



Writing a Value on Startup

[Kepware](#) document KTAN-90048

Kepware OPC servers v4.190.341 and higher; all versions of LinkMaster

2/21/07 – v. 1.01

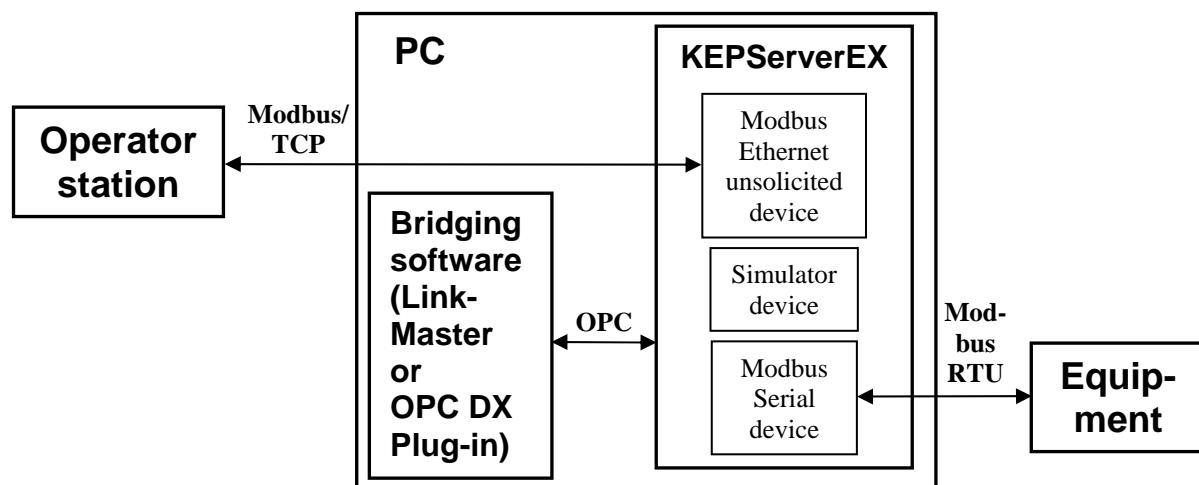
Introduction

This document describes an example of setting up Kepware software to solve the following problem:

- A. A constant value (e.g., a password) is to be written to a piece of equipment automatically on startup.
 - For this example, we will use Modbus RTU (serial) as the communications protocol.
- B. The software also must be able to write this value on command from an operator.
- C. The operator command is sent via a device communications protocol, not via OPC.
 - For this example, we will use Modbus/TCP (Ethernet) as the operator-station comms protocol.
 - OPC is meant for reliable plant networks. A WAN between the operator station and the Kepware software may have occasional network outages that can be better tolerated by a device protocol.
- D. The operator must be able to monitor the communications status of the equipment.

The reader should be moderately familiar with [Kepware OPC servers](#) and LinkMaster or the OPC DX Plug-in.

The following is a block diagram of the software and hardware:



Setup

The block diagram shows two main software components: KEPServerEX (“the server”) and bridging software.

Server Setup

The devices in the server will have the following tags and properties:

Modbus Serial device:

- The tag (we will call it a “value tag”) to which the constant value is to be written
- <Channel>.<Device>._System._Error tag (**automatically provided by the server**)
- <Channel>.<Device>._System._NoError tag (**automatically provided by server versions 4.190.341 and higher**)

Simulator device:

- At the channel level in the Simulator driver, Item Persistence is turned on and a file is specified. (See “Item Persistence File,” below.)
- Tag (“value tag,” e.g., S0 if a string, K0 if a number, K0.0 if a Boolean) holding the constant value
- <Channel>.<Device>._System._Enabled tag (**automatically provided by the server**)

Modbus Ethernet unsolicited device:

- To make an unsolicited device, you can set its device ID to 127.0.0.1.<optional 1-byte node ID>
- Boolean write-trigger tag (the operator station writes to this tag to send the value to the equipment)
- Boolean comms-status tag (reflecting the communications status of the Modbus Serial device)

Bridging Setup

The following links in LinkMaster (or connections in the OPC DX Plug-in) will be necessary:

To write the constant value:

- Modbus Ethernet write-trigger tag to Simulator _Enabled tag
- Simulator value tag to Modbus Serial value tag

To write the value if the equipment’s power is cycled while the server PC keeps running:

- Modbus Serial _NoError tag to Simulator _Enabled tag

To let the operator station monitor the status of the server’s communications to the equipment:

- Modbus Serial _Error tag to Modbus Ethernet comms-status tag

Item Persistence File

To generate the item persistence file, do the following.

- a. You can use an OPC Quick Client (local to the server) to write the value to the Simulator value tag.
- b. Fully shut down the server once (File | Exit and Shutdown, if available, otherwise File | Exit). This will generate the item persistence file.
- c. From then on, the file will automatically be loaded when the server starts up with your project file.

Operation

Writing the Value by Hand

1. The Modbus Ethernet master (at the operator station) writes a 0, then a 1, to the Modbus Ethernet write-trigger tag.
2. The bridging software copies these two writes to the Simulator _Enabled tag.
3. The _Enabled tag changes from 0 to 1, so:
4. The server sends a data-change event for the Simulator value tag to the bridging software.
5. The bridging software writes the data-change value to the Modbus Serial value tag.

Automatic Operation on Server Startup

Whenever the server starts up, steps 4 and 5 in “Writing the Value by Hand” will happen automatically, provided the following note is observed.

Note: The state of the _Enabled tag is saved in the server’s project file. The state must be saved as 1 (device enabled) in order for the value to be sent when the server PC powers up. The state of this tag is 1 by default, and can be reviewed by looking, on the Simulator device’s Device Properties | General tab, at the “Enable data collection” checkbox. If the box is checked, the _Enabled tag is 1.

Automatic Operation on Equipment Restart

If the equipment’s power is cycled while the server PC keeps running, the Modbus Serial _NoError tag will go to 0, then 1. This (since the bridging software is linking this _NoError tag to the Simulator _Enabled tag) will have the same effect as the procedure listed in “Writing the Value by Hand.”