IMPACC-Modbus Gateway

Standalone model
Version 6.40

Limits of Liability and Disclaimer of Warranty

The authors and publisher of this manual have used their best efforts in preparing this manual and the programs contained in it. These efforts include the development, research, and testing of the theories and programs to determine their effectiveness. The authors and publisher make no warranty of any kind, express or implied, with regard to these programs or the documentation contained in this manual. The authors and publisher shall not be liable in any event for incidental or consequential damages in connection with, or arising out of, the furnishing, performance, or use of these programs or this documentation.

If further information is desired by purchaser regarding the installation, operation or maintenance of the software described in this manual, the local Eaton Corporation representative should be contacted.

ALL WARRANTIES OF MERCHANTABILITY AND FITNESS FOR PARTICULAR PURPOSE ARE DISCLAIMED.

Trademarks

IMPACC is a trademark of the Eaton Corporation

Microsoft Windows® is a trademark of Microsoft Corporation.

Modbus® is a trademark of Modicon, Inc.

Eaton Corporation
Westinghouse and Cutler-Hammer Products
Pittsburgh, PA 15220

This manual corresponds with Version 6.40 of the IMPACC-Modbus Gateway.

Chapter 1

Overview

This chapter contains an overview of the IMPACC-Modbus Gateway, including:

- A description of how the IMPACC-Modbus Gateway works
- A list of the Modbus protocol functions that are supported
- An explanation of how communication occurs
- Examples of how the IMPACC-Modbus Gateway might be used
- A description of registers

What Is the IMPACC-Modbus Gateway?

IMPACC is Cutler-Hammer's system of electrical metering and protection devices and the software used to monitor and control those devices. *Modbus* is a communications protocol that stipulates how data is transmitted across a serial bus.

To permit IMPACC devices to be integrated into industrial communications and distributed control systems, the IMPACC-Modbus Gateway supports a subset of the Modbus protocol.

The IMPACC-Modbus Gateway enables a system that uses the Modbus protocol to communicate with IMPACC devices. It permits your Modbus-based system to:

- Receive data from IMPACC devices. For example, you might request the value of the Phase A Current from an IQ Analyzer.
- Send commands to IMPACC devices. For example, you might send a command to close a device.

How Does It Work?

Your Modbus-based system acts as the *master* by requesting information and sending commands. The IMPACC-Modbus Gateway system acts as the *slave* by providing that information and passing commands to the specified device.

The Modbus Gateway is a slave because it *responds* to a command or a request for data, but cannot *initiate* a command or a request for data. It passes commands from your Modbus-based system to IMPACC devices, and collects data from the IMPACC devices and returns it to your Modbus-based system.

You designate what data is passed to the Modbus Master by "mapping" IMPACC data fields to particular Modbus input registers. These registers store the data as it is received from IMPACC devices. By configuring the application in this way, you can control what data is collected from devices and where it is stored in the Modbus registers.

Supported Modbus Functions

The Modbus protocol provides numerous functions. The Modbus Gateway application supports a subset of these functions. The following table lists the functions that the Modbus Gateway supports and general information about each.

Modbus Type	584/984 Address	Point Type/Access	Modbus Function
Input Register	30,001–38,000 (decimal)	16-bit integer, read only	4
Holding Register	40,001–40,200 (decimal)	16-bit integer, read/write	3, 6
Holding Register	40,201–42,000 (decimal)	16-bit integer, read/write	Reserved for future use
Loopback Test	Diagnostic Code 0	16-bit integer	8

Refer to the Modicon Modbus Protocol Reference Guide for more information about the Modbus protocol and functions. Examples are provided in Appendix B.

Communication

The Modbus Master and the IMPACC-Modbus Gateway communicate via an RS232 communication channel (using RTU transmission mode). Data is transmitted using the Modbus protocol.

The IMPACC-Modbus Gateway and devices communicate via an INCOM network. The INCOM protocol is used.

Examples

For example, assume your company has a PLC that uses the Modbus protocol. When the PLC issues a command (for example, to reset a device), the command is passed across a serial communication cable to the Modbus Gateway application, then the Modbus Gateway application interprets the command and passes it to the appropriate IMPACC device to be carried out.

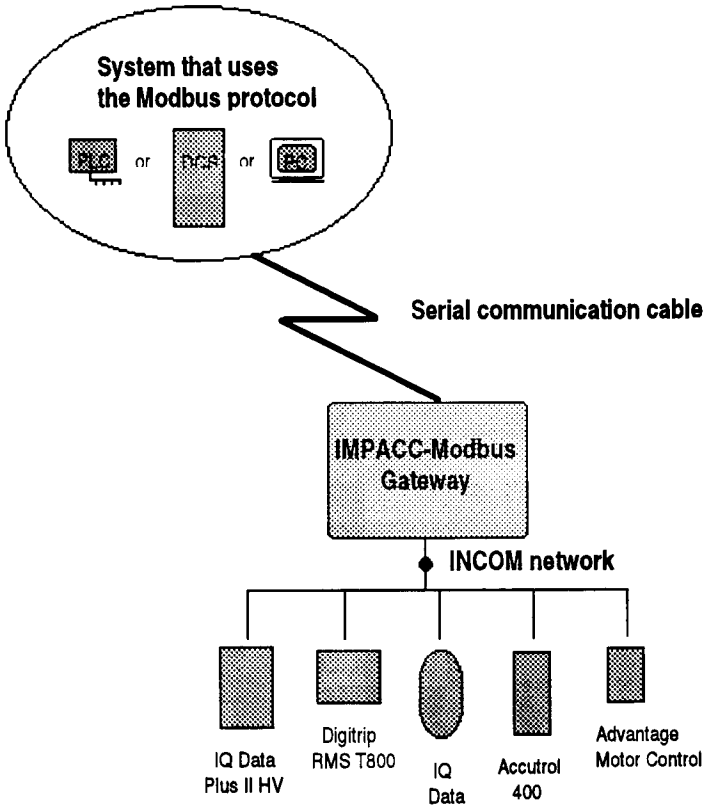
Data is passed through input registers. Commands are passed through holding registers. Refer to the "Defining Input Registers" section on page 48 and the "Defining Holding Registers" section on page 52 for more information about registers.

The IMPACC-Modbus Gateway is an interface between IMPACC and a system that uses the Modbus protocol. The following diagram shows this relationship.

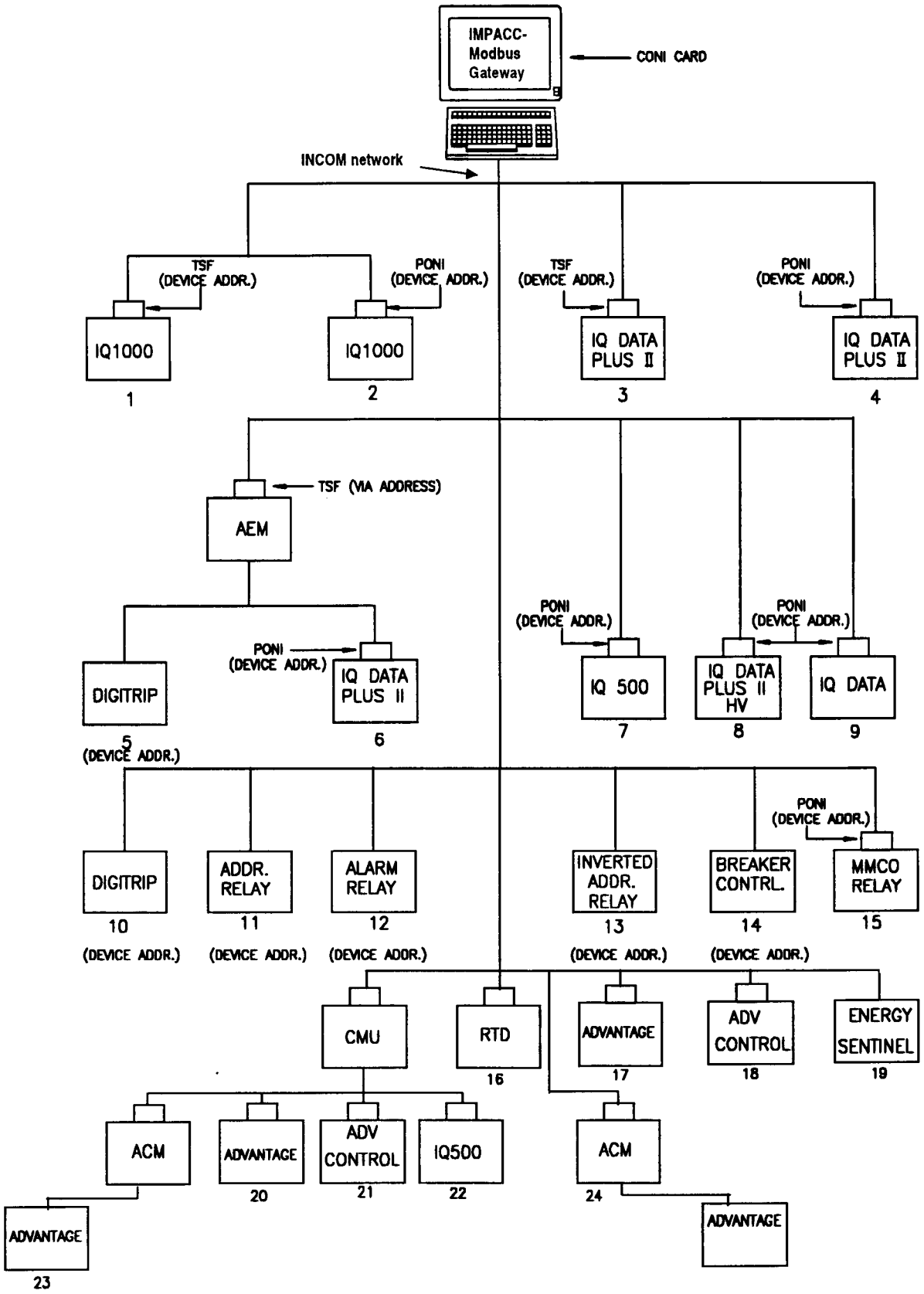
The system that uses the Modbus protocol is the *master*. For example, this might be a Programmable Logic Controller, a Distributed Control System, or a PC used in a Building Automation System. It receives data from IMPACC devices and sends commands that control the devices.

By mapping IMPACC data to specific locations (the Modbus register locations you define), the IMPACC-Modbus Gateway enables the master to communicate with IMPACC devices.

The Modbus Gateway system is the *slave*. It responds to requests from the master.



The following diagram shows a detailed example of communication between the Modbus Gateway and devices.



Modbus Registers

A register is a storage area for data. The IMPACC-Modbus Gateway uses two types of registers to store data and commands.

Input registers store data collected from IMPACC devices. The Modbus Gateway provides 8000 input registers, each represented by a unique number from 30,001 to 38,000.

Holding registers store commands that are to be sent to IMPACC devices. The Modbus Gateway provides 200 holding registers, each represented by a unique number between 40,001 and 40,200.

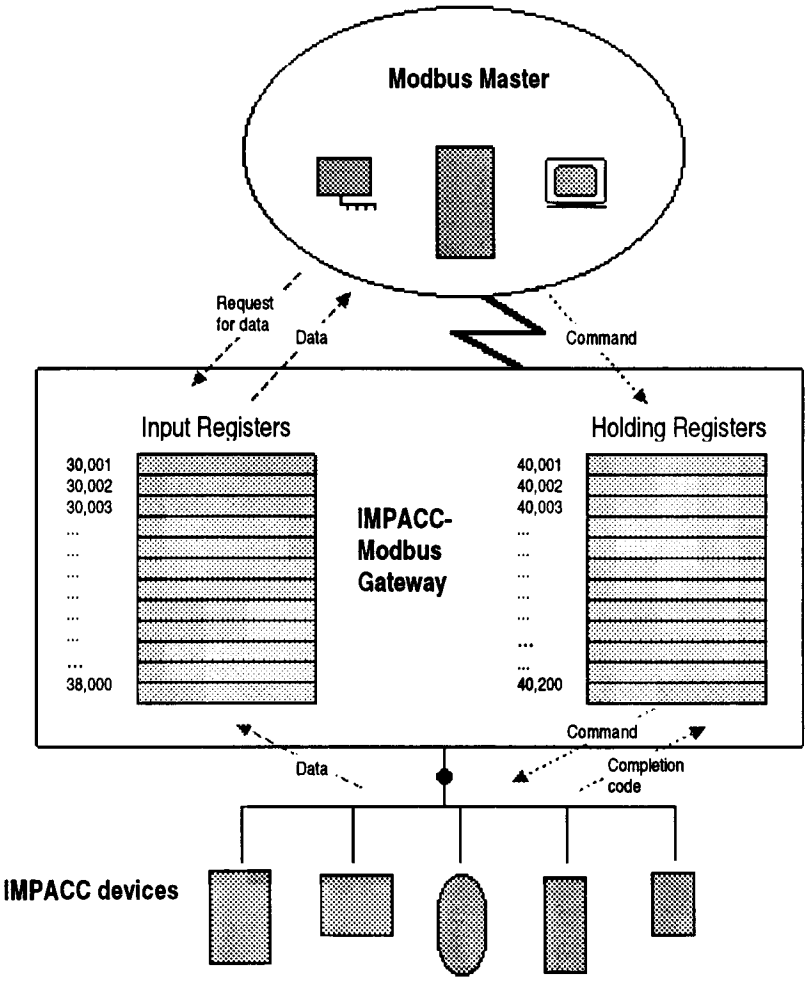
For information about configuring registers, refer to the "Defining Input Registers" section on page 48 and the "Defining Holding Registers" section on page 52.

The following illustration shows how information flows through the Modbus Gateway and the role that registers play.

The Modbus Master (which might be a PLC, a DCS, etc.) requests data.

The data is retrieved and returned to the master. For example, register 30,003 might store the value of the "VCOIL" (coil voltage) field of device #20. The coil voltage value ("162 volts") is returned to the master.

Data from the device is stored in an input register.



The Modbus Master sends a command intended for a particular device. For example, it might send the command "0x600" to device #102.

The command is received by the Modbus Holding register that is mapped to the device. For example, if register 40,003 is mapped to device #102, register 40,003 will receive the command and then relay it to the device.

The device performs the command. When finished, the device writes a code to the holding register indicating that the task is complete or that an error occurred.