Connect to a Modbus Device Tutorial

Connect to a Modbus device and map register addresses to tags.

This tutorial teaches you to:

- Configure a Modbus TCP communication channel
- Add a device node for a Modbus slave
- Map Modbus register addresses to UNS tags
- Monitor communication status at runtime

This tutorial teaches you to:
1. Create the Device Channel
2. Add a Device Node
3. Create Device Points
4. Start the Modbus Device Simulator
5. Test Communication

Prerequisites

- Add Tags to UNS (Tutorial)

---

1. Create the Device Channel

1. Navigate to **Devices Channels**
2. Click **New channel** and configure:
    - Channel Name: `ModbusChannel`
    - Protocol: `Modbus`
    - Interface: `TCP/IP`

---

2. Add a Device Node

1. Go to **Devices Nodes**
2. Create a node with:
    - Name: `PLC1`
    - Channel: `ModbusChannel`
    - PrimaryStation: `127.0.0.1;502;1` (semicolon-separated `IP;Port;SlaveID` — 502 is the standard Modbus TCP port, 1 is the unit/slave ID)

---

3. Create Device Points

Modbus addresses are composed as an **operand prefix** + a **5-digit zero-padded address**. The operand selects the register family:

- `0xxxxx` — Coil (read/write bit)
- `1xxxxx` — Discrete Input (read-only bit)
- `3xxxxx` — Input Register (read-only 16-bit)
- `4xxxxx` — Holding Register (read/write 16-bit)

1. Navigate to **Devices Points**
2. Add the points below.

Holding Register — Tank 1 Level:

- Tag Name: `TankFarm/Tank1/Level`
- Node: `PLC1`
- Address: `400001` (Holding Register #1)
- Data Type: `Native`
- AccessType: `ReadWrite`

Holding Register — Tank 2 Level:

- Tag Name: `TankFarm/Tank2/Level`
- Node: `PLC1`
- Address: `400002` (Holding Register #2)
- Data Type: `Native`
- AccessType: `ReadWrite`

Tip: `Native` lets the driver pick the data type from the register width. To force a specific type, use the canonical enum values (`Real`, `Single`, `Bit`, etc.) — see the Modbus driver Reference for the full list and for byte/word-order options.

---

4. Start the Modbus Device Simulator

1. Navigate to **Devices Channels**
2. Select `ModbusChannel`
3. Run the built-in Modbus Simulator (right-click the channel row **Run Simulator**). The simulator listens on `127.0.0.1:502` by default and emulates a Modbus slave so you can test the loop without hardware.

---

5. Test Communication

1. Click **Start Runtime**.
2. Open **Devices Monitor**.
3. Confirm the node is healthy: `ActivityCounter` is increasing and `LastErrorCode` stays at `0`.
4. Open the tag-value view (**Unified Namespace Tags**, runtime grid) and confirm `TankFarm/Tank1/Level` and `TankFarm/Tank2/Level` are receiving non-stale values from the simulator.
