# Application Note:
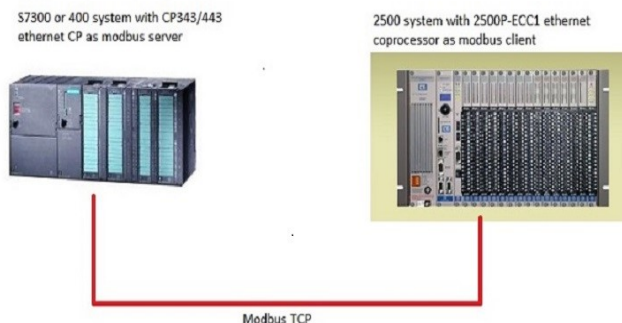
## Communicating between CTI 2500 Series® PLC and Simatic S7 PLC using 2500P-ECC1 and Modbus/TCP Protocol

**2500 Series® System Application Note**

### Application

A CTI 2500 Series® PLC is configured to exchange data with a Siemens S7 PLC, allowing Modbus registers to be written or read from the CTI CPU. The S7 PLC functions as a Server and the CTI PLC as a Client.



### Assumptions

- The CTI PLC uses the 2500P-ECC1 Ethernet Communications Coprocessor card as the network interface. The S7 PLC uses a CP343/443-1 card with a configured connection to the CTI PLC.The Siemens CP cards which are released for this application are 6GK7 343-1EX11-0XE0, 6GK7 443-1EX11-0XE0 or later versions. The Lean versions of the CP343/443 can also be used.
- The S7 PLC is functioning as Server and the CTI PLC as Client.
- The IP address for the CP343-1 is 10.200.64.93
- The IP address for the 2500P-ECC1 is 10.200.64.22
- In this example the CTI PLC will read 10 words from the S7 PLC using Modbus register 1-10 which corresponds to address V29851-V29860 in the CTI PLC and write 10 words to the S7 PLC using Modbus register 11-20 which corresponds to address V29701-V29710 in the CTI PLC.
- A Function block is used in the S7 PLC to map S7 addresses to Modbus registers. It can be used in Step7 V5.1 or higher.
- This Function block is freeware and does not require a software license.

### Description

The configuration has to be carried out on both the S7 side and on the CTI site at the 2500P-ECC1 using the 2500P-ECC1 configuration tool Version 1.4 or higher. No logic is needed in the user program of the CTI PLC.

A function block FB written by Lars Weiß has to be called cyclic within OB1 and in the corresponding "Register DB" S7 memory DBW(16 bit integers) and DBX (bits) are mapped to Modbus registers.

**The FB supports Modbus Functions:**

| | |
|---|---|
| 01h,02h | Read n Bits |
| 03h,04h | Read n Words |
| 05h | Write a Bit |
| 06h | Write a Word |
| 10h | Write n Words |
| 0fh | Write n Bits |

The structure of this "Register DB" is not important. In this example the structure of this datablock consists of 20 integers and the first integer DBxx.DBW0 corresponds to modbusregister 0, DBxx.DBW2 corresponds to modbusregister 1 etc. But if C-memory coils had to be read or written DBxx.DBX0.0 is C1, DBxx.DBX0.1 is C2 etc. A mix of both integers and coils is also possible.

## S7 configuration

Using Step7, the hardware needs to be configured including an Ethernet network. In NetPro an unspecified connection has to be added as shown below.

## Configuring the CP-343

The CP343-1 uses the IP address 10.200.64.93 and is connected to the Ethernet network as shown below

Network configuration 1.
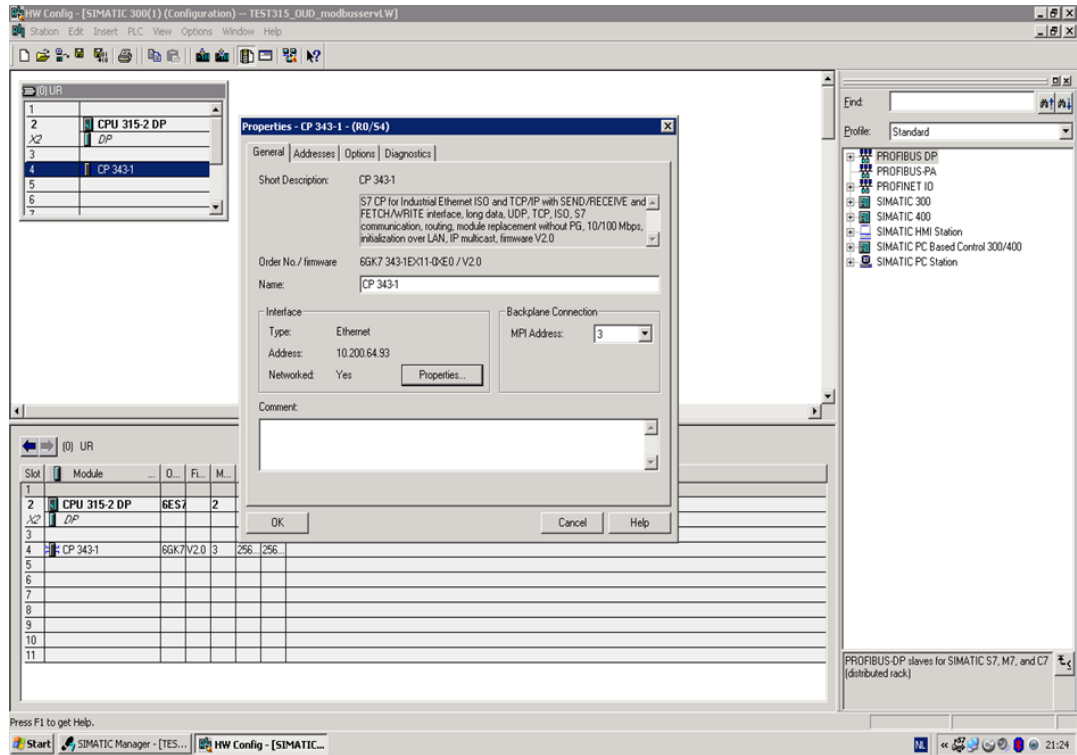


Network configuration 2.
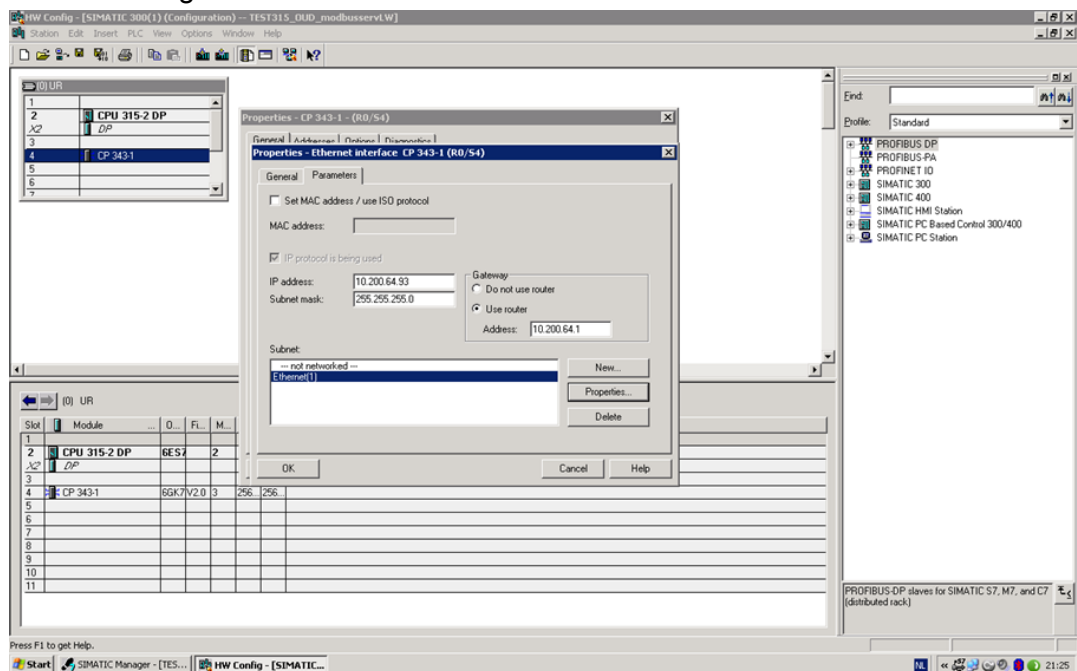
## Configuring the TCP connection for the open Modbus TCP protocol

Under General Information the ID number "1" and the CP LADDR "W#16#100" can be read out.  The "active connection establishment" feature must **not** be selected because the ECC will establish the connection.

For Modbus TCP server as a standard TCP port 502 is used.

Netpro configuration 1.
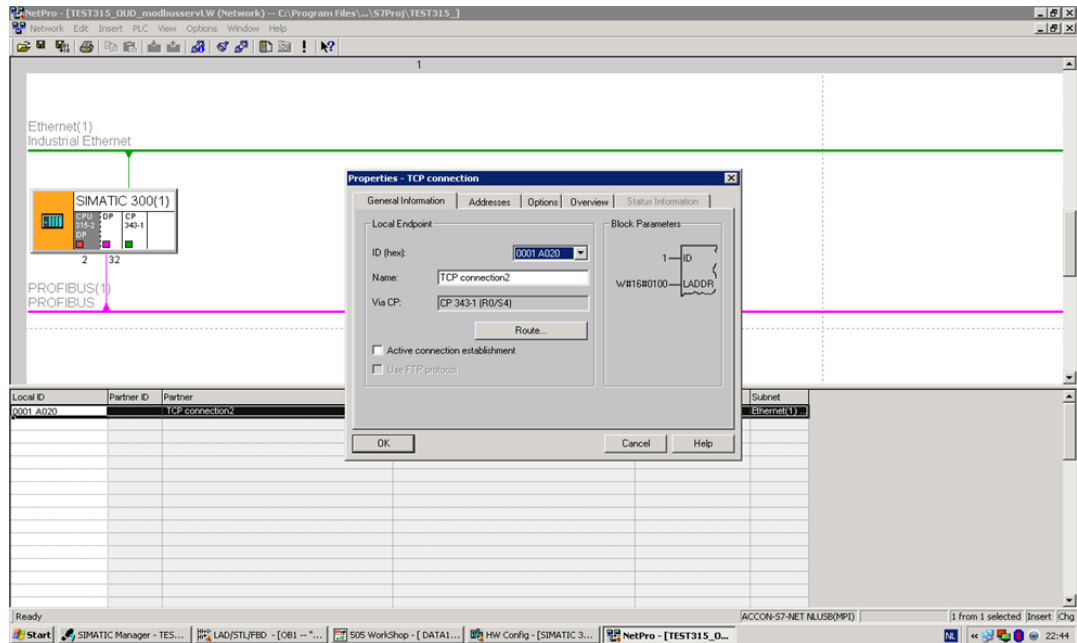


Netpro configuration 2.  Leave the remote fields empty.
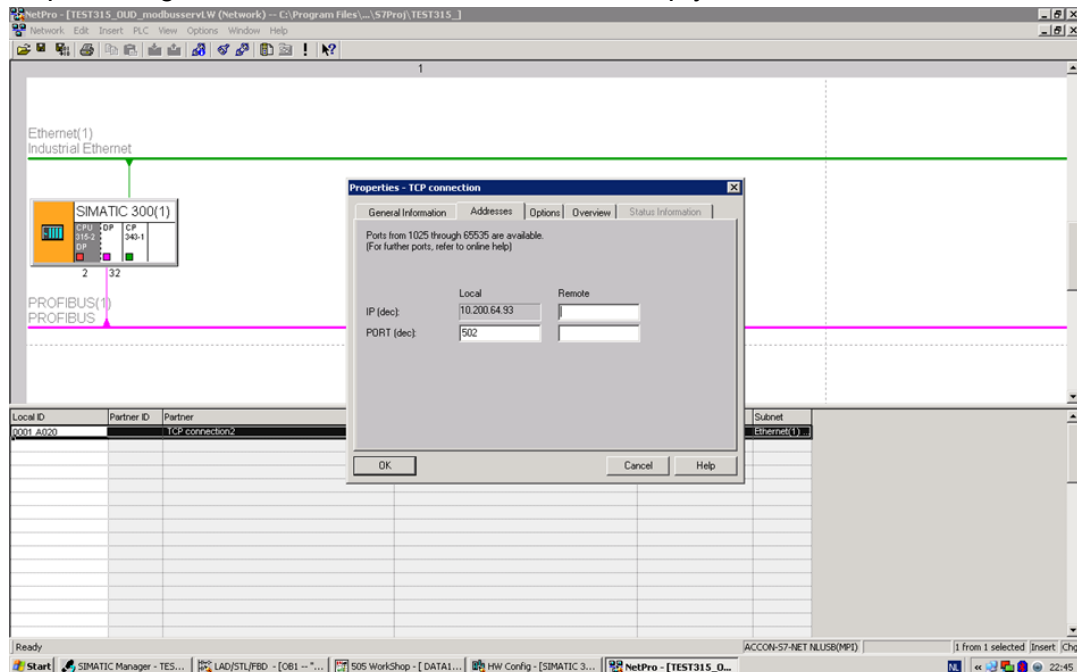


Transfer both Hardware and Netpro connections to the cpu.

## Program with the CP343-1 as server

10 words of data are read from the "Register DB" in the S7 PLC by the ECC and are stored in to V29851 – V29860 of the CTI PLC.

10 words of data V29701-V29710 from the CTI PLC are written by the ECC into the "Register DB" from the S7 PLC.

Program in the S7 plc.
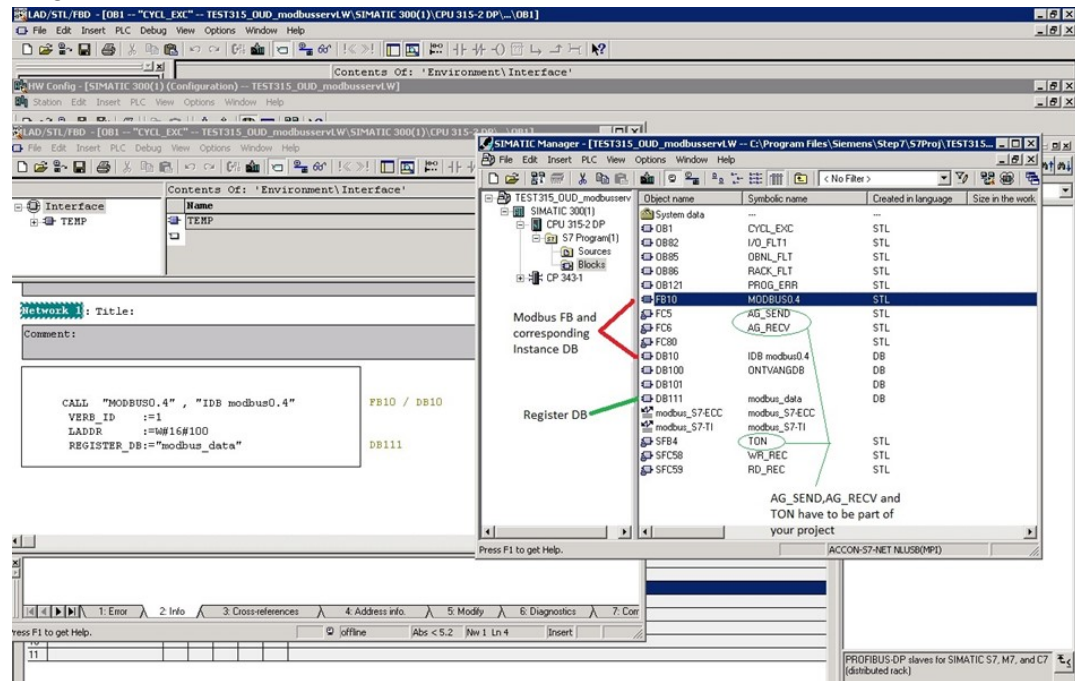


AG_SENDand AG_RCV  can be found in "Libraries - Simatic_NET_CP - CP-300 - (FC5,FC6)"

TON can be found in "Libraries - Standard library - System Function Blocks - (SFB4)"

The "REGISTER_DB" you have to create and in this example we use one that has an array of 40 integers.
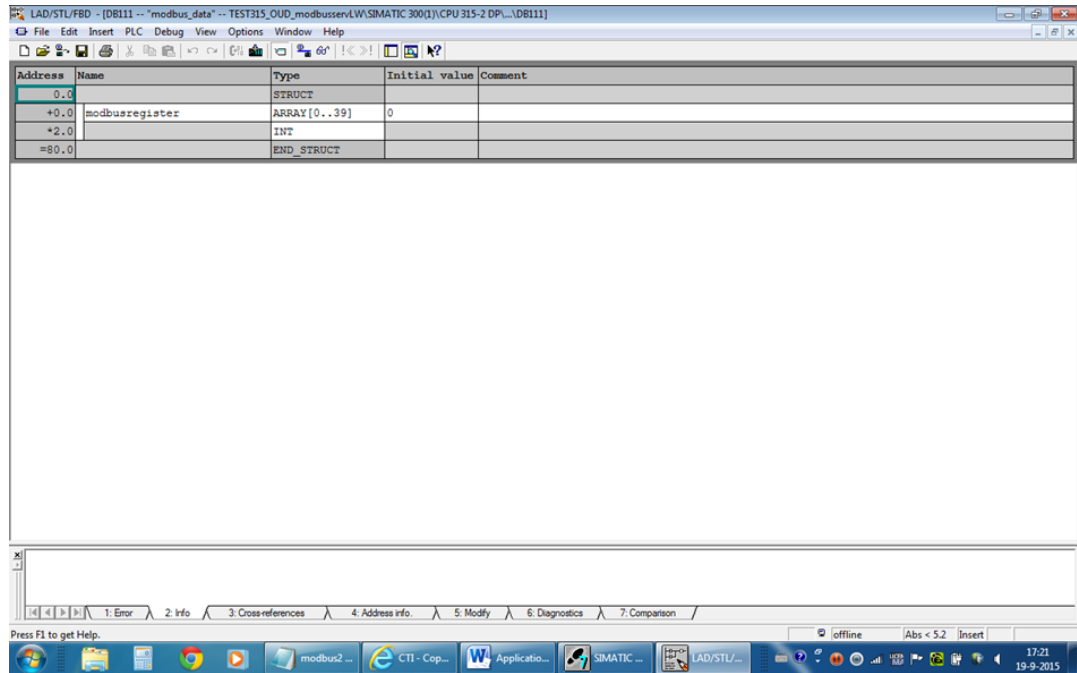


At this point you have the possibility to test the Modbus communication to the S7 PLC by using a Modbus master simulator software tool like for example Modscan 32 or Modbus Poll. These simulation tools can be downloaded from the internet..

## Configuring the 2500P-ECC1

To read and write data from the S7 plc that we previously configured to emulate a Modbus TCP server we need to configure the ECC as a Modbus TCP client and add a Modbus read and write request.

Add a Modbus client connection to the S7 PLC. Use the IP address of the CP343/443 card to configure the Modbus Client connection.

Add a Modbus Unit and Read request.



Add a Tagname out of the "Tagname Database" in this memory area the data read from the S7 PLC will be stored.

Add a Modbus write request.
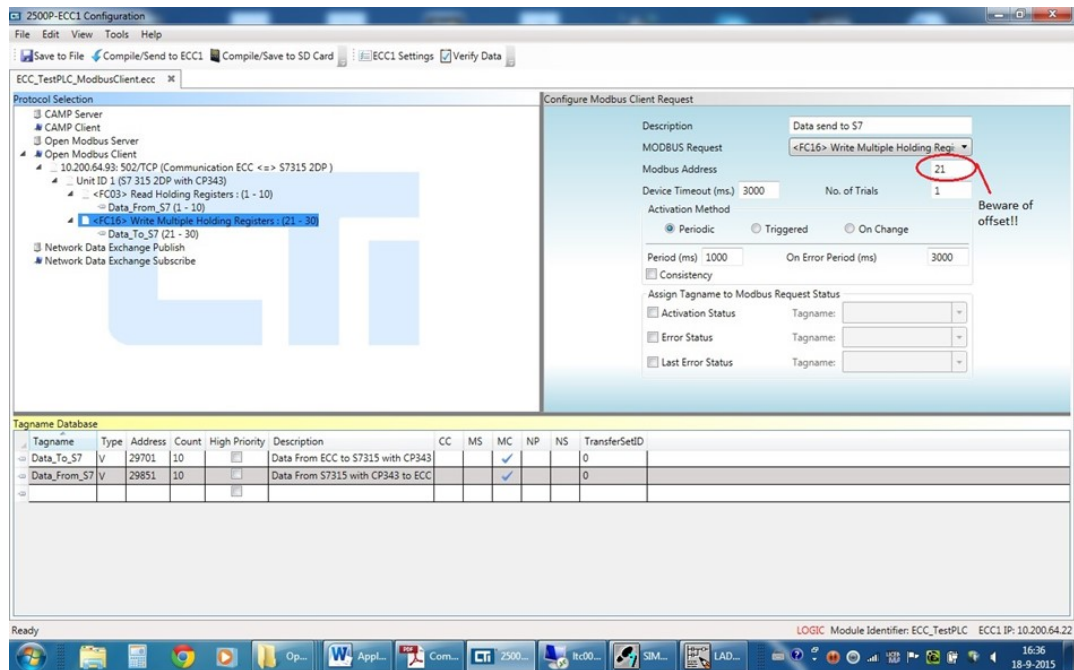


Add a Tagname out of the "Tagname Database". The contents of this memory area will be written to the S7 PLC.
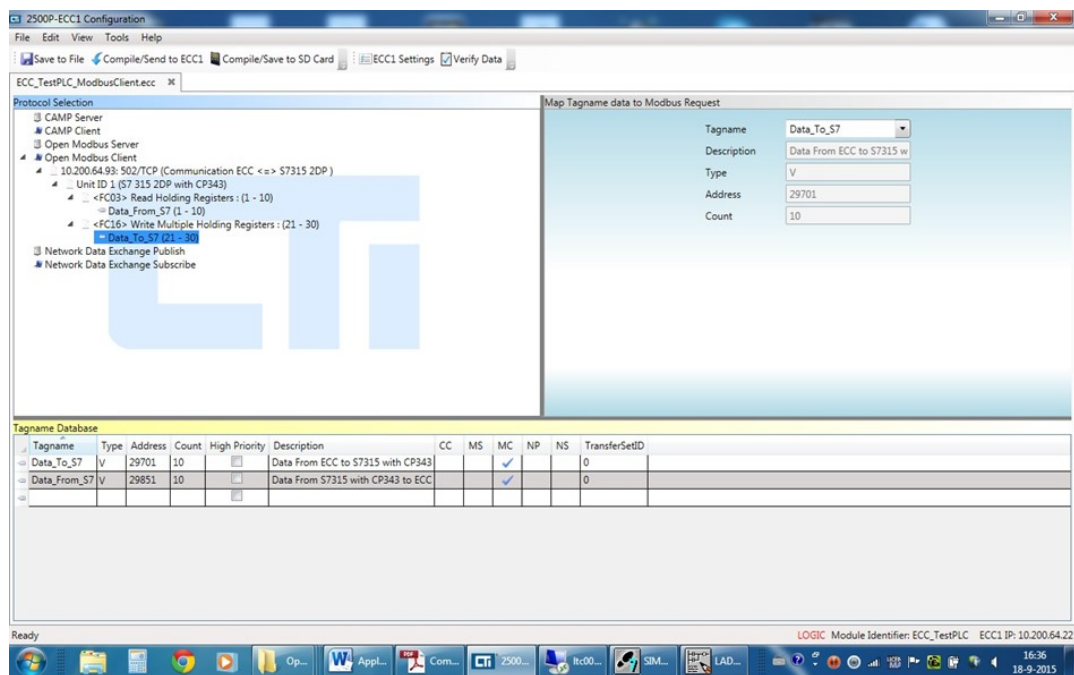


**Transfer the configuration to the 2500P-ECC1 module**

## Test the communication

Data window in the CTI PLC with the Configured communication memory area.



Data window in the S7 PLC with the Configured communication memory area.

Webpage of the ECC showing the active Modbus client connection to the CP343 in the S7 PLC.
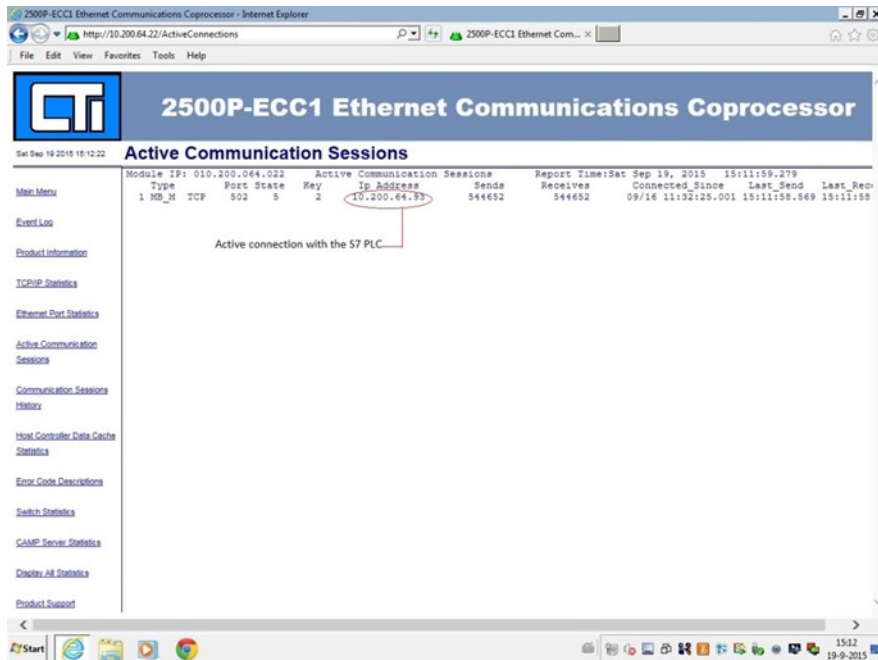


Online diagnostics of the CP343 showing the connection with the ECC.
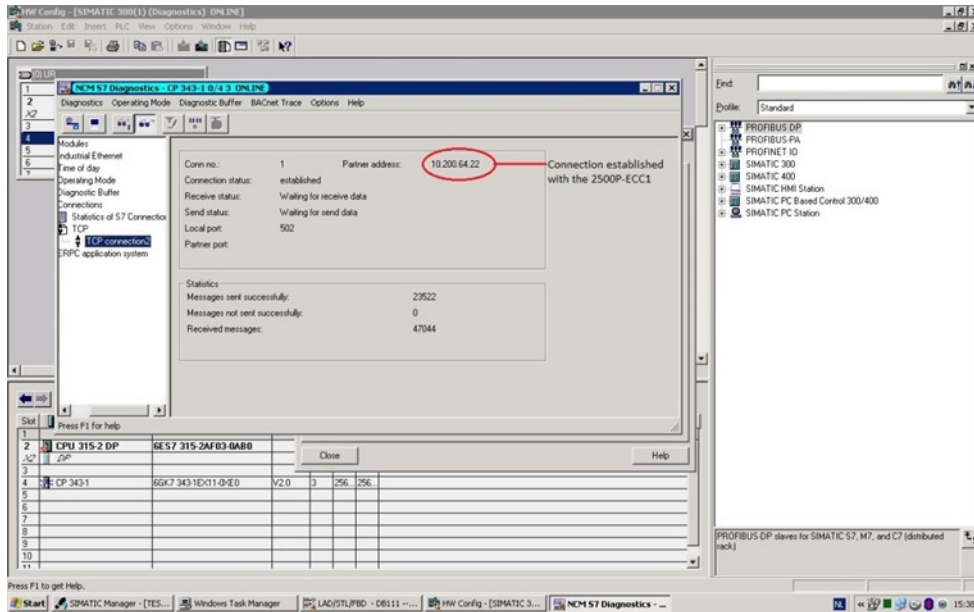
## Example of this communication implemented on the factory floor of a malting plant

The communication has also been tested and running live in a malting plant with 2500P-ECC1 and S7416 with CP443 as shown in the next screenshots.  The S7 PLC is situated in the utility area and is on a different VLAN than the ECC1.

Webpage CP443 with the same modbus configuration.
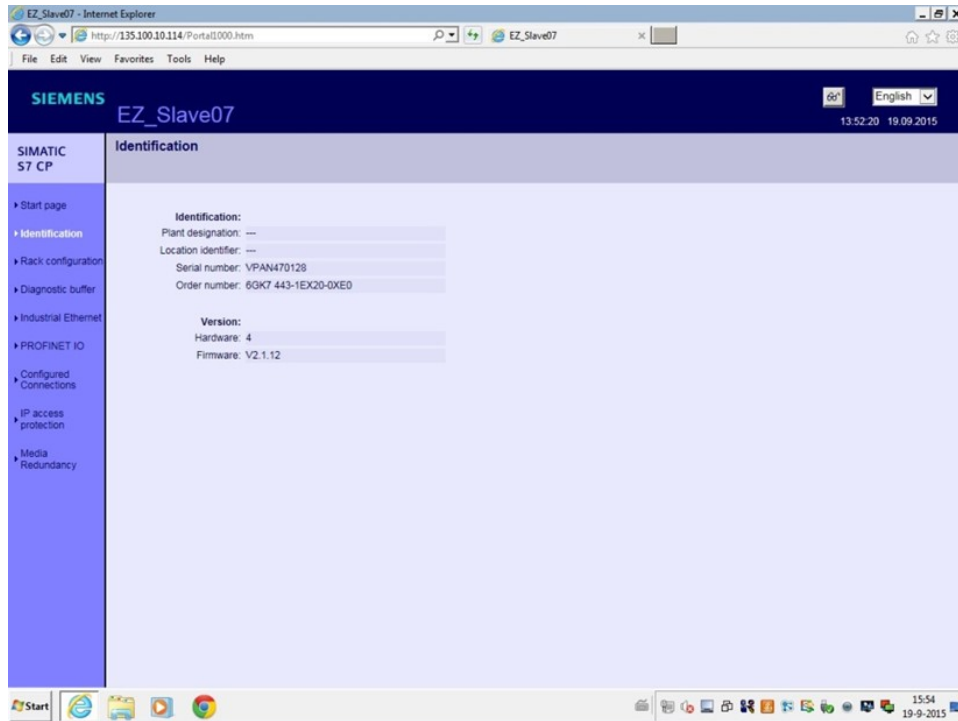
Webpage CP443 with the same modbus configuration IP from CP443 = 135.100.10.114, 2500P-ECC1 = 10.200.64.26



Webpage 2500P-ECC1 showing Modbus communication with the CP443.



Control Technology Inc.
5734 Middlebrook Pike, Knoxville, TN 37921-5962
Phone: +1.865.584.0440    Fax: +1.865.584.5720
www.controltechnology.com