

Modbus and Serial Configuration

The Gateway has a very advanced Modbus master application with a convenient graphical interface. Below are the capabilities of the application:

- Modbus TCP/IP and Modbus RTU support.
- Any number of serial ports
- Support for external USB/serial interfaces
- JSON, TCP/IP and MQTT string formation
- Data logging in the event of no connection to a server
- Storage of all data in internal memory

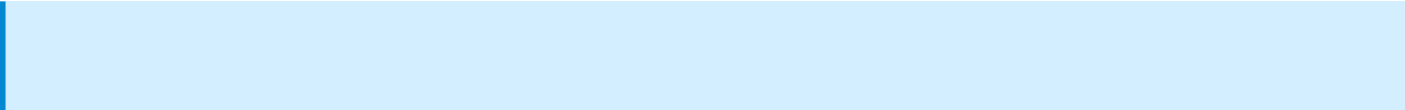
“ The Modbus Master application is optimized for industrial use, allowing you to easily configure communication parameter, define polling schedules, and integrate Modbus data into your automation or monitoring workflows.

A detailed guide to **Modbus Master** configuration is available [here](#).

1.1 Modbus Fundamentals

Modbus is an industrial communication protocol used to exchange data between a client/master and one or more server/slave devices. The client starts the request and the server returns the requested data. This guide only covers the Modbus TCP and Modbus RTU functions used by this Modbus Master application.

Modbus TCP	Uses Ethernet/TCP networking. Configure the slave IP address and TCP port. Port 502 is the standard Modbus TCP port.
Modbus RTU	Uses a serial communication link such as RS-485. Configure the serial port and the serial communication settings to match the slave.



Always start with the slave manufacturer's Modbus map. It tells you the communication settings, slave/unit address, function code, register/coil address, quantity and data format required for that device.

1.2 Select the Communication Method

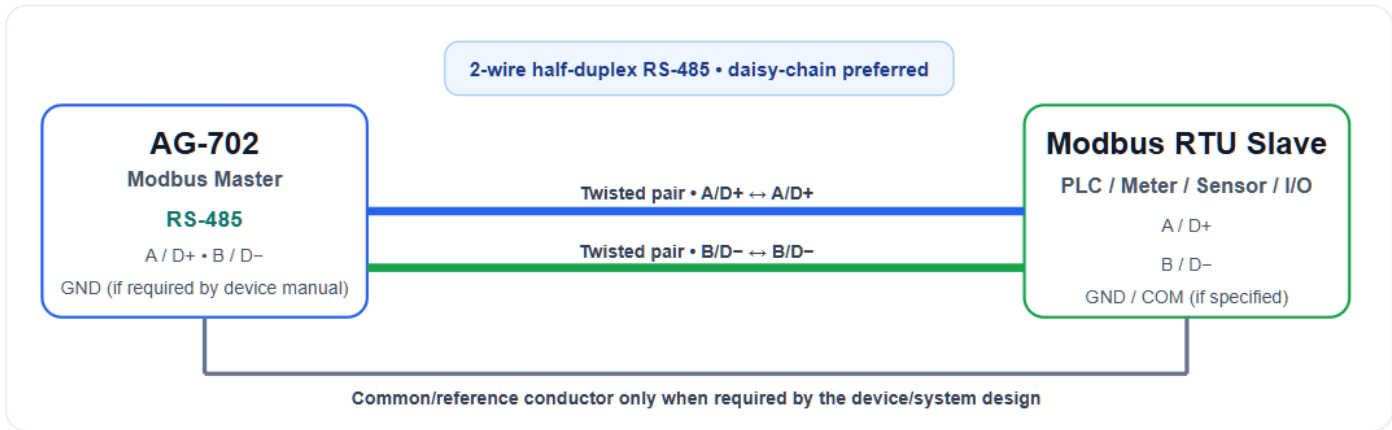
Requirement	Use	What the user connects/configures
Slave is on an Ethernet/IP network	Modbus TCP	AG-702 Ethernet → network/switch → Modbus TCP slave; configure IP address and port.
Slave is a field device on a serial bus	Modbus RTU over RS-485	AG-702 RS-485 → RS-485 slave bus; configure serial port, baudrate, data bits, parity, stop bits and slave ID.
Slave requires point-to-point serial	Modbus RTU over RS-232	Use the AG-702 RS-232 interface only when the slave and application support that serial interface.

Do not mix the settings. Modbus TCP uses an IP connection; Modbus RTU uses serial communication parameters.

1.3 Wiring & physical connections

Use the examples below as connection patterns. Exact connector names and polarity markings must always be checked against the connected device's own hardware manual.

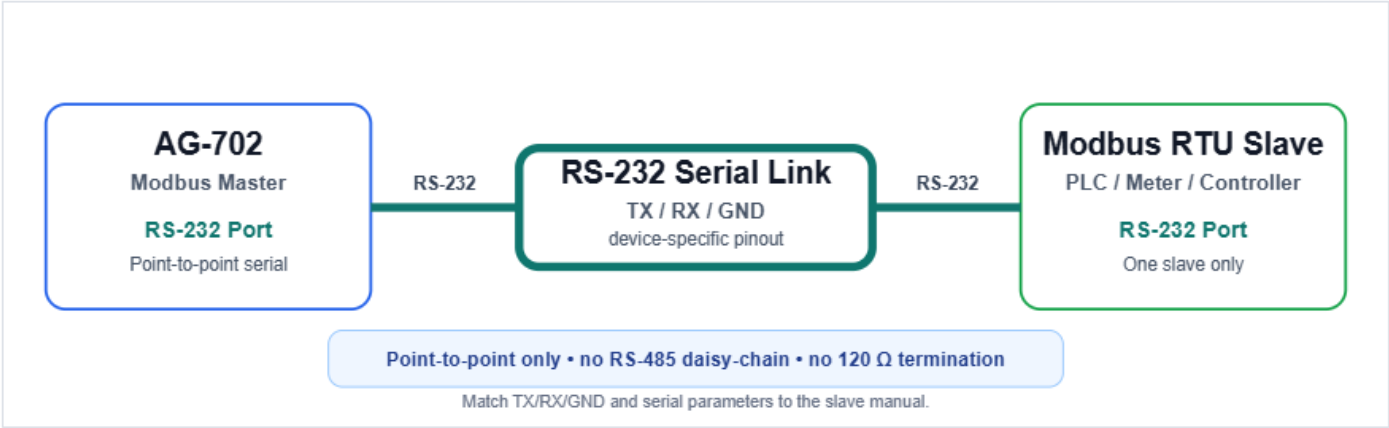
Example A — Modbus RTU over RS-485 (2-wire)



Use	Use the AG-702 RS-485 interface with a compatible Modbus RTU slave.
Cable	Use a twisted-pair cable suitable for RS-485. Keep the bus physically ordered and avoid unnecessary long stubs.

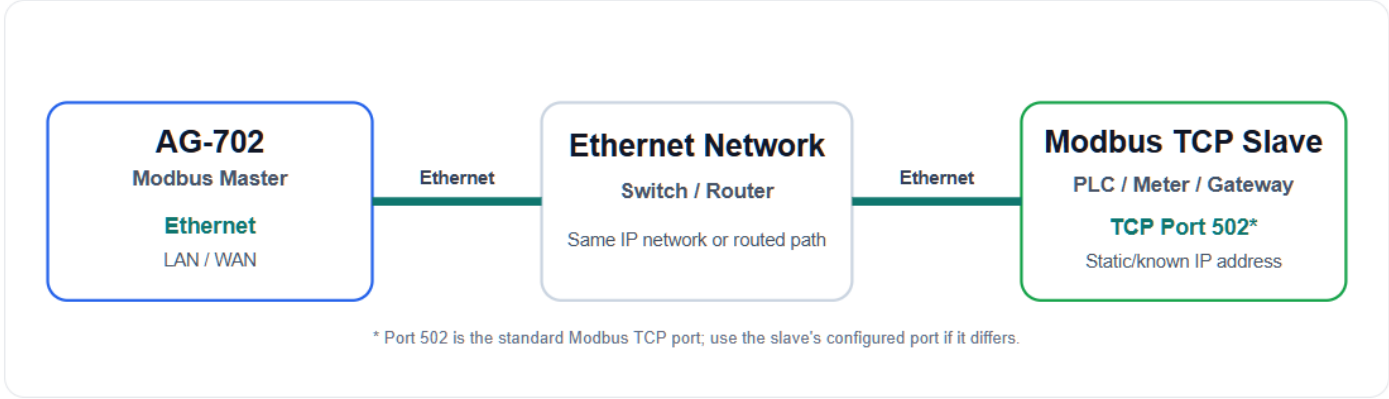
Termination	For a properly terminated bus, termination matches the cable impedance; Schneider recommends 100–120 Ω at the two physical ends of the RS-485 network where required.
Biasing	Failsafe biasing may be required when the bus can be left electrically idle. Only one node should provide the biasing.

Example B — Modbus RTU over RS-232 (point-to-point)



Important: RS-232 is a point-to-point serial interface. Do not use an RS-485 bus wiring model on an RS-232 port. Match TX/RX/GND and the slave's serial parameters exactly as specified by the slave hardware manual.

Example C — Modbus TCP over Ethernet

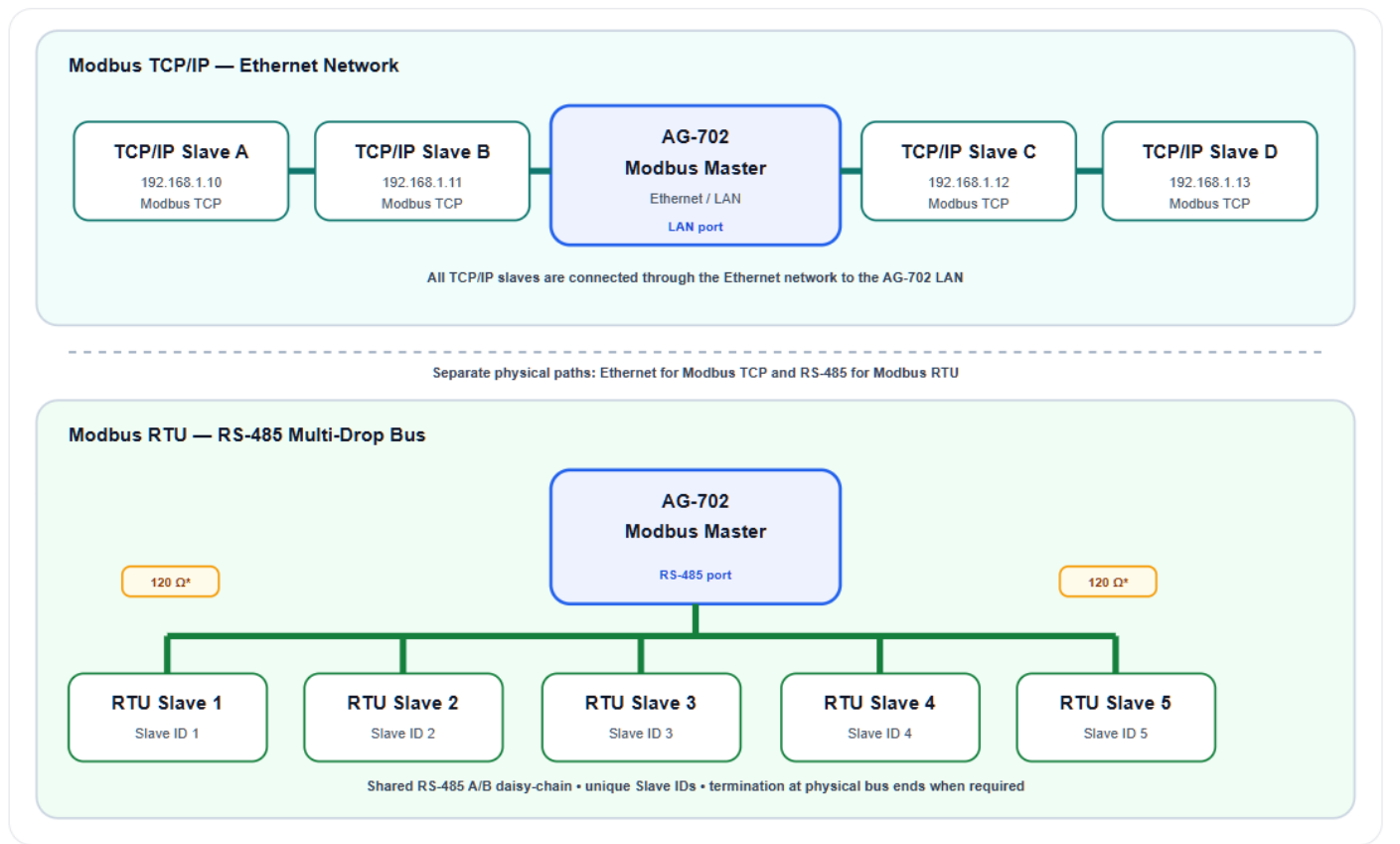


Use	Connect an AG-702 Ethernet interface to the IP network containing the Modbus TCP slave.
Configure	Use the slave IP address and TCP port in the Modbus Master application. Port 502 is the standard Modbus TCP port.

2. Modbus Deployment Scenarios

These diagrams show how the AG-702 can collect data from multiple RS-485 Modbus RTU slaves and multiple Modbus TCP/IP slave devices.

2.1 AG-702 Mixed TCP/IP & RS-485 Deployment



In this example the AG-702 is the central Modbus Master. Five RTU slaves share one RS-485 daisy-chain connected directly to the AG-702 RS-485 port: three slaves are shown on the left and two on the right. Four separate TCP/IP slaves connect through the Ethernet network on the AG-702 LAN side.

RS-485 side — five RTU slaves	• All five devices connect to the AG-702 RS-485 port. • Three slaves are shown on the left and two on the right. • All five share one RS-485 daisy-chain. • Each slave has a unique Slave ID.
Ethernet side — four TCP/IP slaves	• TCP/IP Slave A, B, C and D are slave devices. • They connect to the AG-702 LAN through the same Ethernet network. • Each slave has its own reachable IP address. • They are not gateways in this example.

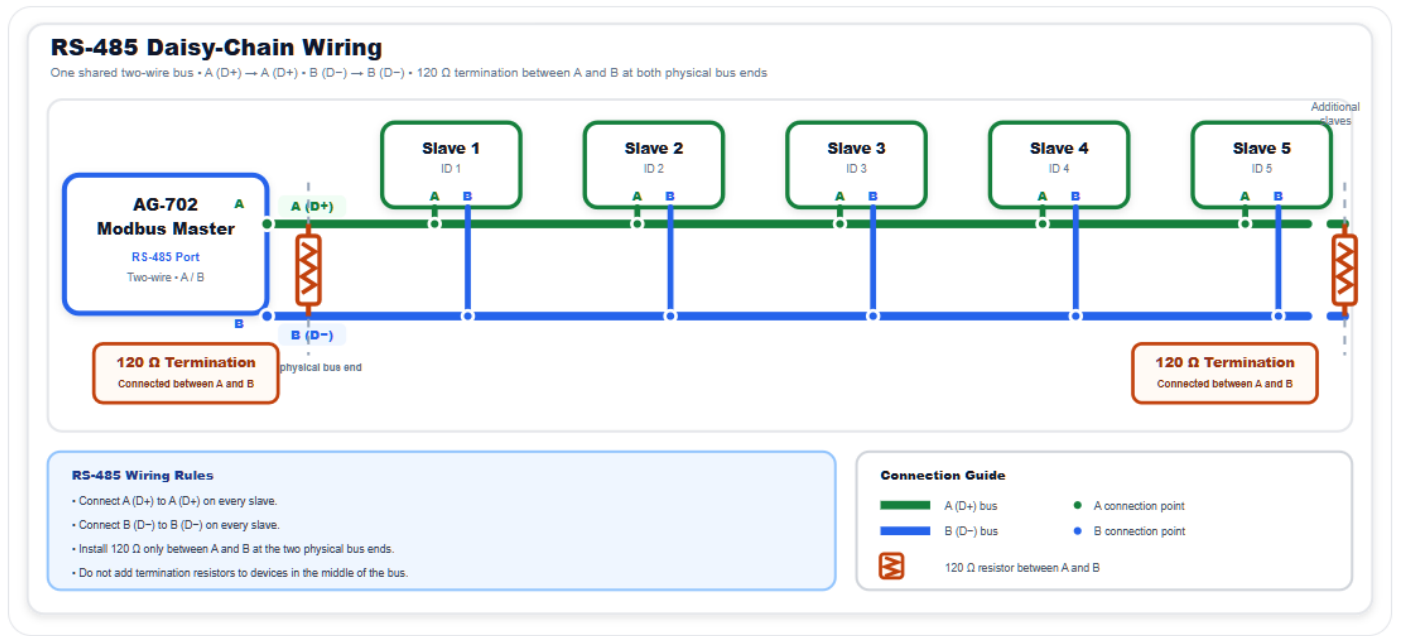
Device group	Physical connection	Protocol	Addressing
--------------	---------------------	----------	------------

RTU Slave 1-5	AG-702 RS-485 → shared A/B bus	Modbus RTU	Slave IDs 1-5
TCP/IP Slave A-D	AG-702 LAN → Ethernet network → slave	Modbus TCP	IP address + device/unit identifier as required

Important: RTU slaves are connected to the AG-702 RS-485 port. TCP/IP slaves are connected through the AG-702 Ethernet/LAN port. These are two separate physical communication paths.

2.2 RS-485 Daisy-Chain Wiring — Multi-Slave Field Example

A single AG-702 RS-485 port can serve multiple Modbus RTU slave devices on one shared two-wire bus. Connect **A (D+)** to **A (D+)** and **B (D-)** to **B (D-)** throughout the chain, assign a unique Slave ID to every device, and install a **120 Ω termination resistor between A and B** only at the two physical ends of the bus.



Important: RS-485 is a shared two-wire bus. Keep the topology physically ordered as a daisy-chain, keep drop cables short, connect A to A and B to B consistently, and place 120 Ω termination resistors only at the two physical ends of the bus.