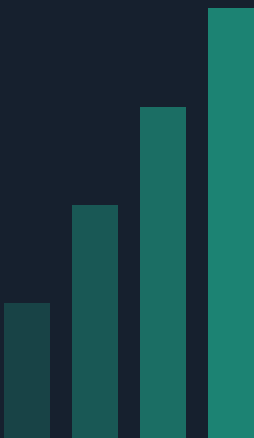


Modbus Manager Pro

User Manual

Modbus polling, simulation and HMI dashboards for industrial automation

Software version v1.4.0 • modbusmanager.com



Contents

1 Introduction

2 Installation and First Run

3 The Interface at a Glance

4 Connecting to a Device

TCP/IP, RTU and ASCII

Gateways and Modbus Security

5 Polling Registers

Function codes and addressing

Data formats and scaling

6 Writing Values

7 Charts and Live Monitoring

8 Diagnostic Tools

Slave Scan

Address Scan

Serial Monitor

Test Center

9 Slave Mode (Simulator)

10 Tags and Structured Text

Register vs internal tags

The script loop

11 Dashboard / HMI Builder

Pages and runtime

Element reference

Binding to registers or tags

Charts group and the channel editor

12 Workspaces

13 Licensing and Trial

14 Languages

15 Troubleshooting and FAQ

Appendix A Modbus quick reference

1. Introduction

Modbus Manager Pro is a desktop application for communicating with Modbus devices — PLCs, variable-speed drives, energy meters, inverters and any other equipment that speaks Modbus. It combines a fast register poller, a built-in slave simulator, a tag engine with Structured Text scripting, and a full drag-and-drop HMI dashboard in a single tool.

How Pro differs from Standard

The Standard edition covers everything you need for day-to-day polling, writing, scanning and the built-in slave simulator. The **Pro** edition adds the complete **Dashboard / HMI builder**: multi-page operator screens with gauges, trends, process graphics and live charts that you can run full-screen as a small SCADA-style front end — all driven by the same connection you poll with.

Architecture

Modbus Manager Pro runs a local Node.js bridge that talks to your device through the *modbus-serial* library, while the operator interface runs in a desktop window. Everything happens on your machine — no cloud account is required.

What you can do

- Poll coils, discrete inputs, holding registers and input registers over TCP, RTU or ASCII.
- Write single and multiple values back to a device.
- Discover devices and live registers with the Slave Scan and Address Scan tools.
- Simulate one or more Modbus slave devices for testing without hardware.
- Define **tags** and compute derived values with **Structured Text** logic.
- Build interactive **HMI dashboards** with 30+ widget types and run them full-screen.
- Work in 11 languages and save complete setups as portable workspace files.

2. Installation and First Run

Run the Windows installer and launch Modbus Manager Pro from the Start menu or desktop shortcut. On first launch the application starts its local bridge automatically; no manual configuration is needed.

Trial period

New installations begin a **14-day free trial** with the full Pro feature set unlocked. The remaining days are shown in the title bar. The trial start date is tied to your machine, so reinstalling does not reset it.

Activating a license

- 1 Open the license dialog from the toolbar (the license/info button at the top right).
- 2 Copy your **Machine ID** shown in the dialog and send it when purchasing, or paste the key you received.
- 3 Enter the license key and activate. A valid key permanently unlocks the application on that machine.

Machine-bound keys

License keys are generated from your Machine ID, so a key is valid only on the computer it was issued for. If you change hardware, request a new key for the new Machine ID.

3. The Interface at a Glance

The top toolbar holds the global controls — **Connect**, polling configuration, data-format options, language selector, workspace save/open and the license button. Below it sits a row of tabs; each tab is a self-contained workspace for one job.

Tab	Purpose
Registers	Main polling grid — live register values for the active poll.
Chart	Plot polled values over time as a line chart.
Write	Send single or multiple values to coils and holding registers.
Serial Monitor	Raw request/response frames for protocol-level debugging.
Test Center	Send ad-hoc Modbus requests and inspect the replies.
Slave Scan	Probe a range of slave IDs to find responding devices.
Address Scan	Sweep an address range to discover populated registers.
Slave	Run the built-in slave simulator (one or more virtual devices).
Dashboard	Pro HMI builder — design and run operator screens.
Tags	Define tags and write Structured Text logic.

The language selector applies instantly across the whole interface, and the data-format toggles (word swap, 32-bit float, byte order) change how register values are interpreted in the grid and on the dashboard.

4. Connecting to a Device

Click **Connect** to open the connection dialog. It offers nine transport modes, grouped as direct serial, direct network, and framing-over-network for gateways.

Mode	Use it for
TCP/IP	Modbus TCP devices over Ethernet (PLCs, drives, sensors). Default port 502.
RTU	Direct serial RS-485 / RS-232 via a COM port or USB adapter.
ASCII	Serial Modbus ASCII — each byte sent as two hex characters.
RTU / TCP	RTU framing tunnelled over a TCP connection (serial gateway / converter).
ASCII / TCP	ASCII framing over a TCP connection.
UDP	Modbus UDP/IP.
RTU / UDP	RTU framing over UDP/IP.
ASCII / UDP	ASCII framing over UDP/IP.
TCP-SEC	Modbus/TCP Security (TLS). Default port 802.

TCP/IP

- 1 Open **Connect** and select the **TCP/IP** tab.
- 2 Enter the device IP address (for example 192.168.1.100) and port (usually 502).
- 3 Set the Slave / Unit ID (often 1) and click Connect.

RTU (RS-485 / RS-232)

- 1 Connect your USB-RS485 adapter and note its port — COM3, COM4 on Windows; /dev/ttyUSB0 on Linux.
- 2 Open **Connect** → **RTU** and choose the serial port.
- 3 Match the device's baud rate, parity, data bits and stop bits, then connect.

RS-485 options

RTU mode exposes RTS-toggle and echo-removal options for adapters that need them, plus a custom baud-rate field that overrides the dropdown when set.

Automatic reconnection

If a connection drops, Modbus Manager Pro attempts to recover it on its own and reflects the live connection state in the interface, so transient network or cable glitches do not require a manual reconnect.

5. Polling Registers

Once connected, configure the poll from the toolbar and watch values stream into the **Registers** grid.

Function code and addressing

Choose the function code that matches the data you want to read:

FC	Reads	Data type
FC01	Coils	Read/write bits
FC02	Discrete Inputs	Read-only bits
FC03	Holding Registers	Read/write 16-bit words
FC04	Input Registers	Read-only 16-bit words

- **Start address** and **quantity** define the block to read.
- **Address base** switches between Base 0 (protocol addressing) and Base 1 (documentation addressing) so the displayed addresses match your device's manual.
- **Poll delay (ms)** sets the interval between successive polls.

Data formats and scaling

Register values can be displayed as 16-bit integers, combined into 32-bit floats, and reinterpreted with word swap and big- or little-endian byte order using the toolbar toggles. Per-address **scaling** lets you convert a raw register into engineering units (for example a gain and offset) directly in the grid.

Send a register straight to the dashboard

Each row in the grid has a quick action to add that register to the Dashboard, creating a bound widget in one click — handy when building a screen from a live device.

6. Writing Values

The **Write** tab sends values back to the device. Use it to set a single coil (FC05) or holding register (FC06), or to write a block of multiple registers at once. Enter the target address, the value or values, and confirm the write; the result is reported immediately.

Writes affect live equipment

Writing to a connected controller changes its state. Confirm the slave ID, address base and value before sending, especially on production machinery.

7. Charts and Live Monitoring

The **Chart** tab plots the values from the active poll over time, so you can watch trends, spot noise and verify that a signal behaves as expected. It complements the numeric grid for signals that are easier to understand visually, such as temperatures, speeds or power readings.

8. Diagnostic Tools

Slave Scan

Probes a range of slave / unit IDs and reports which ones respond. Use it when you know a device is on the bus but are unsure of its address.

Address Scan

Sweeps a range of register addresses on a known slave to find which addresses return data, helping you map an undocumented device.

Serial Monitor

Shows the raw Modbus request and response frames. This protocol-level view is invaluable for diagnosing timing, framing or CRC problems on serial links.

Test Center

Lets you compose and send individual Modbus requests — including diagnostics (FC08) — and inspect the exact reply, which is ideal for one-off checks and learning how a device behaves.

9. Slave Mode (Simulator)

The **Slave** tab turns Modbus Manager Pro into a Modbus server: it answers requests from a master as if it were real hardware. You can define one or more virtual **slave devices**, each with its own register map, and let an external master (or a second copy of the tool) poll them.

Slave mode is useful for:

- Developing and testing a SCADA/HMI or master application before the hardware exists.
- Reproducing a device's register layout for training or demonstrations.
- Validating your dashboard screens against known, controllable values.

Consistent display

The slave simulator and the master poller use the same register-display conventions (including standard ASCII rendering of register bytes), so values look identical whether you are polling a device or simulating one.

10. Tags and Structured Text

Tags give your data meaningful names and let you compute derived values. They are the bridge between raw registers and a readable dashboard.

Tag types

Type	Description
Register (REG)	Bound to a real register — a slave ID, register type and address. Its value tracks the live device.
Internal (INT)	A computed or stored value that lives only inside the tool. Set it from a Structured Text script or use it as an intermediate result.

The Structured Text script

The Tags tab includes a **Structured Text** editor — the same IEC 61131-3 style language used on PLCs. Your script can read register tags, apply logic and write results to internal tags. Set a **loop interval** (milliseconds) and start the loop to have the script run continuously.

```
(* Structured Text example *)
IF MotorSpeed > 100 THEN
  AlarmActive := TRUE;
ELSIF MotorSpeed < 10 THEN
  AlarmActive := FALSE;
END_IF;

(* Write to internal tag *)
OutputValue := MotorSpeed * 2;
```

Tags feed the dashboard

Any tag — register or internal — can be selected as the source for a dashboard widget. Compute an alarm flag or a scaled value once in a script, then show it anywhere on your screens.

11. Dashboard / HMI Builder

The **Dashboard** is the heart of the Pro edition: a drag-and-drop canvas for building operator screens. Drop widgets from the toolbox, bind them to registers or tags, arrange them, then switch to Runtime to operate the live screen.

Pages, runtime and export

- **Pages** — build multiple screens and switch between them with the page tabs. A page can be marked as a **popup** that opens when an element is clicked instead of appearing in the tab bar.
- **Runtime** — toggles between edit mode and live operation. In runtime, widgets show live values and controls become interactive.
- **Fullscreen** — runs the dashboard as a borderless operator display.
- **Export as PNG** — captures the current screen as an image for reports or documentation.
- **Snap-to-grid**, multi-select, copy/paste and undo/redo make layout quick and precise.

The toolbox

Widgets are organised into collapsible categories. Drag a widget onto the canvas, or click it to drop one at the default position.

Category	Elements
Standard Objects	Rectangle, Rounded Rect, Circle, Ellipse, Polygon, Polyline, Circular Arc, Ellipse Arc, Pie Segment, Ellipse Segment, Static Text, Connector

Category	Elements
Basic	Gauge, Lamp, Bar, Value, Button, Toggle, Line
Solar / Energy	Semi Gauge, Trend, Flow Arrow, Icon + Value, Progress Bar, Status Badge
Process	Tank, Valve, Pump, Pipe
Input / Data	Slider, Setpoint, Clock
Charts	Bar Graph, Multi Trend, Alarm List
Actions	Align, copy, delete and other layout operations on selected elements

Solar / Energy category

The Solar/Energy widgets target PV and inverter monitoring (for example Huawei SmartLogger installations), making it quick to build a plant-overview screen from inverter registers.

Binding a widget: register or tag

Select a widget to open the **properties panel**. Most value-driven widgets offer a **Source** choice:

- **Register** — enter a slave ID, register type and address; the widget reads that register directly. The required register is polled automatically in the background even if no manual poll covers it.
- **Tag** — pick a tag by name; the widget shows that tag's value. Use this for computed or scaled values produced by a script.

Other properties — label, colour, min/max scale, units, decimals, thresholds and size — appear in the panel according to the widget type.

The Charts group and the channel editor

Three widgets show several signals at once, so each one holds a list of **channels** rather than a single binding:

Widget	What it shows
Bar Graph	One vertical bar per channel against a shared min–max scale — good for comparing several live values side by side.
Multi Trend	Several lines on one time chart, each channel keeping its own history up to a configurable number of points.
Alarm List	A table of monitored channels; a row turns active (red) when its value reaches the channel's threshold.

To configure them, select the widget and click **Edit channels...** in the properties panel. The channel editor opens as a dialog where you can:

- 1 Add or remove channels with the **+ Add channel** button and the remove (x) button.
- 2 Set each channel's **name**, **source** (Register or Tag) and **colour**.
- 3 For a **Register** channel, choose the slave ID, register type and address.
- 4 For a **Tag** channel, pick a tag from the dropdown.
- 5 On an Alarm List, set each channel's **threshold**; on a Bar Graph set the shared **min/max**; on a Multi Trend set the **max points** of history.

Mix registers and tags freely

Within a single chart widget you can mix register channels and tag channels — for example plot a raw measurement next to a scripted setpoint on the same Multi Trend.

How channels update

Register channels refresh at the polling rate; tag channels refresh whenever their tag updates. Each is sampled on its own cadence so trend history stays evenly spaced.

12. Workspaces

A **workspace** captures your whole setup — connection settings, poll configuration, tags, scripts and every dashboard page — in a single file. Use **Save Workspace** on the toolbar to write it to your Documents folder, and **Open Workspace** to restore it later or move it to another machine.

Reusable templates

Build a dashboard once for a device family (for example a 24-inverter solar plant) and reuse the workspace file as a starting template for similar installations.

13. Licensing and Trial

- **Trial:** 14 days with the full feature set, starting at first launch.
- **Machine ID:** a unique identifier for your computer, shown in the license dialog.
- **Activation:** enter the key issued for your Machine ID to unlock permanently.
- **Status:** the title bar shows the days remaining, or that the product is licensed.

Reinstalling

The trial start is recorded outside the application folder, so uninstalling and reinstalling will not extend or reset the trial period.

14. Languages

The interface is fully translated into 11 languages, selectable at any time from the toolbar: English, Chinese, Hindi, Spanish, French, Arabic, Bengali, Russian, Portuguese, Urdu and Estonian. Right-to-left layouts (Arabic and Urdu) are handled automatically.

15. Troubleshooting and FAQ

Symptom	Things to check
No response from a TCP device	Confirm IP address and port (502 by default), that the device is reachable on the network, and that the slave/unit ID is correct.
Serial (RTU) connection fails	Verify the COM port, baud rate, parity, data and stop bits match the device. Check adapter drivers and RS-485 wiring (A/B polarity).
Values look wrong or scaled oddly	Try the word-swap and byte-order toggles, check the 32-bit float option, and confirm the address base (0 vs 1) matches the device manual.
A dashboard widget shows nothing	Open its properties (or Edit channels) and confirm the source, slave ID and address — or that the selected tag exists and is updating.

Symptom	Things to check
A chart channel never moves	For register channels, make sure the device actually returns that address; for tag channels, confirm the tag updates (run the script loop if needed).
Trial expired	Activate a license key for your Machine ID from the license dialog.

16. Appendix A — Modbus quick reference

Read function codes

FC	Operation	Object
01	Read Coils	Read/write bits
02	Read Discrete Inputs	Read-only bits
03	Read Holding Registers	Read/write 16-bit words
04	Read Input Registers	Read-only 16-bit words

Write / other function codes

FC	Operation
05	Write Single Coil
06	Write Single Register
08	Diagnostics

Addressing

Base 0 uses raw protocol addresses (the first register is 0). **Base 1** uses documentation addresses (the first register is 1). Pick the base that matches your device's manual so the addresses you type line up with the datasheet.

© Ergo • modbusmanager.com • Modbus Manager Pro v1.4.0 — This manual describes the application's features; the interface may evolve between releases.