



Connectware™



Digi Connect Customization Guide



Making
DEVICE NETWORKING
easy™

© Digi International Inc. 2003. All Rights Reserved

The Digi logo and RealPort are registered trademark of Digi International, Inc.

Connectware, Digi Connect ME and Digi Connect EM are trademarks of Digi International, Inc.

Information in this document is subject to change without notice and does not represent a commitment on the part of Digi International.

Digi provides this document “as is,” without warranty of any kind, either expressed or implied, including, but not limited to, the implied warranties of fitness or merchantability for a particular purpose. Digi may make improvements and/or changes in this manual or in the product(s) and/or the program(s) described in this manual at any time.

This product could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes may be incorporated in new editions of the publication.

Contents

About This Guide

Objective	7
Audience	7
Related Information.....	7

Chapter 1 Introducing the Customization Kit

Introduction	9
Contents of the Customization Kit	9
Customization Kit Sample Applications	10
Customization Kit Software Tools.....	11

Chapter 2 Communicating with My Device

Introduction	13
TCP Communication.....	14
UDP Communication	16
SSL/TLS Communication.....	17
RealPort COM Port Redirection.....	17
Alarm Notification over the Network.....	18

Chapter 3 Getting Familiar with the Technology

Introduction	19
Test Driving the TCP Client Sample	19
Test Driving the TCP Server Sample.....	20
Test Driving the UDP Serial Sample.....	21
Test Driving RealPort.....	22
Test Driving the SSL Client Sample.....	23

Chapter 4 Customizing Software

Introduction	25
TCP Client and TCP Server Sample Applications	25
UDP Sample Applications.....	25
SSL Client Sample Application	25
Device Discovery Sample Application	26
Browser Interface Sample Application	26
Uploading a Browser Interface Application	26
Resetting the Module's Configuration to Defaults.....	27
Updating the Firmware.....	28
Rebooting the Module.....	28

Index	29
--------------------	-----------

About This Guide

Objective

This manual will help you

- Get familiar with the Digi Connect ME or Digi Connect EM module and the Customization Kit
- Use sample applications and software tools to customize a network solution.

Audience

This manual is intended for anyone responsible for network-enabling or customizing a device using Digi Connect products.

Related Information

In addition to this manual, the Digi Connect Customization Kit includes a hardware reference for your platform, a user's guide, and several online html files that document sample applications.

Chapter 1 Introducing the Customization Kit

Introduction

The customization kit includes a CD with sample source code, documentation and software tools for Digi Connect platforms.

Contents of an Integration Kit

An integration kit contains the customization CD and an embedded development platform (Connect ME or Connect EM):

- A development board with power supply and international connectors.
- A wrist strap to protect the module and development board from electrostatic discharge (ESD)
- A loopback plug
- A standard Ethernet cable and a cross-over Ethernet cable
- A serial cross-over cable and a serial straight-through cable

Customization Kit Sample Applications

The customization kit contains several sample applications to aid customization efforts. Some applications demonstrate ways to communicate with a networked device. The Device Discovery application locates network-connected modules, enabling users to access and configure it. The configuration application enables users to configure and monitor the module. All can be easily customized to your company's "look and feel."

The kit contains the following samples:

- Three TCP client and three TCP server source code samples and compiled TCP client and TCP server versions to familiarize you with the technology
- Three UDP source code samples and a compiled version
- An SSL source code sample and a compiled version
- A Device Discovery source code sample and a compiled version
- A configuration application that enables users to configure and monitor the module. This application, written in Java, can be trimmed to expose only those parts of the configuration required by your users.

Customization Kit Software Tools

The customization kit includes the following software tools:

- The Digi RealPort driver, which provides COM port redirection
- Sun Java Run-Time Environment (JRE), which provides the Java browser plug-in required to run the Java-based configuration interface application. If your browser already has a Java plug-in, you do not need to install the JRE.
- Adobe Acrobat Reader, which is required to read integration kit online documents. If you already have the Acrobat Reader, installing it again is not required.
- Internet Explorer (IE) 6, which is included because the Java plug-in is required for browsers that access the module's Java-based web interface. If your browser supports the Java plug-in, installing this version is not required.

The JRE, Acrobat Reader, and IE 6 are not installed by default. To install them, choose the Advanced installation option.

Chapter 2 Communicating with My Device

Introduction

This chapter describes communication models, ways an application can use an Ethernet network to communicate with a device. Its purpose is to familiarize you with the technology.

The integration kit includes documented source code sample applications that demonstrate the various communication models described here. See the associated ReadMe.html file for information on each sample application.

One option, RealPort COM Port redirection, requires no development work. The driver is included in the kit and works once set up procedures are completed.

TCP Communication

The TCP client and TCP server options use the TCP transport service, a connection-oriented service providing reliable network transfer. The service is connection-oriented in that TCP entities at both ends set up a virtual connection prior to data transfer and maintain it until the transfer is complete. This virtual connection is what enables reliable service, the guarantee that everything that is sent is received, and received in the correct order.

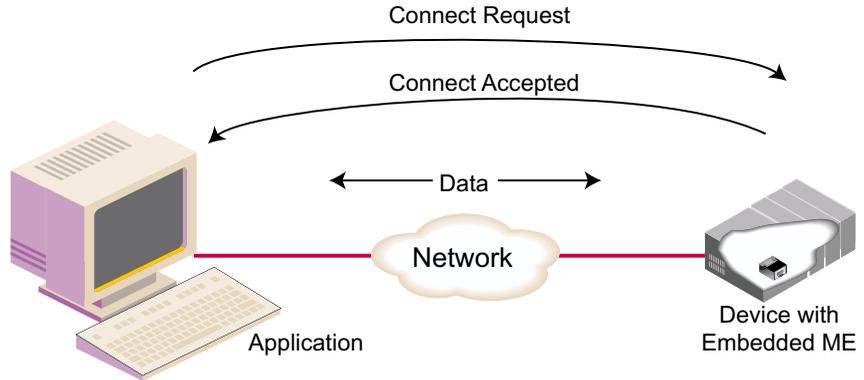
TCP guarantees delivery by requiring acknowledgment of transmitted data. If an acknowledgment is not received within a specified period, data is retransmitted.

Reliability is a key TCP feature, an attribute that requires slightly more network overhead than does UDP, a connectionless transport service described later in this chapter. Telnet applications require TCP transport.

TCP Client Option

With this option, the application initiates communication with a device. The device waits for incoming connections and then can communicate bidirectionally with the application.

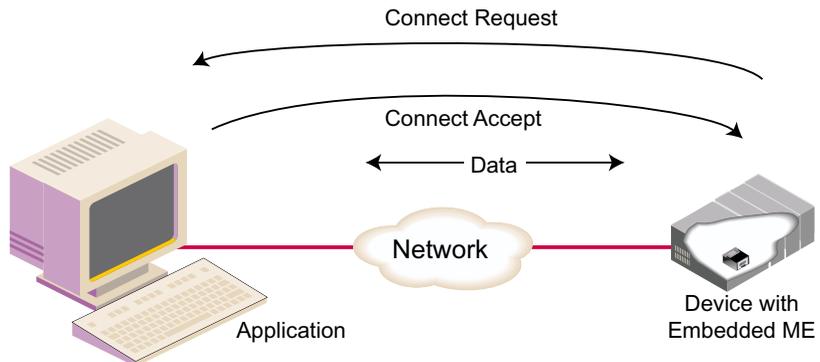
In this model, the application manages network connections and keeps track of device IP addresses. This model is best used when the application needs to manage or notify users of network outages and communication failures. The following figure depicts the TCP client option.



TCP Server Option

With this option, the application waits for incoming communication initiated by the device.

Again, communication is bidirectional. This model works well when the application does not or cannot manage a list of device IP addresses or when the set of devices on the network changes frequently. An example is where extra registers or curb-side baggage check stations are added or changed frequently.



UDP Communication

This option uses the UDP transport service, a connectionless service that does not guarantee packet delivery. It is connectionless in that packets are simply transmitted to the other side; no attempt is made to establish an ongoing connection to ensure that the other side is powered on or connected to the network, that all packets reach the destination, or that packets are not repeated or received out of order.

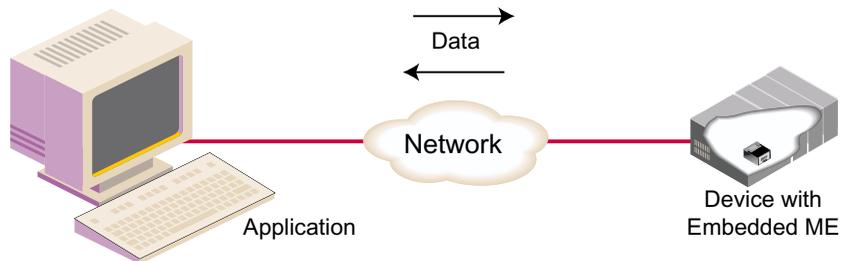
The following are two key UDP advantages:

- The connectionless service produces little network overhead.
- A single communication can be sent to multiple recipients.

This application is best used when communication with the device is “packet oriented” or structured with commands and response.

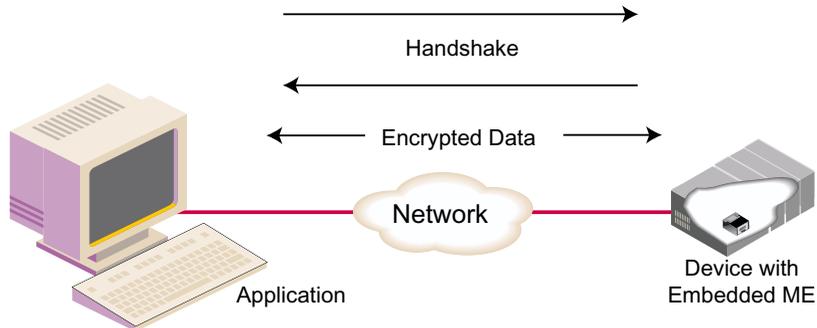
UDP Options

With these options, the application can initiate network communication with a device connected to the module’s serial port or it can simply wait for incoming communication from the device. The following figure depicts this option.



SSL/TLS Communication

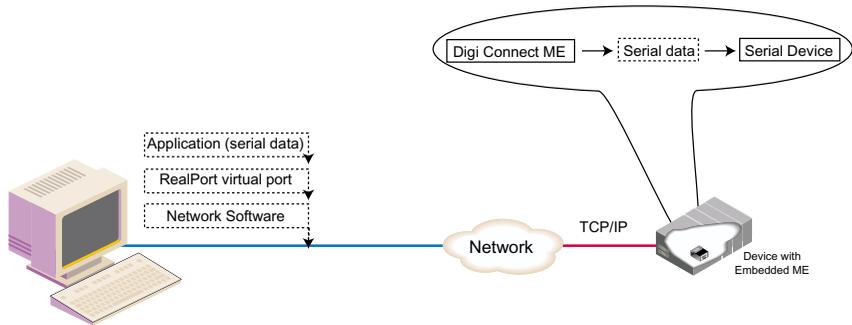
With this option, an application initiates SSL/TLS communication over the TCP transport service. The client and server exchange SSL handshakes, determining protocol version and public and private keys. Once connected, all data passed between the application and device is encrypted. The figure illustrates this option.



RealPort COM Port Redirection

With this option, an application designed to communicate with a serial device using a COM port simply opens a RealPort virtual COM port that directs the data across the network to your device.

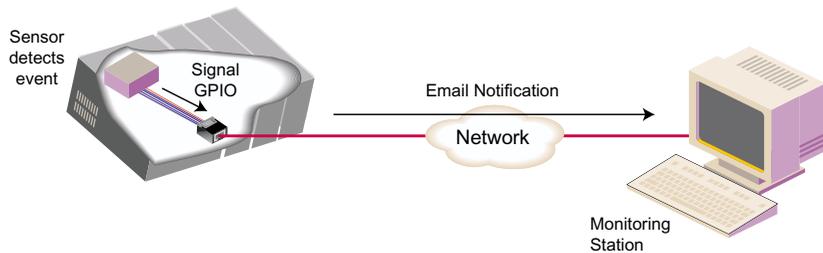
The advantage of this option is that neither the application nor the serial device need to be changed from their original serial communication model to work on a network. Both continue to use serial communication techniques, and neither is aware of the intervening network. The following figure depicts this option.



Alarm Notification over the Network

With this option, the device uses the Digi Connect ME GPIO ports to signal a status change, an event that triggers an email notification to a monitoring station.

The following figure illustrates this feature. A sensor within the device uses one of the module's GPIO ports to signal that an event has occurred. The module then sends email notification to a monitoring station across the network.



Introduction

This chapter provides procedures that enable you to see the Digi Connect product and related applications in action. It describes the steps required for setup and shows you how to use the applications.

Test Driving the TCP Client Sample**Introduction**

The TCP client application sets up a TCP socket connection to the module and sends the “Hello World” text string, which is echoed back in the application window.

Prerequisite Tasks

Before you begin, complete the quick start setup described in the *Digi Connect User’s Guide*.

Procedure

1. Access the web interface by entering the module’s IP address in a browser’s URL window.
2. Choose Serial Ports from the Configuration menu.
3. Configure the module to function as a TCP server by doing the following:
 - a. Click the TCP tab.
 - b. Check Enable TCP Server.

- c. Specify 2101 as the Raw TCP port.
- d. Click Save.
4. Connect the loopback plug to port 1 of the development board.
5. Start the TCP client application by choosing Start > Programs > Digi Connect > Samples > TCP Client.
6. At the prompt supply the IP address assigned to the module and TCP port 2101.

Note: If you have forgotten the IP address, use device discovery to locate the module and learn its IP address.
7. Watch the exchange of text strings between the application and the module.

Test Driving the TCP Server Sample

Introduction

The TCP server application waits for an incoming TCP connection and then displays the “Hello World” text in the application window.

Prerequisite Tasks

Before you begin, complete the quick start setup described in the *Digi Connect User's Guide*.

Procedure

1. Start the TCP server application by choosing Start > Programs > Digi Connect > Samples > TCP Server.
2. Access the module's web interface by entering its IP address in a browser's URL window.
3. Choose Serial Ports from the Configuration menu.
4. Configure the module to function as a TCP client by doing the following:

- a. Click the TCP tab.
 - b. Check Automatically establish TCP connections.
 - c. Specify DCD in the Connect field.
 - d. Enter the IP address of the PC running the sample in the Connect To field.
 - e. Specify Raw TCP as the Service Type.
 - f. Specify 7777 as the TCP port number.
 - g. Click Save.
5. Connect the loopback plug to port 1 of the development board.
 6. Return to the TCP server application to monitor the exchange of text strings.

Test Driving the UDP Serial Sample

Introduction

The UDP serial sample displays the “Hello World” string. Unlike the previous examples, it uses UDP transport, not TCP, and the application performs both client and server functions.

Prerequisite Tasks

Before you begin, complete the quick start setup described in the *Digi Connect User’s Guide*.

Procedure

1. Access the web interface by entering the module’s IP address in a browser’s URL window.
2. Choose Serial Ports from the Configuration menu.
3. Configure the module as a UDP server by doing the following:
 - a. Click the UDP tab.
 - b. Check Enable UDP Server.
 - c. Specify 2101 as the Raw UDP port.

- d. Click Save.
4. Configure the module as a UDP client by doing the following:
 - a. Click Enable UDP Client.
 - b. Configure a destination by doing the following (1) Supplying the IP address of the PC on which the sample application is installed. (2) Supplying a UDP port of 7777. (3) Ensuring that the Enabled field is checked.
 - c. Click Save.
5. Connect the loopback plug to port 1 of the development board.
6. Start the UDP application by choosing Start > Programs > Digi Connect > Samples > UDP Serial.
7. At the prompt, supply the module's IP address and a UDP port number of 2101.
8. Watch the exchange of text strings between the application and the module.

Test Driving RealPort

Introduction

Test driving RealPort involves installing the driver and then launching a HyperTerminal connection on a RealPort virtual port.

Prerequisite Tasks

Before you begin, complete the set up tasks described in the *Digi Connect User's Guide*.

Procedure: Installing RealPort

Use this procedure to install RealPort software on the same computer that the integration kit software and tools were installed.

1. From the Start Menu, choose Programs > Digi Connect > RealPort > InstallRealPortXxx, here Xxx represents a version or versions of Microsoft Windows.
2. Follow the on-screen prompts and use the online Help to complete installation. During installation, RealPort virtual COM ports will be installed on your system.

Using HyperTerminal to Test RealPort

1. Start a HyperTerminal session on a RealPort virtual port. This port is the first one after standard COM ports. So, if you have two standard COM ports, this will be COM3.
2. Type characters and watch them echo back to the screen.

Test Driving the SSL Client Sample

Introduction

The SSL client sample application initiates communication with the module by creating a secure network socket and then establishing a TCP connection to the module's port.

Prerequisite Tasks

Before you begin, complete the quick start setup described in the *Digi Connect User's Guide*.

Procedure

1. Access the web interface by entering the module's IP address in a browser's URL window.
2. Choose Serial Ports from the Configuration menu.
3. Configure the module to function as a TCP server by doing the following:
 - a. Click the TCP tab.
 - b. Check Enable TCP Server.
 - c. Specify 2601 as the Raw TCP port.
 - d. Click Save.

4. Put the loopback plug in port 1 of the development board.
5. Start the SSL client application by choosing Start > Programs > Digi Connect > Samples > SSL Client.

A command line window appears.

6. Type the following:

```
webclient -sip-address -p2601
```

where *ip-address* is the module's IP address and 2601 is the TCP port number.

7. Watch the exchange of text strings between the application and the module.

Introduction

This chapter provides information on sample applications and on Digi Connect tools required to customize the browser interface and maintain a customized product.

TCP Client and TCP Server Sample Applications

The customization kit has three TCP client and three TCP server sample applications. It includes C++ samples that use the Windows Socket Architecture (WSA), C++ samples that use the Microsoft Foundation Classes (MFC) architecture, and C samples that use the Berkeley Software Design (BSD) socket model. Access application files by choosing Start > Programs > Digi Connect > Samples.

UDP Sample Applications

The customization kit has three UDP sample applications. It includes a C++ WSA sample, a C++ MFC sample, and a C BSD socket sample. Access application files by choosing Start > Programs > Digi Connect > Samples > UDP.

SSL Client Sample Application

The customization kit has a C++ WSA sample application. Access application files by choosing Start > Programs > Digi Connect > Samples > SSL Client.

Device Discovery Sample Application

The Device Discovery sample application is written in C++ and uses the Microsoft Foundation Classes (MFC) library. Access application files by choosing Start > Programs > Digi Connect > Samples > Device Discovery.

Browser Interface Sample Application

The sample browser interface application consists of two files. The applet code base is in configME.jar and is launched by index.htm.

If you used defaults to install the integration kit, application files can be found in the following folder:

```
C:\Program Files\Digi Connect Customization  
Kit\Factory_Defaults\Connect_ME
```

Uploading a Browser Interface Application

Introduction

This section describes how to upload a browser interface application to replace the module's default application. You will use this procedure to do the following:

- Replace the default application with your customized application
- Restore the default application if required

Procedure

1. Access the administration page by entering the following in a browser's URL window:

```
ip-address/admin/administration.htm
```

where *ip-address* is the module's IP address.

Note: It is administration.htm, not html.

2. Choose File Management from the menu.
3. Do one of the following:

- If you are uploading a customized version of the configME.jar file, browse to the file and choose Upload.
 - If you are restoring the default interface application, browse to the configME.jar file and choose Upload. If you installed this in the default locations, it is in C:\Program Files\Digi Connect Customization Kit\Factory_Defaults\Connect_ME.
4. Repeat these steps for the index.htm file if you have made changes to the default file.

Resetting the Module's Configuration to Defaults

Introduction

This section describes how to restore the configuration to Digi factory defaults.

Procedure

1. Access the administration page by entering the following in a browser's URL window:
ip-address/admin/administration.htm
where *ip-address* is the module's IP address.
Note: It is *administration.htm*, not *html*.
2. Choose Factory Default Settings from the menu.
3. Check the keep IP address settings box appropriately and then click the Restore button.

Updating the Firmware

Introduction

This section describes how to update firmware.

Procedure

1. Download a firmware file from the Digi web site.
2. Access the administration page by entering the following in a browser's URL window:

ip-address/admin/administration.htm

where *ip-address* is the module's IP address.

Note: It is [administration.htm](#), not html.

3. Choose Update Firmware from the menu.
4. Browse to the downloaded firmware file.
5. Choose Update.

Rebooting the Module

Introduction

This section describes how to reboot the module.

Procedure

1. Access the administration page by entering the following in a browser's URL window:

ip-address/admin/administration.htm

where *ip-address* is the module's IP address.

Note: It is [administration.htm](#), not html.

2. Choose Reboot from the menu.
3. At the prompt, click the Reboot button.

B

browser interface application
 customizing 26
 restoring the default 26
 uploading to the device 26

C

COM port redirection 17
configuration defaults, resetting 27

D

default configuration, resetting 27
device discovery
 customizing the sample 26
device monitoring 18
documentation, related 7

F

firmware, updating 28

I

integration kit contents 9

R

RealPort
 description 17
 test driving the sample 22
rebooting the module 28

related documentation 7
resetting the default configuration 27

S

sample applications 9
software tools 11
 Sun JRE 11
SSL
 communication option 17
 customizing the sample 25
 test driving the sample 23
SSL/TLS option 17

T

TCP client
 communication option 14
 customizing the sample 25
 test driving the sample 19
TCP server
 communication option 15
 customizing 25
 test driving the sample 20

U

UDP
 communication options 16
 customizing the sample 25
 test driving the sample 21



Connectware™

Digi International

**11001 Bren Road East
Minnetonka, MN 55343
U.S.A**

952-912-3444

www.digi.com



PN:(1P) 9000321 A