

The Modbus Device Module operates within the Universal Server architecture and provides comprehensive support for devices using Modbus communications protocols, including the Omni, Enron, and ScadaPack Modules. The Modbus Module permits the user to create either standard or user-defined Modbus devices to match virtually any field device using either Modbus ASCII, Modbus RTU, or Modbus TCP protocols for real-time communications. The ScadaPack Module permits the user to create either standard or user-defined devices to match Control Microsystems devices using Enron Modbus or Telebus protocols for real-time communications, including the 4202GFC. In addition, the Modbus module provides full support for collection of Electronic Flow Measurement (EFM) data using industry standard Enron RTU or Omni Archive methods.

Device Features

Each Modbus device includes features common to all devices within the Universal Server Architecture:

- Full support for OPC 1.0 and 2.05, SuiteLink, and DDE clients, in any combination.
- Each device may be configured for both a primary and secondary circuit for communications using any of the available circuits defined in the Universal Server. Each device may be configured to use the primary circuit only, secondary circuit only, or alternate between the two. Options also include automatic fail-over between the circuits.
- Available circuits may include direct serial, radios, dialup, CDPD, TCP/IP, UDP/IP, and terminal server connections.
- Each device includes a complete set of standard diagnostic items that may be viewed using the Universal Server remote diagnostics viewer or added to any OPC, SuiteLink, or DDE client.
- Devices may be configured to perform data acquisition automatically, only when clients are connected, or upon demand by client programs only.
- Command items allow complete control of devices from connected clients. Command items include setting device scan status, selecting active circuits, and requesting immediate demand scans.
- User-selectable log options provide comprehensive log messages for system monitoring and troubleshooting. The **Device Diagnostics** screen, below, performs interactive diagnostics and displays streaming device transactions and activity.

Device State:
Comm Status:

Parameter Name	Current	Today	Description
Last Update	03/09/04 12:23:03	03/09/04 12:23:03	Time of Last Diagnostic Update
Last Reset	03/09/04 12:23:03	03/09/04 12:23:03	Time of Last Diagnostic Reset
Total Transactions	0	0	Total Scan Attempts since reset
Good Scans	0	0	Total Successful Scans
% Throughput	0.0	0.0	Percent Throughput 0.0 - 100.0 %
Failed Scans	0	0	Total Failed Scans for this device
Good Outputs	0	0	Total Command Output Transactions
Failed Outputs	0	0	Total Failed Command Outputs
Retries	0	0	Total Number of Transaction Retries
Circuit Failures	0	0	Total Communications Failures
No Connections	0	0	Failed Connection Attempts
Lost Connections	0	0	Lost Connections
Device Timeouts	0	0	Device Timeouts (No Reply)
Invalid Replies	0	0	Invalid Device Responses (Bad CRC's)
Error Responses	0	0	Device Replied with Error Response
Requested Scan Interval	0	0	Current Requested Scan Interval (msec)
Last Scan Interval	0	0	Most Recent Scan Interval (msec)
Max Scan Interval	0	0	Maximum Scan Interval (msec)
Avg Scan Interval	0	0	Average Scan Interval (msec)
Scan Duration	0	0	Last Scan Duration (msec)
Item Count	0	0	Total Client Device Items
Active Items	0	0	Total Active Client Items

Status:
Update Interval: secs

Last Error:
Last Good Poll:

Real-Time Features

The Modbus Module includes several standard Modbus Device types, such as Modicon 484, 584, and 984 PLCs, Enron and Daniel 2500 RTUs, Omni RTUs, ScadaPack Modules (including support for Control Microsystems ScadaPacks with Enron and Telebus support) and others. In addition, the user may define and configure additional Modbus devices using the interactive editor. Once created, user-defined devices may be selected and used in exactly the same way as standard device types. Standard features of the Modbus Module include:

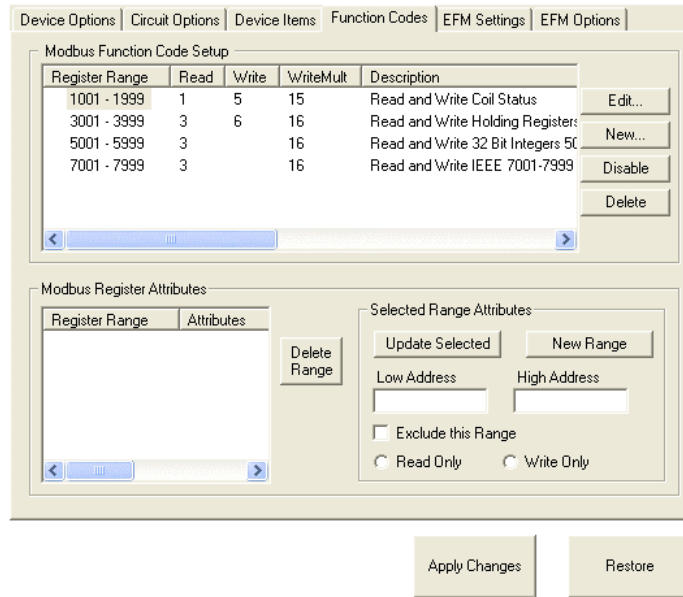
- Interactive wizards to create tag names for Modbus address items for client access. All tag names are available in any OPC Browser.
- Client programs may access Modbus items by direct Modbus address or with user-defined tag names.
- The tag creation process permits custom conversion of data acquired from the device, including data type conversion (integer, float, string, etc), engineering units scaling, byte and word ordering, and special bit test operations.
- Supports in-line modifiers for direct Modbus address items to allow client programs to specify a full range of custom conversions on each Modbus address.
- Tag items may be designated as autoscan to force data acquisition of the item upon startup with no client connected.
- Real-time acquisition may be set to occur at any time period, at a specific time of day, or upon client demand only.
- Optimized device scanning to minimize the number of messages required for data acquisition for current active items.
- Automatic retries of failed device IO transactions with user-configurable retries and response delays.
- Prioritized processing of output commands and demand read transactions.
- Supports full import/export of single device or all Modbus devices to/from CSV format to permit creation of large systems or to make global configuration changes.

User-Defined Device Support

The Modbus Module may be configured to support real-time communications with any device that supports the Modbus Protocol using the on-line interactive device editor. Features of this editor include:

- User may define or modify Modbus function codes used to read or write Modbus address ranges within the field device. Each address range allows unique function code used to read data, write a single address and write multiple addresses. The user may employ standard function codes or define a custom code supported by the field device.
- User may define exclusion ranges within a Modbus address range to designate addresses as excluded, read-only, or write-only.
- Each address range may be configured for custom conversion including native field data type (coils, integer, float, string, etc.), server data type (any OPC data type), bytes per register, byte ordering, word ordering, and all other parameters used by Modbus-compliant devices.

Once a new device has been created using the register/function code editor, the user may save the device as a user-defined device type. The user has the option of saving both the register/function code setup and any tag names created using the device item editor. User-defined devices may be used in exactly the same way as standard Modbus devices. The **Function Codes** screen for the Modbus Module is displayed below.



Electronic Flow Measurement

The Modbus Device Module includes standard support for automatically uploading EFM data from field devices that support either the Enron, Omni, or Telebus formats. All uploaded EFM data is forwarded to the EFM Server for storage and reporting. Refer to the Electronic Flow Measurement (EFM) datasheet for information concerning EFM data storage and reporting.

Specifications

<p><u>Hardware Requirements</u></p> <p>Minimum:</p> <ul style="list-style-type: none"> • PIII 900 MHz Processor • 100 MB Hard Disk • 64 MB RAM 	<p>Recommended:</p> <ul style="list-style-type: none"> • P4 1.4 GHz Processor • 500 MB Hard Disk • 128 MB RAM 	<p><u>Software Requirements</u></p> <ul style="list-style-type: none"> • Windows 2000 SP4 or greater • Windows XP SP1 or greater
---	---	---



MTL Open System Technologies LP

2450 South Shore Blvd., Suite 210
 League City, TX 77573
 Phone: (281) 334-9111 Fax: (281) 334-4324
info@mtlmost.com
www.mtlmost.com