitor ○ Level

Stop On: ☐ Busy ☐ Arb. Loss ☐ Slave Nak

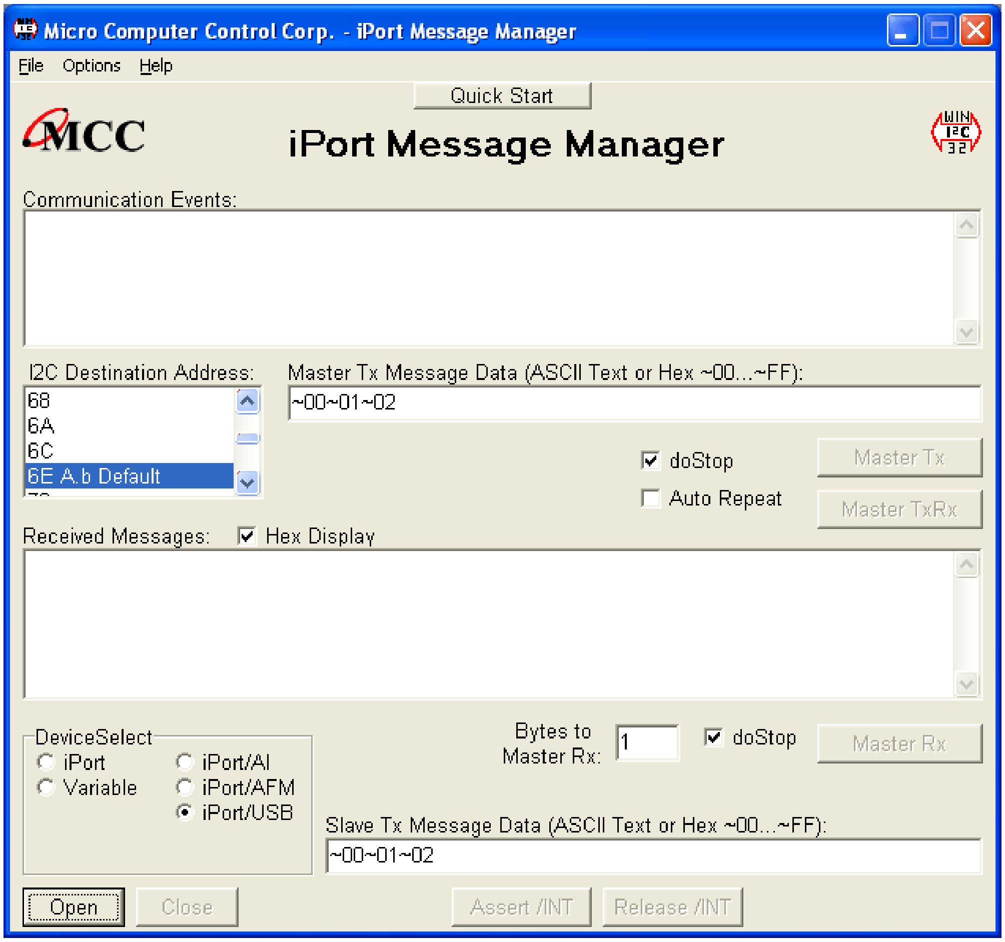
Beep On: ☐ Busy ☐ Arb. Loss ☐ Slave Nak ☐ /INT Assert

Main Screen (Typical)

1.2 iPort Message Manager

The iPort Message Manager operates with all versions of our I^{2}C Bus Host Adapters. Using the Message Manager, you can perform all four (4) modes of I^{2}C Bus message activity, including:

- Master Transmit
- Master Receive
- Slave Transmit
- Slave Receive



Communication Events:

I2C Destination Address: 68, 6A, 6C, 6E A.b Default

Master Tx Message Data (ASCII Text or Hex ~00...~FF): ~00~01~02

☑ doStop

□ Auto Repeat

Received Messages: ☑ Hex Display

DeviceSelect

○ iPort ○ iPort/AI

○ Variable ○ iPort/AFM

● iPort/USB

Bytes to Master Rx: 1 ☑ doStop

Slave Tx Message Data (ASCII Text or Hex ~00...~FF): ~00~01~02

Main Screen (Typical)

2 System Requirements

One of the following MCC I^{2}C Bus adapters:

1. iPort/AFM 2 (#MIIC-213) RS-232 to I^{2}C Bus Host Adapter with ASCII Fast Mode Interface.
2. iPort/LAN 2 (#MIIC-210) Ethernet to I^{2}C Bus Host Adapter.
3. i2cStick 2 (#MIIC-209) USB to I^{2}C Bus Host Adapter.
4. iPort/USB 2 (#MIIC-208) USB to I^{2}C Bus Host Adapter.
5. i2cStick (#MIIC-207) USB to I^{2}C Bus Host Adapter.
6. iPort/LAN (#MIIC-205) Ethernet to I^{2}C Bus Host Adapter.
7. iPort/USB (#MIIC-204) USB to I^{2}C Bus Host Adapter.
8. iPort/AFM (#MIIC-203) RS-232 to I^{2}C Bus Host Adapter with ASCII Fast Mode Interface.
9. iPort/AI (#MIIC-202) RS-232 to I^{2}C Bus Host Adapter with ASCII Interface
10. iPort (#MIIC-201) Windows to I^{2}C Bus Host Adapter.
11. iPort DLL/USB (#MIIC-201D/U) I^{2}C Bus Host Adapter.
12. Variable Clock Rate (#MIIC-201-V) I^{2}C Bus Host Adapter.

- Windows XP, Vista, 7, or 8.
- 1 free RS-232 Serial Port, USB port for USB-based adapters, or Ethernet port or Network access for Ethernet-based adapter

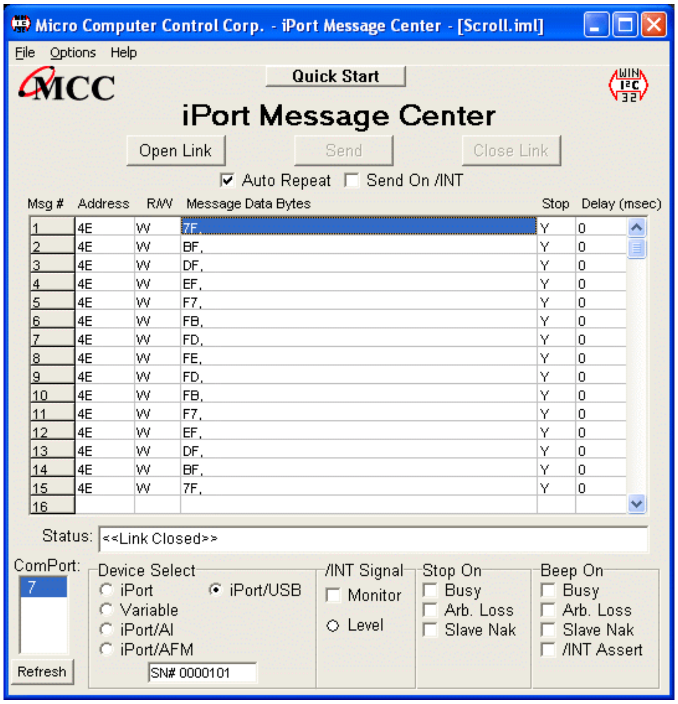
3 iPort Utility Pack Installation

The iPort Utility Pack for Windows software and installation instructions are available for download at the follow link:

www.mcc-us.com/iPutilPk/iPutilPk.htm

4 iPort Message Center

The iPort Message Center supports I^{2}C Master Transmit and Master Receive activities for all MCC I^{2}C Bus host adapters. With this program you can create, save, and execute scripts of I^{2}C Master messages.



☑ Auto Repeat ☐ Send On /INT

Msg #	Address	R/W	Message Data Bytes	Stop	Delay (msec)
1	4E	W	7F,	Y	0
2	4E	W	BF,	Y	0
3	4E	W	DF,	Y	0
4	4E	W	EF,	Y	0
5	4E	W	F7,	Y	0
6	4E	W	FB,	Y	0
7	4E	W	FD,	Y	0
8	4E	W	FE,	Y	0
9	4E	W	FD,	Y	0
10	4E	W	FB,	Y	0
11	4E	W	F7,	Y	0
12	4E	W	EF,	Y	0
13	4E	W	DF,	Y	0
14	4E	W	BF,	Y	0
15	4E	W	7F,	Y	0
16					

Status: <<Link Closed>>

ComPort: 7

Device Select: ○ iPort ● iPort/USB ○ Variable ○ iPort/AI ○ iPort/AFM SN# 0000101

/INT Signal: ☐ Monitor ○ Level

Stop On: ☐ Busy ☐ Arb. Loss ☐ Slave Nak

Beep On: ☐ Busy ☐ Arb. Loss ☐ Slave Nak ☐ /INT Assert

Main Screen (Typical)

The iPort Message Center allows a PC to become an I^{2}C Master transmitter or receiving device, sending I^{2}C messages between the PC and one or more I^{2}C devices across an I^{2}C Bus.

The iPort Message Center is designed to be a simple application for experimenting with I^{2}C messages. It provides methods to:

1. Enter/Edit a list of I^{2}C Master Transmit or Receive Messages.
2. Save and/or Load a list of I^{2}C Master messages to/from disk.
3. Transmit the current list of I^{2}C Master messages, with the option to auto-repeat upon completion, send on INT signal assertion (with INT signal supported adapters only), and beep or stop on special I^{2}C Bus events.
4. Use command line arguments to automatically load, send, and save I^{2}C messages from a batch file or another program.

Each I^{2}C message can transfer up to 999 bytes of 8-bit data, with Repeated Start and Time Delay options.

4.1 Message Center Operations

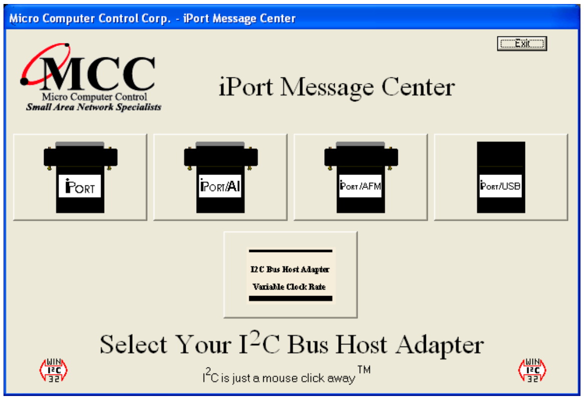
Communicating with another device on the I^{2}C Bus is easy. Just install the software as described in Section 3, then follow these simple steps:

4.1.1 Starting the Message Center

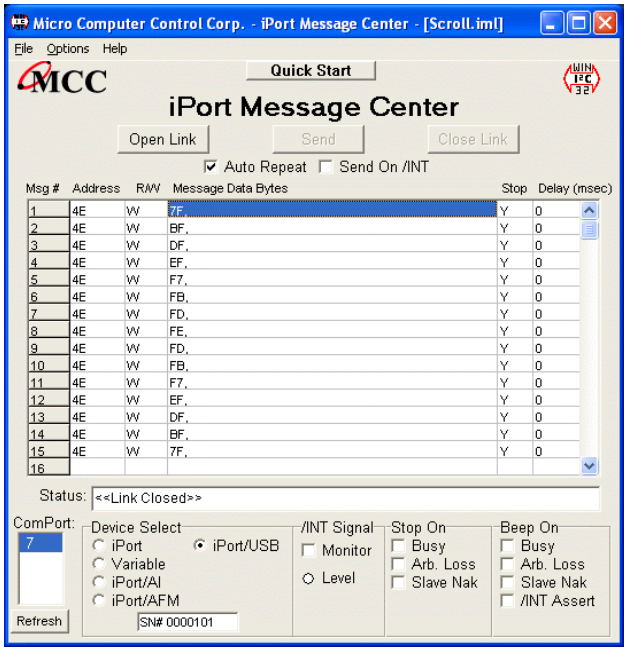
Click, Start | Programs | iPort Utility Pack | iPort Message Center

4.1.2 Selecting the Adapter

Select the I^{2}C adapter you are using by clicking the corresponding adapter image (see Opening Screen), or the Device Select checkbox (see Main Application Screen).


Micro Computer Control Corp. - iPort Message Center
Exit
MCC
Micro Computer Control
Small Area Network Specialists
iPort Message Center
iPort
iPort/AI
iPort/AFM
iPort/USB
I2C Bus Host Adapter
Variable Clock Rate
Select Your I2C Bus Host Adapter
I2C is just a mouse click away TM
WIN I2C 32
WIN I2C 32

Opening Screen (Typical)



☑ Auto Repeat ☐ Send On /INT

Msg #	Address	R/W	Message Data Bytes	Stop	Delay (msec)
1	4E	W	7F,	Y	0
2	4E	W	BF,	Y	0
3	4E	W	DF,	Y	0
4	4E	W	EF,	Y	0
5	4E	W	F7,	Y	0
6	4E	W	FB,	Y	0
7	4E	W	FD,	Y	0
8	4E	W	FE,	Y	0
9	4E	W	FD,	Y	0
10	4E	W	FB,	Y	0
11	4E	W	F7,	Y	0
12	4E	W	EF,	Y	0
13	4E	W	DF,	Y	0
14	4E	W	BF,	Y	0
15	4E	W	7F,	Y	0
16					

Status: <<Link Closed>>

ComPort: 7

Device Select
○ iPort ● iPort/USB
○ Variable
○ iPort/AI
○ iPort/AFM
SN# 0000101

/INT Signal
☐ Monitor
○ Level

Stop On
☐ Busy
☐ Arb. Loss
☐ Slave Nak

Beep On
☐ Busy
☐ Arb. Loss
☐ Slave Nak
☐ /INT Assert

Main Screen (Typical)

4.1.3 Select the Communications Port

Use the “ComPort:” control to select the communication port connected to the I^{2}C adapter. If a USB or Ethernet based device is selected, the serial number for the adapter is displayed. In addition to legacy RS-232 ports and USB or Ethernet based Virtual Communication Ports, Message Center supports USB, network or Bluetooth connected I^{2}C Bus adapters via the Windows Com driver.

4.1.4 Options Menu

Use the Options menu to override default Baud Rate and I^{2}C Bus Clock rate settings. Default settings and options are adapter dependant.

4.1.5 Establish Adapter Communications Link

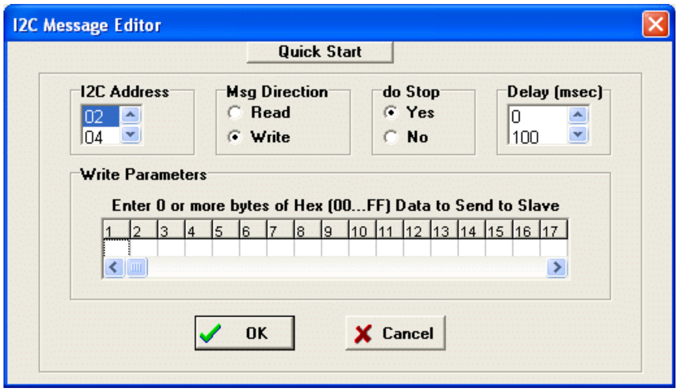
Establish the communications link to the I^{2}C adapter by clicking the Open Link button.

The Message Center sets the adapter’s own I^{2}C Slave address to 0xFE. Once the link has opened successfully, you are an active I^{2}C node. I^{2}C messages entered into the message spreadsheet can be transmitted upon request. If the link open is not successful, follow the on-screen directions. Make sure the communications port is working, is enabled in the Windows Device Manager, and is not being used by other software.

4.1.6 Entering or Editing I^{2}C Messages

I^{2}C messages can be entered with the Message Editor, or a previously recorded message list can be loaded from the File menu.

To enter or edit a message, open the “I^{2}C Message Editor” screen by double clicking on a message row in the spreadsheet.



I2C Address
02
04

Msg Direction
○ Read
● Write

do Stop
● Yes
○ No

Delay (msec)
0
100

Write Parameters

Enter 0 or more bytes of Hex (00...FF) Data to Send to Slave

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17

Use the I^{2}C Message Editor to:

1. Set I^{2}C Address.

   The I^{2}C Address is the I^{2}C slave address of the slave device being addressed on the bus. All slave addresses are displayed as even numbers (00...FE), representing the 7 most significant bits of the 8-bit slave address transmitted on the bus (aaaa aaa0).

   The I^{2}C adapter automatically supplies the 8th, least significant, Read/Write bit when it sends the slave address across the bus. For master write operations, the Read/Write bit is always transmitted as a logical 0 (aaaa aaa0). For master read operations, the Read/Write bit is always transmitted as a logical 1 (aaaa aaa1).

   Use the I^{2}C Address control to set the slave address of the slave device you want to address on the bus.

2. Set Message Read/Write Direction.

   As a bus master device, the I^{2}C adapter can write data to, or read data from, any device on the bus. Use the Msg Direction control to specify if the current message is a master write, or master read, operation. Upon making your selection, additional Write or Read parameters appear.

3. Specify Repeated Start Messages.

   I^{2}C Bus communications support an operation called Repeated Start. In this operation, a message is sent across the bus beginning with a Start Condition, but without a Stop Condition at the end of the message. The next message sent across the bus begins with a Start Condition, in this case a Repeated Start.

An I^{2}C Bus master, that successfully sends a message on the bus, owns the bus until that master sends a message with a terminating Stop Condition. The Repeated Start operation allows the bus master to retain control of the bus while sending one or more messages on the bus. This prevents other bus masters, in a multi-master system, from accessing the bus and interfering with message sequences.

The Message Center supports Repeated Starts with the doStop control. Sending an I^{2}C message with doStop enabled will cause the message to be terminated with a Stop Condition. Sending an I^{2}C message with doStop disabled will cause the message to end without a Stop Condition, allowing the next message to be sent with a Repeated Start.

4. Set Time Delay.

   Message Center supports time delays after the completion of a message. Time delays can be used to synchronize or sequence bus messages with the activity of external devices.

5. Specify Write Data or Read Byte Count.

   Enter the hexadecimal data you want to write to a slave receiver device, or the number of data bytes to read from a slave transmitter. Message Center supports up to 32 bytes of 8-bit data per message.

   NOTE: The data you send may have special meaning to the receiving slave device, but to the Message Center, and the I^{2}C adapter, message data has no special meaning. Consult your slave device’s data sheet for details.

Click OK to accept the message and enter it into the spreadsheet.

Master Write messages display the message data in the spreadsheet. Master Read messages display 0xFF placeholders in the spreadsheet. Upon execution, actual data received from a slave transmitter replaces the placeholders in the message spreadsheet.

Repeat above steps for additional messages. The Message Center supports up to 32,000 messages in a list.

4.1.7 Inserting and Deleting Messages

You can insert a new message between existing messages by clicking once on a message below where you want to insert, then press the “Insert” key on your keyboard. The Message Editor also remembers the last message displayed, so double clicking on a blank spreadsheet row will allow you to copy a message. Delete a message by single clicking on the message row and pressing the “Delete” key on your keyboard.

4.1.8 Saving or Loading Message Lists

Message Center I^2C message lists can be saved to, or loaded from, a disk file. To save the current message list, click File|Save on the menu bar. To open an existing message list, click File|Open List on the menu bar.

Message lists are maintained in ASCII text files (*.IML) that can be edited manually or created with a customer-developed program. See message list files for details.

4.1.9 Send the Message List

An I^2C message list can be sent manually, or automatically in response to an INT signal assertion (with INT signal supported adapters only). To send the list manually, click the Send button on the main application screen. To send the list in response to an INT signal assertion (low), enable the “/INT Signal Monitoring” checkbox, and check the “Send on /INT” checkbox. The list will be sent each time the INT signal is asserted.

The Message Center also supports the repeated sending of a message list. If the Auto Repeat checkbox is checked, a message list will automatically repeat upon completion.

4.1.10 Special Event Handling

The Message Center supports the early termination of a message list, and beep on special events. See the “Stop On” and “Beep On” controls on the main application screen of available options.

4.1.11 Slave Not Acknowledging

If you get a “Slave Not Acknowledging” message in the Status window, this could indicate you have the wrong address in the I^{2}C Destination Address, or the device is not answering to its address. Some slave devices temporarily stop acknowledging their address. Consult the slave device’s data sheet for details.

4.2 Command Line Arguments

The Message Center can be controlled via command line arguments. This feature allows the Message Center to be accessed from a batch file or another program.

Message Center **Command Line Arguments**	
Command	**Description**
iPort, iPort/AI, iPort/AFM, Variable, iPort/USB, iPort/LAN, i2cStick, iPort/USB2, i2cStick2, iPort/LAN2, iPort/AI2, iPort/AFM2	Specify I^{2}C adapter type.*
COM1...COM99	Specify RS-232 communication port.
BAUD19200, BAUD57600, BAUD115200	Set RS-232 Baud Rate.*
CLOCK12.5K, CLOCK23K, CLOCK86K, CLOCK100K, CLOCK400K, VCLOCK	Set I^{2}C Bus Clock Rate.*
Monitor/INT	Enable /INT Signal Monitor.*
StopOnBusy	Stop sending on I^{2}C adapter busy.
StopOnArbLoss	Stop sending on I^{2}C Bus Arbitration Loss.
StopOnNak	Stop on Slave Negative Acknowledgment.
BeepOnBusy	Beep on I^{2}C adapter busy.
BeepOnArbLoss	Beep on I^{2}C Bus arbitration loss.
BeepOnNak	Beep on Slave Negative Acknowledgment.
BeepOn/INT	Beep on /INT signal assert (low).*
AutoLoad	Load I^{2}C message list file.
AutoSave	Save I^{2}C message list file.
AutoOpen	Open link to I^{2}C adapter.
AutoSend	Send I^{2}C message list.
AutoExit	Exit after sending message list.

* Adapter specific commands. See command details below.

Command Line Syntax: imsgctr.exe AdapterType argument-list

Example: imsgctr.exe iPort/AFM adctest01.iml AutoOpen AutoSend AutoExit

4.2.1 Set Adapter Type

iPort/AFM2	iPort/AFM 2 (#MIIC-213)
iPort/AI2	iPort/AI 2 (#MIIC-212)
iPort/LAN2	iPort/LAN 2 (#MIIC-210)
i2cStick 2	i2cStick 2 (#MIIC-209)
iPort/USB2	iPort/USB 2 (#MIIC-208)
i2cStick	i2cStick (#MIIC-207)
iPort/LAN	iPort/LAN (#MIIC-205)
iPort/USB	iPort/USB (#MIIC-204)
iPort/AFM	iPort/AFM (#MIIC-203)
iPort/AI	iPort/AI (#MIIC-202)
iPort	iPort (#MIIC-201)
Variable	Variable Clock (#MIIC-201-V)

The Adapter Type argument should be the first argument in the argument list as it controls the availability of other arguments. If the Adapter Type is not specified, the startup adapter selection screen will be presented.

4.2.2 Set RS-232 Communication Port

1st Available ComPort	(Default)
COM1...COM99	

Set the RS-232 communications port attached to the I^{2}C adapter.

4.2.3 Set RS-232 Baud Rate

BAUD19200	(Default*)
BAUD57600	(iPort/AFM, iPort/USB, iPort/LAN, i2cStick*, iPort/USB 2*, i2cStick 2*, iPort/LAN 2, iPort/AFM 2 ONLY)
BAUD115200	(iPort/AFM, iPort/USB, iPort/LAN, i2cStick*, iPort/USB 2*, i2cStick 2*, iPort/LAN 2, iPort/AFM 2 ONLY)

Set the RS-232 Baud Rate. *iPort/USB 2, i2cStick, and i2cStick 2 internally remaps 19.2K baud and 57.6K baud to 115.2K baud.

4.2.4 Set I^{2}C Bus Clock Rate

CLOCK12.5K	(iPort ONLY)

CLOCK23K	iPort/AFM, iPort/USB, iPort/LAN, i2cStick, iPort/USB 2, i2cStick 2, iPort/LAN 2, iPort/AFM 2 ONLY
CLOCK86K	iPort/AFM, iPort/USB, iPort/LAN, i2cStick, iPort/USB 2, i2cStick 2, iPort/LAN 2, iPort/AFM 2 ONLY
CLOCK100K	Default
CLOCK400K	iPort/AFM, iPort/USB, iPort/LAN, i2cStick, iPort/USB 2, i2cStick 2, iPort/LAN 2, iPort/AFM 2 ONLY
VCLOCK=nnnHz	(Variable ONLY. nnn=451...57787)

Set the I^{2}C Bus Clock Rate to the specified value. The defaults rate for the Variable Clock adapter is 451Hz. The Variable Clock adapter does not support all rates within the specified range. The Message Center will adjust the specified rate to the nearest available supported rate.

4.2.5 Enable /INT Signal Monitor

Monitor/INT (on INT signal supported adapters only. Default=OFF)

Enable /INT signal monitoring.

4.2.6 Stop On Busy

StopOnBusy (Default=OFF)

Stop sending I^{2}C messages if the adapter returns a "Busy" response to the host computer.

4.2.7 Stop On Arbitration Loss

StopOnArbLoss (Default=OFF)

Stop sending I^{2}C messages if the adapter returns a "Bus Arbitration Loss" response to the host computer. Bus Arbitration Loss occurs when another I^{2}C Bus master wins arbitration while the adapter is attempting to become a bus master.

4.2.8 Stop On Slave Negative Acknowledgment

StopOnNak (Default=OFF)

Stop sending I^{2}C messages if the adapter returns a "Slave Not Acknowledging" response to the host computer. Slave Not Acknowledging occurs when the adapter is attempting to become a bus master and no slave device acknowledges the transmitted slave address.

4.2.9 Beep On Busy

BeepOnBusy (Default=OFF)

Generate a host computer beep if the adapter returns a "Busy" response to the host computer.

4.2.10 Beep On Arbitration Loss

BeepOnArbLoss (Default=OFF)

Generate a host computer beep if the adapter returns a "Bus Arbitration Loss" response to the host computer. Bus Arbitration Loss occurs when another I^{2}C Bus master wins arbitration while the adapter is attempting to become a bus master.

4.2.11 Beep On Slave Negative Acknowledgment

BeepOnNak (Default=OFF)

Generate a host computer beep if the adapter returns a "Slave Not Acknowledging" response to the host computer. Slave Not Acknowledging occurs when the adapter is attempting to become a bus master and no slave device acknowledges the transmitted slave address.

4.2.12 Beep On /INT Assert

BeepOn/INT (on INT supported adapters only. Default=OFF)

Generate a host computer beep if the adapter returns an "/INT Signal Assert" response to the host computer. /INT Signal Assert occurs if /INT Signal Monitoring is enabled and a high to low transition is detected on the adapter /INT signal connector.

4.2.13 Load I^{2}C Message List File

AutoLoad=filename

AutoLoad="file name"
filename.iml
"file name.iml"

Automatically open file with extension .IML and load messages into Message Center spreadsheet.

4.2.14 Saved I^{2}C Message List File

AutoSave=filename
AutoSave="file name"

Automatically save message list to the specified file upon executing AutoExit. Use to save message data read from a slave transmitter device.

4.2.15 Auto Open

AutoOpen Auto Open Link to I^{2}C Adapter

Open link to the adapter.

4.2.16 Auto Send

AutoSend Auto Send I^{2}C Message List

Send I^{2}C messages loaded with the AutoLoad command.

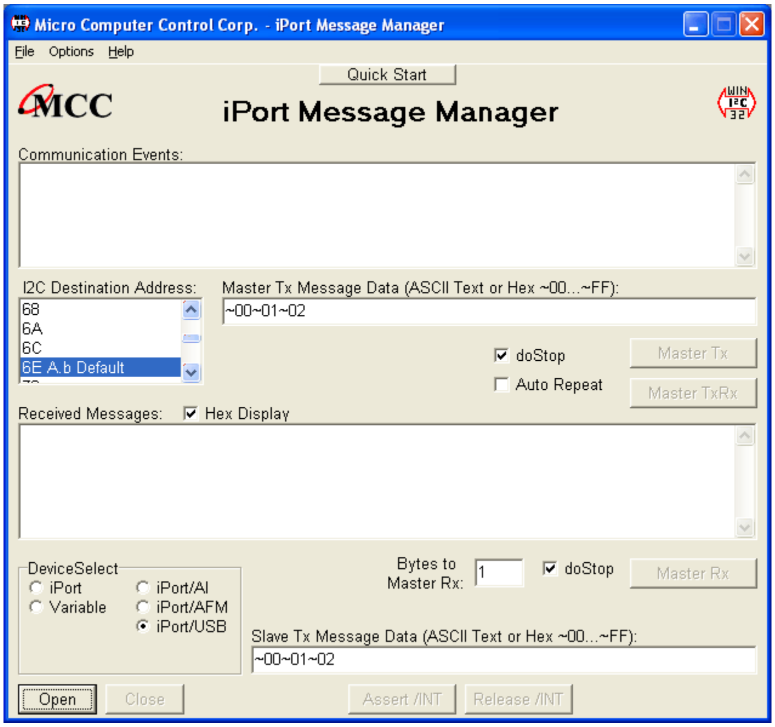
4.2.17 Auto Exit

AutoExit Auto exit after sending the message list.

Message Center will auto exit after sending the last message in the I^{2}C message list.

5 iPort Message Manager

The iPort Message Manager supports I^{2}C Master and Slave, Transmit and Receive activities for all MCC I^{2}C Bus host adapters, allowing a PC to become an I^{2}C Master or Slave device, transmitting or receiving I^{2}C messages between the PC and one or more I^{2}C devices across an I^{2}C Bus.



I2C Destination Address: 68, 6A, 6C, 6E A.b Default

Master Tx Message Data (ASCII Text or Hex ~00...~FF): ~00~01~02

☑ doStop

□ Auto Repeat

☑ Hex Display

DeviceSelect: ○ iPort ○ Variable ○ iPort/AI ○ iPort/AFM ● iPort/USB

Bytes to Master Rx: 1

☑ doStop

Slave Tx Message Data (ASCII Text or Hex ~00...~FF): ~00~01~02

Main Screen (Typical)

The Message Manager is designed to be a simple application for experimenting with I^{2}C messages. Message Manager provides methods to:

1. Set the I^{2}C adapter's own I^{2}C Slave address, General Call Enable, and other operating parameters.
2. Master Transmit ASCII text or Hex (00...FF) data to a specified I^{2}C Slave Receiver device.
3. Master Receive data from a specified I^{2}C Slave device.
4. Perform Master Read after Write operations.
5. Slave Transmit data to a requesting I^{2}C Master device.
6. Display Master or Slave Receive data in hexadecimal or ASCII.
7. Display I^{2}C Bus communication events.
8. Assert or release the INT signal (on supported adapters only).

5.1 Message Manager Operations

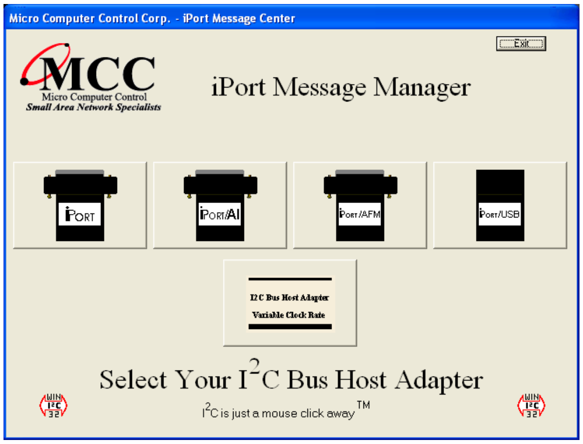
Communicating with another device on the I^{2}C Bus is easy. Just install the software as described in Section 3, then follow these simple steps:

5.1.1 Starting the Message Manager

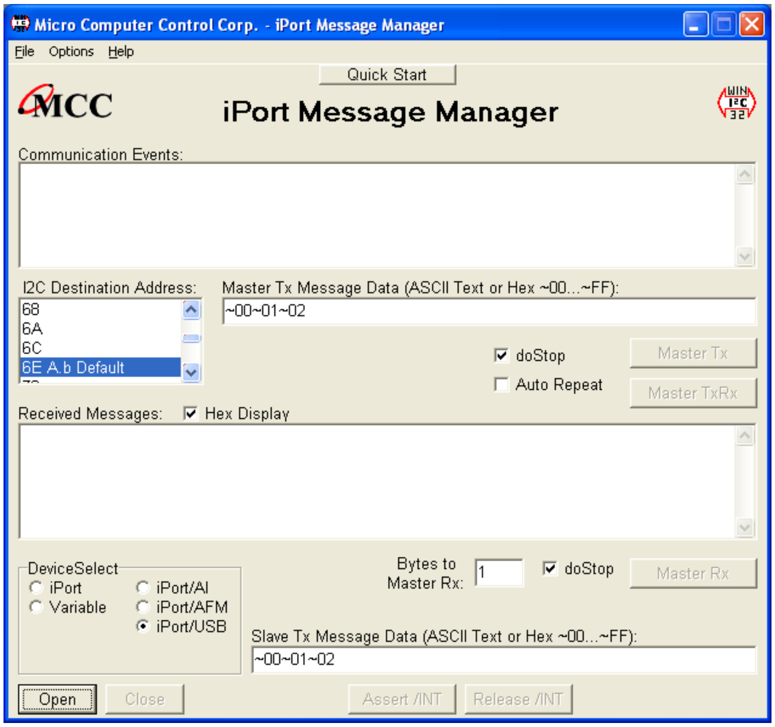
Click, Start | Programs | iPort Utility Pack | iPort Message Manager

5.1.2 Select the Adapter

Select the I^{2}C adapter you are using by clicking the corresponding adapter image (see Opening Screen), or the Device Select checkbox (see Main Screen).


Micro Computer Control Corp. - iPort Message Center
Exit
MCC
Micro Computer Control
Small Area Network Specialists
iPort Message Manager
iPort
iPort/AI
iPort/AFM
iPort/USB
I2C Bus Host Adapter
Variable Clock Rate
Select Your I2C Bus Host Adapter
I2C is just a mouse click away TM
WIN I2C 32

Opening Screen (Typical)



I2C Destination Address: 68, 6A, 6C, 6E A.b Default

Master Tx Message Data (ASCII Text or Hex ~00...~FF): ~00~01~02

☑ doStop

□ Auto Repeat

☑ Hex Display

Bytes to Master Rx: 1 ☑ doStop

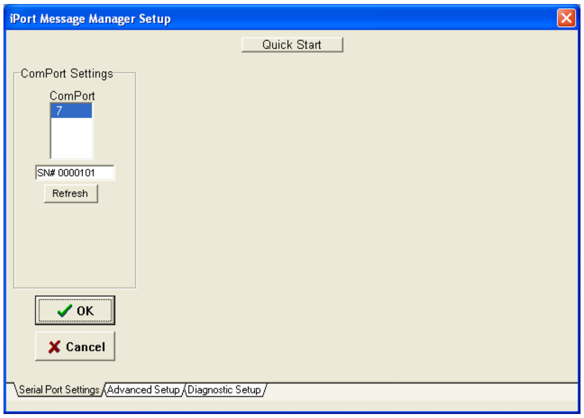
DeviceSelect: ○ iPort ○ Variable ○ iPort/AI ○ iPort/AFM ● iPort/USB

Slave Tx Message Data (ASCII Text or Hex ~00...~FF): ~00~01~02

Main Screen (Typical)

5.1.3 Establish Adapter Communications Link

On the main screen, click the Open button to view the Set Up Screen. Three levels of setup options are available, Basic, Advanced, and Diagnostic. Only Basic setup is required.



ComPort: 7

SN# 0000101

Basic Set Up Screen

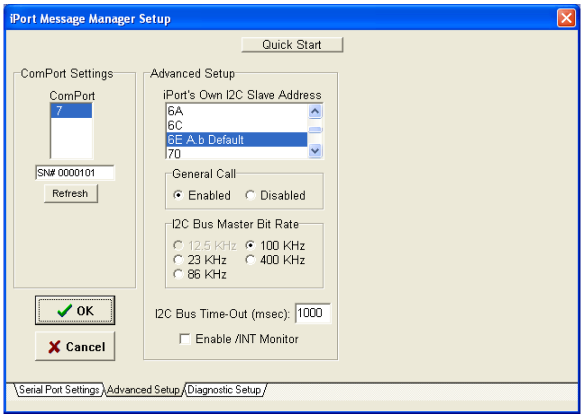
5.1.3.1 Basic Setup

Use the “ComPort” control to select the communication port connected to the I^{2}C adapter. The serial number for the selected adapter is displayed for supported adapters. In addition to RS-232, USB, and Ethernet adapters, Message Manager supports USB, network, Bluetooth connected adapters via the Windows Com driver.

Select from the list of available baud rates. Then click OK.

After a few moments, the Communication Events window on the Main Application screen should report “I^{2}C Open Successful.”

If open is not successful, follow the on-screen instructions. Make sure the communications port is working, is enabled in the Windows Device Manager, and is not being used by other software. Additional communication port open information is available in the log file. See Diagnostic Setup options.



ComPort Settings

ComPort: 7

SN# 0000101

Advanced Setup

iPort's Own I2C Slave Address: 6A / 6C / 6E A.b Default / 70

General Call: ● Enabled ○ Disabled

I2C Bus Master Bit Rate: ○ 12.5 KHz ● 100 KHz ○ 23 KHz ○ 400 KHz ○ 86 KHz

I2C Bus Time-Out (msec): 1000

□ Enable /INT Monitor

Advanced Set Up Screen

5.1.3.2 Advanced Setup

On the Advanced Setup screen you can set the following parameters:

Adapter's Own I^{2}C Slave Address

Select the I^{2}C adapter's own slave address. The adapter will acknowledge messages sent to this slave address. The default address is 0x6E.

General Call Enable

General Call Enable allows the I^{2}C adapter to respond as a slave receiver to the I^{2}C General Call Address (0x00). General Call is used by a master to broadcast an I^{2}C message to multiple devices. The default value is enabled.

I^{2}C Bus Master Bit Rate

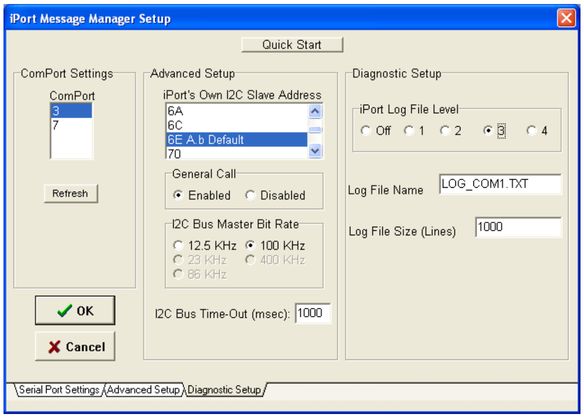
Select I^{2}C Bus speed during master operations. 100kHz is standard mode. 400kHz is fast mode. Available rates are I^{2}C adapter dependant.

I^{2}C Bus Time-Out

Specify how long the I^{2}C adapter will wait before reporting an I^{2}C Bus inter-byte time-out (0 = no time-out, 1 to 32767 milliseconds, iPort/AI fixed at 1 second).

Enable INT Signal Monitor (on supported adapters)

Enables monitoring of the INT signal state. INT state changes are reported in the main screen Communications Events window.



ComPort Settings

ComPort: 3 (selected), 7

Advanced Setup

iPort's Own I2C Slave Address: 6A, 6C, 6E A.b Default (selected), 70

General Call: ● Enabled ○ Disabled

I2C Bus Master Bit Rate: ○ 12.5 KHz ● 100 KHz ○ 23 KHz ○ 400 KHz ○ 86 KHz

I2C Bus Time-Out (msec): 1000

Diagnostic Setup

iPort Log File Level: ○ Off ○ 1 ○ 2 ● 3 ○ 4

Log File Name: LOG_COM1.TXT

Log File Size (Lines): 1000

Diagnostic Set Up Screen

5.1.3.3 Diagnostic Setup (on supported adapters)

On the Diagnostic Set-up screen you can set the following parameters:

Log File Level

A log file is available for troubleshooting communication problems between the host computer and the I^{2}C adapter. The log file is an ASCII text file viewable with any text editor. Select logging level. Level 1 provides minimum information. Level 4 provides maximum information.

Log File Name

Specify a log file name. Unless a path is specified, the log file will be created in the current working directory.

Log File Size

Specify log file length in lines. The log file overwrites earlier entries upon reaching the specified number on lines.

5.1.4 Sending I^{2}C Messages

5.1.4.1 Master Operations

5.1.4.1.1 Specifying the Destination Address

The Destination Address is the I^{2}C slave address of the slave device being addressed on the bus. All slave addresses are displayed as even numbers (00...FE), representing the 7 most significant bits of the 8-bit slave address transmitted on the bus (aaaa aaa0).

The I^{2}C adapter automatically supplies the 8th, least significant, Read/Write bit when it sends the slave address across the bus. For master write operations, the Read/Write bit is always transmitted as a logical 0 (aaaa aaa0). For master read operations, the Read/Write bit is always transmitted as a logical 1 (aaaa aaa1).

On the main screen, use the I^{2}C Destination Address list control to set the slave address of the slave device you want to address on the bus.

5.1.4.1.2 Repeated Start Messages

I^{2}C Bus communications support an operation called Repeated Start. In this operation, a message is sent across the bus beginning with a Start Condition, but without a Stop Condition at the end of the message. The next message sent across the bus begins with a Start Condition, in this case a Repeated Start.

An I^{2}C Bus master, that successfully sends a message on the bus, owns the bus until that master sends a message with a terminating Stop Condition. The Repeated Start operation allows the bus master to retain control of the bus while sending one or more messages on the bus. This prevents other bus masters, in a multi-master system, from accessing the bus and interfering with message sequences.

The Message Manager supports Repeated Starts with the doStop checkbox. Sending an I^{2}C message with doStop checked will cause the message to be terminated with a Stop Condition. Sending an I^{2}C message with doStop unchecked will cause the message to end without a Stop Condition, allowing the next message to be sent with a Repeated Start.

5.1.4.1.3 Auto Repeat

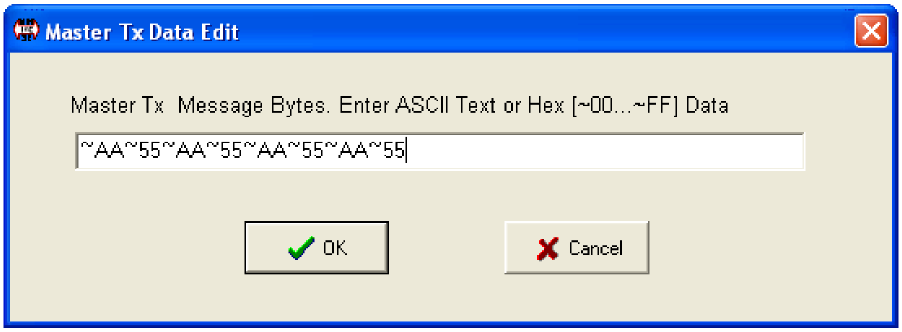
The situation often arises, where you would like to automatically repeat a master message operation.

The Message Manager supports auto-repeat with the Auto Repeat checkbox. You can automatically repeat a master operation by checking the Auto Repeat control before clicking the Master Tx, Master Rx, or Master TxRx buttons. The master operation repeats until the Auto Repeat control is unchecked.

5.1.4.1.4 Master Transmitting Data

Specifying Master Tx Message Bytes

Master Tx Message Bytes is the ASCII or Hexadecimal data you want to transmit to a slave receiver device. With the Message Manager, entering master transmit data is easy. On the main application screen, click on the Master Tx Message Bytes box to open the data editor.


Master Tx Data Edit

Master Tx Message Bytes. Enter ASCII Text or Hex [~00...~FF] Data

~AA~55~AA~55~AA~55~AA~55

In the data editor, enter one or more ASCII text characters or hexadecimal data bytes. Each hexadecimal byte is entered as two ASCII-Hex characters (00 to FF) preceded by a tilde (~) character. ASCII text and hex data can be intermixed, as long as each hex byte is preceded by a tilde.

For example, to enter hex data bytes 0x00, 0x01, and 0x02, enter the characters ~00~01~02 into the text box.

Each iPort Message Manager I^{2}C message can include up to 80 bytes of 8-bit ASCII binary data.

NOTE: The data you send may have special meaning to the receiving slave device, but to the Message Manager, and the I^{2}C adapter, message data has no special meaning. Consult your slave device's data sheet for details.

Click OK to accept the data.

Sending Master Transmit Messages

Click the Master Tx button to write the specified Master Tx Data Bytes to the selected destination slave device. If Auto Repeat is checked, the message will automatically repeat upon completion.

The Communications Events window on the main screen should report “Master Tx Complete.” If this message does not appear, check the slave device address, connections, and power.

If you get a “Slave Not Acknowledging” message in the Communications Events window, this could mean you have the wrong address in the I^2C Destination Address, or the device is not answering to its address. Consult your slave device’s data sheet for details.

5.1.4.1.5 Master Receive Data

Specifying Data to Read

On the lower part of the main screen, set the Bytes to MasterRx edit box to the number of bytes you want to read. For example: Set this to 1 to read a single byte. Click on the MasterRx button to read the data from the selected slave device.

Data received from the slave is displayed in the Received Messages text box on the main screen. The Communications Events window should report “Master Rx Transfer Complete.” If this message does not appear, check the slave device address, connections, and power.

If you get a “Slave Not Acknowledging” message in the Communications Events window, this could mean you have the wrong address in the I^2C Destination Address, or the device is not answering to its address. Consult your slave device’s data sheet for details.

Negative Acknowledge Last Byte

On supported adapters, the doNak checkbox gives you the option to acknowledge, or negatively acknowledge, the last byte read from a slave device. Some Slave Transmitter Devices require a negative acknowledgment on the final byte read from the slave device. I^{2}C adapters not supporting this option automatically negatively acknowledge the last byte read.

5.1.4.1.6 Master Transmit and Receive

The Master TxRx button sends a master write message with no Stop Condition, immediately followed by a Repeated Start master read message with Stop.

5.1.4.2 Slave Operations

In addition to performing I^{2}C Bus master operations, the Message Manager can also perform I^{2}C bus slave transmit and receive operations.

5.1.4.2.1 Slave Transmit Data

Slave transmit data is entered in the Slave Tx Message Bytes text box control on the main screen. Data in this text box is automatically sent to a requesting master upon receiving a slave transmit request.

Like Master Transmit data, Slave Transmit data is entered with the data editor. To enter data to be transmitted, click on the Slave Tx Message Bytes text box to open the data editor. See “Specifying Master Tx Message Bytes” section for data entry details.

5.1.4.2.2 Slave Receive Data

Data bytes received from a Master Transmitter are automatically displayed in the main application screen Received Message window. Received data is displayed in ASCII printable, or hexadecimal (~00 to ~FF) formats. Use the Hex-Display checkbox to force ASCII printable data to display in hexadecimal format .

6 Uninstalling Software Components

Software components include the iPort Utility Pack for Windows, and for USB-based adapters, the Virtual Communications Port (VCP) Device Driver. The following instructions can be used to remove either or both software components from your computer.

6.1 Uninstalling iPort Utility Pack for Windows

To uninstall the iPort Utility Pack for Windows software, use the Windows Control Panel “Programs and Features” (formerly “Add or Remove Programs”) utility. Note that uninstalling the iPort Utility Pack for Windows software does not uninstall the device or driver software.

6.2 Uninstalling VCP Device Driver

The VCP Device Driver can be uninstalled using Windows Device Manager (Start | Settings | Control Panel | System | Device Manager | Ports (COM & LPT)), or the Driver Uninstall short-cut on the iPort Utility Pack Start menu.

Revision Report

This section defines revisions and changes made to this document:

Revision: 1.00

1 Initial Release

Additional Information

For additional information on the I^{2}C Bus, please refer to the following:

“What is I^{2}C?”
www.mcc-us.com/I2CBusTechnicalOverview.pdf

“Frequently Asked Questions (FAQ)”
www.mcc-us.com/faq.htm

"The I^{2}C and How to Use It"
www.mcc-us.com/i2chowto.htm