WHITE PAPER

MODBUS TCP FOR TEST AND MEASUREMENT SYSTEMS

CONNECTING DATA ACQUISITION AND INDUSTRIAL AUTOMATION



DEWETRON

ABSTRACT

In modern test bench technology, energy analysis, and industrial process monitoring, the seamless integration of highly dynamic physical measurement data with static or quasi-static system parameters poses a critical challenge. While physical quantities such as voltages, currents, vibrations, and pressures must be acquired at high sampling rates in the kilohertz or megahertz range, peripheral components - including photovoltaic inverters, battery management systems (BMS), climatic chambers, and programmable logic controllers (PLCs) - typically deliver their status data via standardized field bus protocols.

This whitepaper describes the technical architecture and implementation of the Modbus TCP protocol (Modbus Ethernet) within DEWETRON's OXYGEN measurement and analysis software. It details how external data sources are integrated time-synchronously into the measurement data stream using the Modbus TCP Client Plugin. Furthermore, it demonstrates how the newly introduced Modbus-TCP Server module transmits calculated measurement values (e.g. from power analysis) to higher-level control systems at rates of up to 100 Hz.

INTRODUCTION

The Modbus protocol was developed in 1979 by Modicon (now Schneider Electric) for communication with programmable logic controllers (PLCs). Originally designed for serial interfaces (Modbus RTU via RS-485), it was ported to the Layer 4 TCP/IP protocol with the specification of Modbus TCP. This eliminates serial restrictions such as baud rate conflicts or strict master-slave wiring over dedicated bus lines.

The validation of systems in the wake of global decarbonization and electrification - such as electric powertrains, inverters, or smart grids - requires highly sophisticated measurement tools. Historically, data acquisition (DAQ) systems operated as isolated islands that purely digitized analog signals. Today's test benches, however, demand bidirectional interaction:

- Aggregation Requirement: Data from devices under test (DUT) like solar inverters, frequency converters, or charging stations must be recorded directly alongside high-precision power data.
- Distribution Requirement: The DAQ system acts as an intelligent sensor (e.g. an Advanced Power Analyzer) and must report calculated values such as active power, efficiency, or harmonics back to the plant control system.

Despite its age, Modbus TCP has established itself as a de facto standard in this environment. It is license-free, platform-independent, extremely simple to implement, and robust. DEWETRON OXYGEN addresses these requirements natively through integrated Modbus extensions.

HOW MODBUS TCP WORKS

Client Server architecture

Modbus TCP utilizes a strict request-response architecture based on TCP/IP:

- Modbus Client (Master): Initiates communication, sends requests to the server, and evaluates its responses.
- Modbus Server (Slave): Waits passively for requests on a defined TCP port (default: 502), processes the function codes, and returns the requested data.

Protocol Data Structure (PDU & MBAP)

In Modbus TCP, error control is entirely handled by the underlying Ethernet and TCP layers via the TCP checksum, eliminating the need for the CRC checksum used in Modbus RTU. For network routing and transaction tracking, Modbus TCP introduces an MBAP Header (Modbus Application Protocol Header), which is prepended to the Modbus PDU (Protocol Data Unit).

The MBAP header is structured as follows:

- Transaction Identifier (2 Bytes): Unique identifier for matching requests and responses during asynchronous communication.
- Protocol Identifier (2 Bytes): Always 0 for the Modbus protocol.
- Length (2 Bytes): Number of following bytes (including the Unit Identifier and data fields).
- Unit Identifier (1 Byte): Used for addressing devices behind a Modbus gateway (e.g. serial RTU slaves on a TCP network).

The PDU subsequently contains the Function Code (1 Byte) and the Data Bytes (parameters and payload)

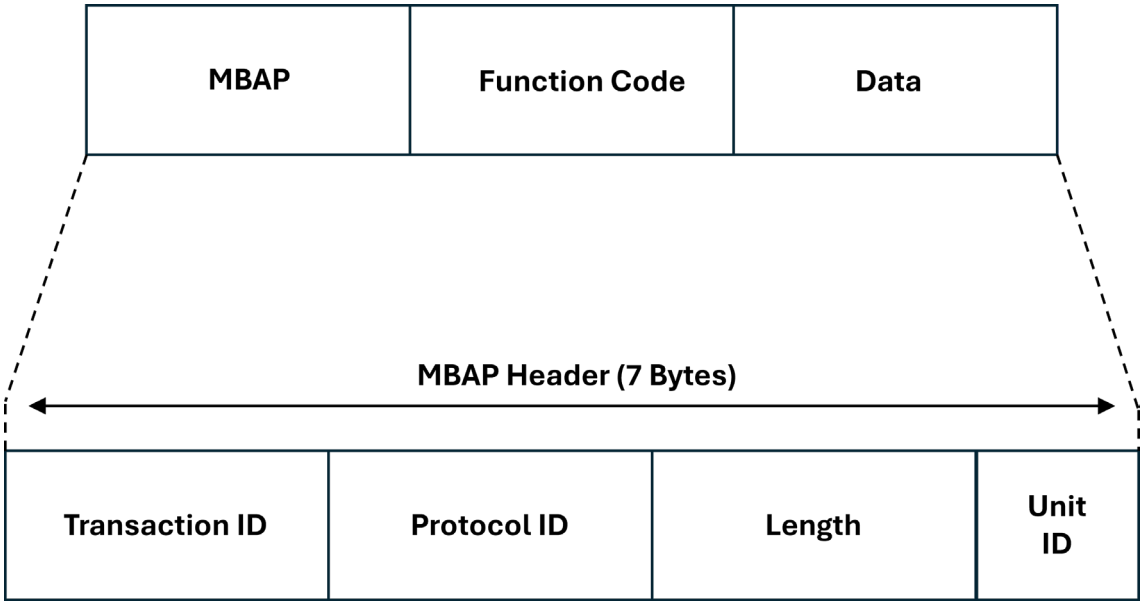


Fig. 1: MBAP

Storage and datamodel

Modbus structures data into four primary, separate table blocks, which are historically addressed on a bit or word basis:

Table type	Data width [Bit]	Access	Description
Discrete Inputs	1	Read Only	Digital inputs (e.g. relay status, switches)
Coils	1	Read-Write	Digital outputs (e.g. valves, LEDs)
Input Registers	16	Read Only	Analog input values (e.g. raw sensor data)
Holding Registers	16	Read-Write	Configuration data, general-purpose measurement data

Tab. 1: Modbus Table Blocks

Addressing in the telegram ranges from 0 to 65535. However, many controllers use the traditional 5- or 6-digit addressing scheme (e.g. 40001 for the first holding register), where the table type is implicitly encoded by the leading digit.

Relevant function codes

Word-based access is of primary importance for industrial measurement technology:

- FC 03 (Read Holding Registers): Reading configuration and measured values.
- FC 04 (Read Input Registers): Reading pure input data.
- FC 06 (Write Single Register): Writing a single 16-bit value.
- FC 16 (Write Multiple Registers): Writing a continuous data block (important for 32-bit/64-bit data types).

Data Conversion and the Data Type Problem

A central weakness of the original Modbus specification is the lack of native support for modern data types such as floating-point numbers (floats) or signed 32-bit integers, as the protocol operates strictly in 16-bit registers. To resolve this limitation, Register Splitting (Modbus Mapping) is used: To transmit a 32-bit value (float or Int32), two consecutive 16-bit registers must be combined. For 64-bit values (double or Int64), four registers are required.

Endianness and Byte Order

While Ethernet protocols typically dictate Big-Endian (Most Significant Byte first), the sequence of register combinations in Modbus is not standardized. This leads to four possible byte orders (word/byte swapping) when reconstructing a 32-bit float:

- Big-Endian (ABCD): Standard order.
- Little-Endian (DCBA): Completely inverted.
- Mid-Little Endian / Word Swapped (CDAB): Bytes within the words are Big-Endian, but the words themselves are swapped.
- Mid-Big Endian / Byte Swapped (BADC): The words are in the correct order, but the internal bytes are swapped.

Measurement software must strictly provide mechanisms to flexibly correct these transformations via software configuration, as misinterpretations lead to incorrect measured values.

INTEGRATING EXTERNAL DEVICES WITH THE OXYGEN MODBUS TCP CLIENT PLUGIN

The OXYGEN Modbus-TCP Client Plugin is designed to enable DEWETRON measuring instruments to actively retrieve data from external Modbus servers (e.g. PV inverters, power analyzers, or climate chambers).

Once the plugin is activated in the OXYGEN setup, the software acts as a client. It establishes periodic TCP connections to specified IP addresses. The plugin parses the incoming Modbus register packets based on a predefined mapping and converts them into scalar OXYGEN channels.

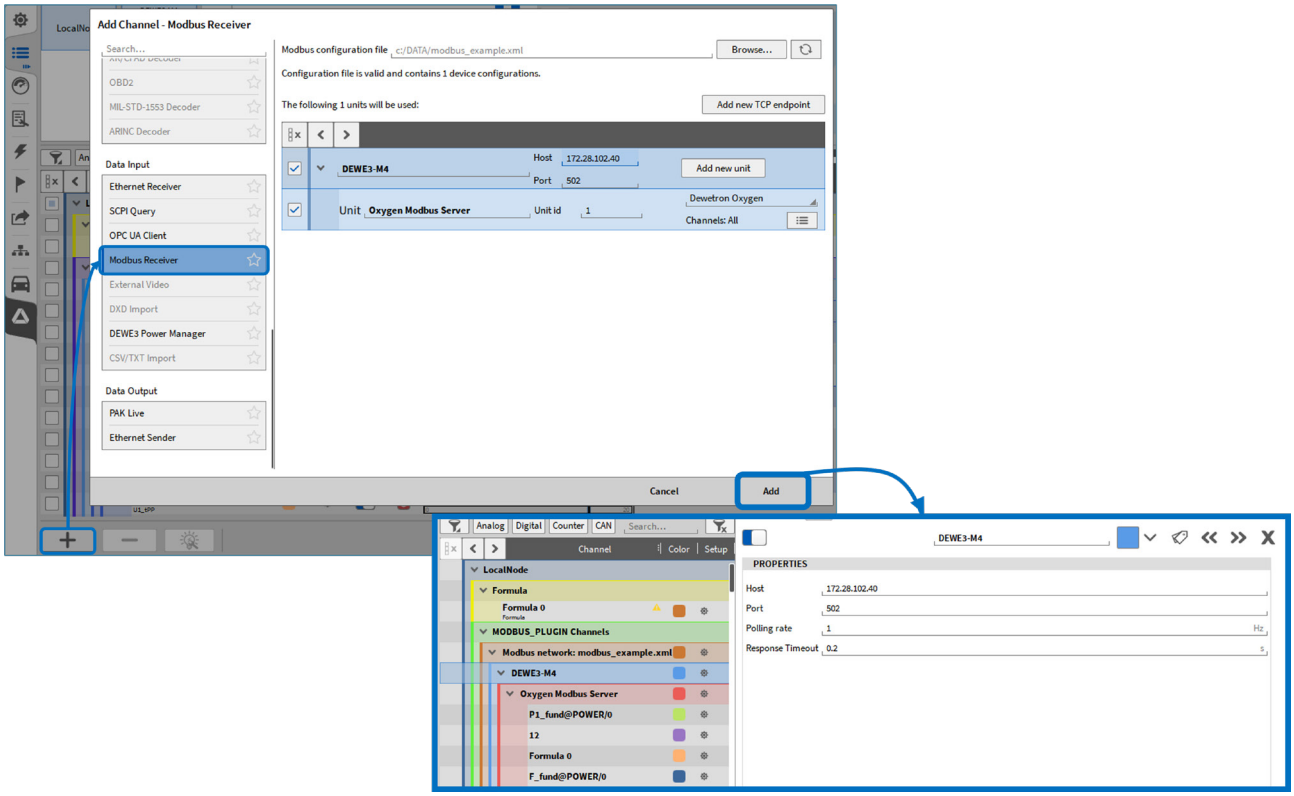


Fig. 2: MODBUS Client (Receiver)

Configuration Workflow in the User Interface

1. Establishing a Connection: A new instance of the Modbus TCP client is added via the Add Channel menu. A configuration file in .xml format is required, containing all network information and device details, which will be automatically shown. Following parameters will be automatically addressed.
2. Host address and port information
3. Unit ID Definition: If a gateway is used, the corresponding device ID is entered
4. Polling rate: The used rate to poll data from the device.
5. Register Configuration: An entry is created in the configuration table for each desired measurement channel automatically according to the selected configuration file:
   - Register Type: Holding or Input Register.
   - Start Address: The exact offset address of the device (0-based or 1-based).
   - Data Type: Int16, UInt16, Int32, UInt32, Float (32-bit), and Double (64-bit).
   - Byte Order: Word/Byte swapping to prevent decoding errors.
6. Scaling and Offset: Sensors often transmit fixed-point numbers (e.g. temperature as the integer 235 instead of 23.5 °C). OXYGEN allows you to define a scaling factor (e.g. 0.1) and an offset directly in the channel.

Application example: Integration of a solar inverter

During performance tests of large-scale photovoltaic systems, a DEWETRON system measures the analog AC and DC currents and voltages with extremely high dynamic range. At the same time, the OXYGEN Modbus Client Plugin queries the inverter via the local Ethernet network.

Transmitted parameters such as inverter temperature, error codes, and MPPT (Maximum Power Point Tracking) status are now available as scalar channels in OXYGEN. They can be displayed directly in a recorder, processed using mathematical functions, and saved in the central measurement file (.dmd).

SHARING MEASUREMENT DATA VIA THE OXYGEN MODBUS TCP SERVER PLUGIN

Introduced with the latest development stages (optimized in version OXYGEN 8.1), the Modbus Server module enables bidirectional communication. OXYGEN makes its internal, highly accurate measurement channels available on the network as Modbus holding registers so that higher-level systems can access them.

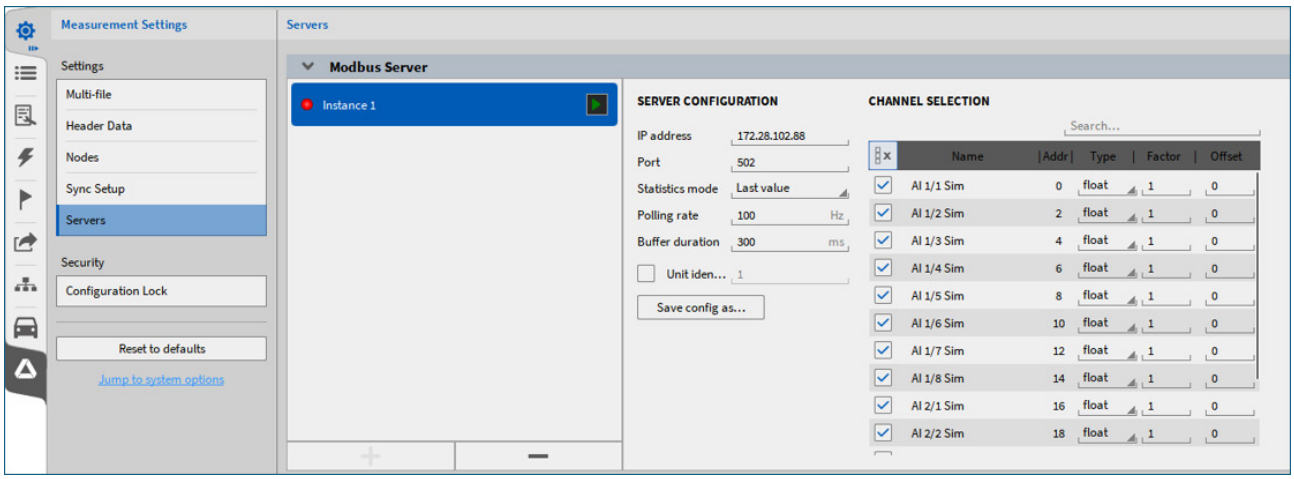


Fig. 3: MODBUS Server (Sender)

Architecture and performance-optimization

Measurement calculations (e.g. RMS values over grid periods) generate data streams that are too fast or too unstructured for traditional fieldbuses. The Modbus Server Module acts as an intelligent buffer:

- Polling Rates: The server supports polling rates of up to 100 Hz. This is more than sufficient to reliably supply control variables to control loops for test bench motors or PLC control systems.
- Channel Support: Both fully synchronous and asynchronous scalar channels are supported.
- Statistics Modes: Before publishing to the tab, you can specify whether the server should output the most recent value (Last Value) or calculate an average (Average) over the query interval.

Data Provision and Register Mapping

In the configuration menu, the user specifically selects which OXYGEN channels are to be made available to the network. The system automatically generates a register map based on this selection.

Since OXYGEN uses high-precision floating-point values internally, the following data types are available by default during server export:

- Float (32-Bit, occupies 2 Register)
- Double (64-Bit, occupies 4 Register)
- Int16 / UInt16 (1 Register)
- Int32 / UInt32 (2 Register)

In addition, a specific scaling factor (factor/offset) can be specified if the receiving client (e.g. an older PLC) cannot process floating-point numbers and the data must be converted back to scaled integers.

Application example: Test Bench Control in E-Mobility Testing

In a drive test bench, the OXYGEN Power Analyzer module calculates the mechanical power (based on torque and speed) and the electrical efficiency of the inverter in real time. The test bench automation computer acts as a Modbus client and periodically retrieves these efficiency values via Modbus-TCP. Based on this feedback, the PLC controls the load cycle of the electric motor without the need to implement proprietary analog interfaces or complex protocols.

SYNCHRONIZATION AND TIMING ASPECTS IN ETHERNET NETWORKS

A critical aspect of using Modbus-TCP in measurement technology is timestamping. Unlike dedicated hardware buses (such as CAN bus or EtherCAT), Modbus-TCP does not have native, hardware-based time synchronization in the protocol frame.

Jitter and Latency in TCP/IP

Since Ethernet is based on the CSMA/CD protocol and TCP forces packet retransmissions in the event of loss, the transmission time of Modbus packets is subject to statistical jitter. Latencies can range from a few milliseconds to hundreds of milliseconds, depending on network load.

OXYGEN Internal Timestamping

To enable meaningful correlation of Modbus data with highly dynamic analog channels (e.g. 200 kS/s), OXYGEN uses advanced software-based timestamping:

- Client-side: As soon as a Modbus response packet reaches the DEWETRON device's network interface, the OXYGEN Core adds the current, high-precision system timestamp (based on the selected time source, such as GPS, PTP, or IRIG-B) to it.
- This ensures that Modbus data is displayed within the software in phase with the transient events of the analog signals, even if the Modbus device's polling rate is low.

SUMMARY

The integration of the Modbus-TCP protocol as both a client and server plugin makes DEWETRON OXYGEN a highly advanced interface for industrial test environments. The client plugin allows for the effortless expansion of the measurement system with peripheral sensor data without tying up expensive analog input cards. The server module, in turn, transforms the DAQ system into an edge computing sensor that feeds calculated high-end measurement data directly into the industrial control loop.

The combination of simple register-based configuration, automatic handling of complex data types (word swapping), and the uncompromising stability of the OXYGEN core on Windows and Linux platforms leads to a sustained increase in efficiency during test bench and system validation.

AUTHORS

Michael Fuchs

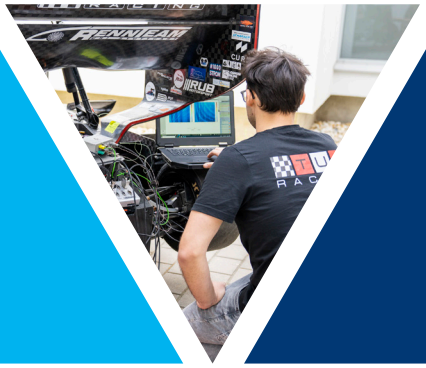


Michael Fuchs has worked as an application engineer for a Graz-based company in the automotive supply industry since 2011. During his employment, he specialized in the post-processing of measurement data from emission test systems in accordance with global technical regulations. In 2022, he joined DEWETRON as an application engineer for automotive, e-mobility, energy applications, test benches, and general test and measurement solutions.









ABOUT DEWETRON

DEWETRON is a manufacturer of precision test and measurement systems and part of the globally operating Anritsu Group. Our reliable measurement data help customers worldwide make processes more predictable, efficient, and safer.

Our strength lies in customized measurement solutions: ready to use right away while remaining flexible to adapt to dynamic testing requirements in the energy, automotive, transportation, and aerospace industries.

More than 35 years of experience and innovation have made DEWETRON a trusted partner in the global test and measurement market.

More than 25,000 DEWETRON measurement systems and over 400,000 measurement channels are in continuous use at leading companies worldwide.

DEWETRON's quality is certified according to ISO 9001 and ISO 14001.
The high integrity of the measurement data is guaranteed by our own accredited calibration lab according to ISO 17025.

THE MEASURABLE DIFFERENCE.

Get to know our
GLOBAL OFFICES

DEWETRON

HEADQUARTERS

DEWETRON GmbH
Parkring 4, 8074 Grambach
AUSTRIA

+43 (0) 316 3070-0
info@dewetron.com
www.dewetron.com