



ATV Process on Ethernet with M221 (I/O profile, using WRITE_READ_VAR)

06/2015



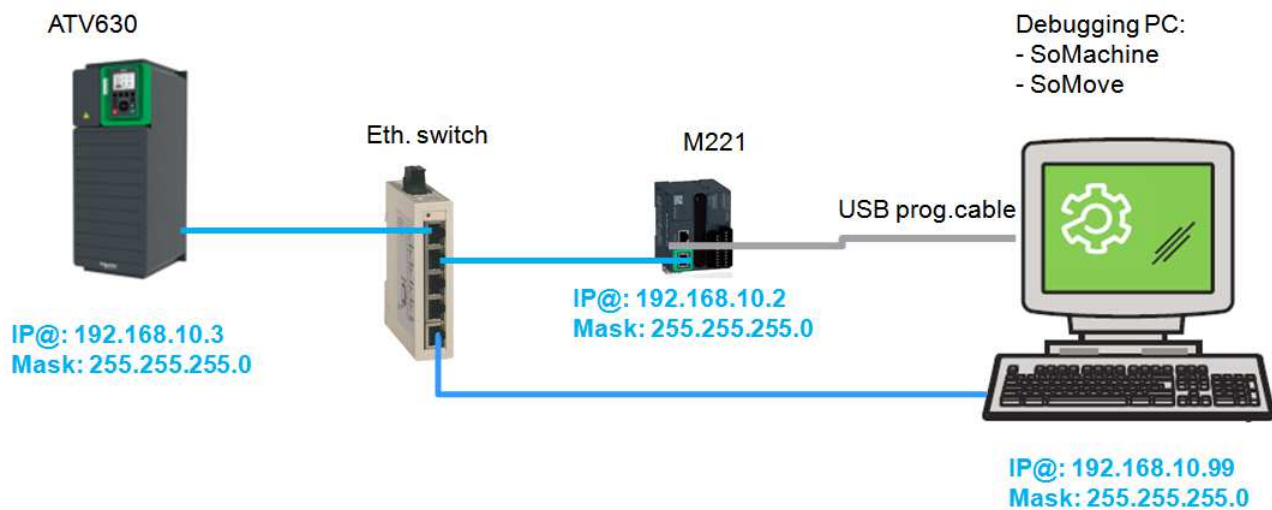
Introduction

This document describes a basic application: **M221** controls the **ATV630** via **Ethernet Modbus TCP/IP**, using **IO profile**.

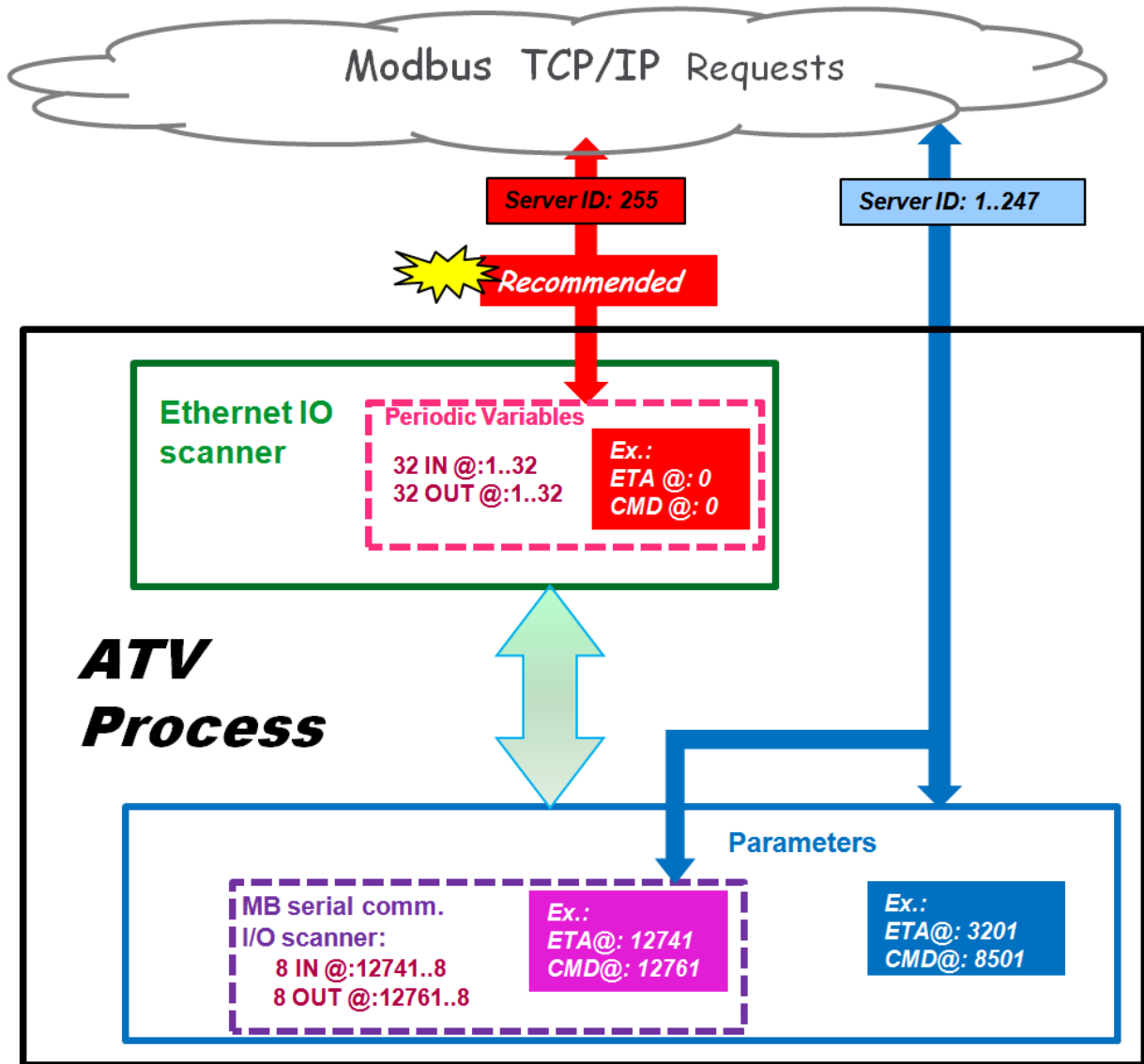
A PC with SoMachine Basic (ver.1.3 SP1) and SoMove (ver.2.2) is also connected to the same network and used as a programming, debugging and configuration tool, both for M221 and ATV630.

Intention was to deliver a kind of '**quick start-up guide**' for the mentioned set of devices and **explanation concerning Modbus TCP/IP servers** built-in to the ATV630

1. Networking scheme:



2. Altivar Process Modbus TCP/IP servers



The embedded Ethernet port has build-in Ethernet I/O scanner service which is exchanging data with the drive in an optimized way – this is the fastest possible way to exchange data via Ethernet with ATV Process.

Access to Ethernet I/O scanner is via **Server ID: 255**.

Access to all drives parameters (except Periodic Variables of Ethernet I/O scanner) is possible via other servers (1..247), however it is not optimized and with lower performance.

Server ID: 248 is reserved for maintenance/debug tools like SoMove – do not use it for communication with PLC's, SCADA's, etc.



3. Data exchange scheme:

Data exchange is mastered by M221 *WRITE_READ_VAR* function
(I/O scanner function - MB function code 23)

ATV Process

[IP@:192.168.10.3](http://192.168.10.3)

M221

[IP@:192.168.10.2](http://192.168.10.2)

INPUTS

1	ETA
2	RFRD
...	
32	



%MW32
%MW33
...
%MW63

OUTPUTS

1	CMD
2	LFRD
...	
32	



%MW0
%MW1
...
%MW31

4. Altivar 71 configuration

For the purpose of this example we use IO profile:

I/O Profile

Using the I/O profile simplifies PLC programming.

The I/O profile mirrors the use of the terminal strip for control by utilizing 1 bit to control a function.

The I/O profile for the drive can also be used when controlling via a fieldbus. The drive starts up as soon as the `run` command is sent. 15 bits of the control word (bits 1...15) can be assigned to a specific function.

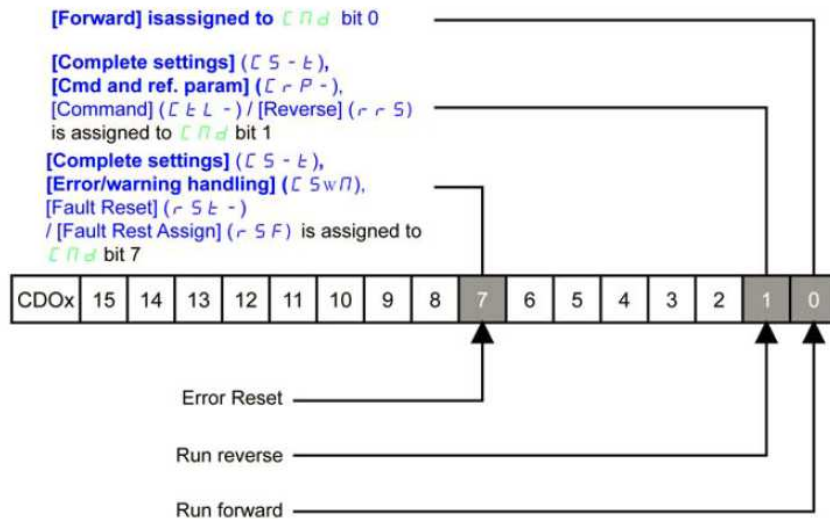
This profile can be developed for simultaneous control of the drive via:

- The terminals
- The Modbus control word
- The CANopen control word
- Ethernet Modbus TCP embedded
- The fieldbus module control word

The I/O profile is supported by the drive itself and therefore in turn by all the communication ports (integrated Modbus, CANopen, Ethernet, PROFIBUS DP, PROFINET, and DeviceNet fieldbus modules).



The I/O profile, here is a simple example, which can be extended with additional features. The command word is made of run forward (bit 0 of CMD), run reverse (bit 1 of CMD), and the function fault reset (bit 7 of CMD).



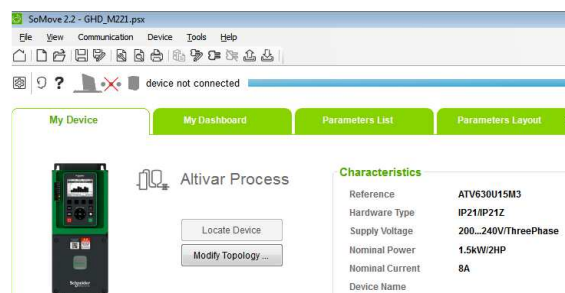
Reference and command are coming from embedded Ethernet.

Set of drive's parameters to organize I/O profile mode:

FR1	[Reference frequency1]	= Embedded Ethernet
CHCF	[Control mode]	= I/O mode
CD1	[Command channel1]	= Ethernet
RRS	[Reverse assignment]	= virtual digital input CMD5.1
RSF	[Fault reset]	= Virtual Digital input CMD5.7
IP assign.mode		= Fixed address
IP address		= 192.168.010.003
Mask		= 255.255.255.000

4.1 Configuration steps:

4.1.1 Using SoMove, create default project for your drive ex.: ATV630U15M3



4.1.2 Change parameters necessary for I/O profile:

Code	Long Label	Current Value
▶ Macro Configuration		
▶ Motor parameters		
▶ Sensors Assignment		
▼ Command and Reference		
FR1	Configuration reference frequency 1	Embedded Ethernet
FR1B	Configuration ref. 1B	Not configured
RCB	Select switching (1 to 1B)	Reference frequency channel 1
RIN	Reverse direction disable	Yes
CHCF	Control mode configuration	I/O mode
CCS	Command switching	Command channel 1
CD1	Command channel 1 assign	Ethernet
CD2	Command channel 2 assign	Modbus communication
RFC	Freq Switching Assignment	Reference frequency channel 1
FR2	Configuration reference frequency 2	Not configured
COP	Copy Ch. 1-Ch.2	No copy
FLOC	Forced Local frequency assignment	Not configured
FLOT	Time-out forc. local	10 s
FLO	Forced local assignment	Not assigned
RRS	Reverse assignment	Virtual Digital input CMD5.1
TCC	2/3-wire control	2-wire control
TCT	Type of 2-wire control	Transition
PST	Stop key enable	Yes
BMP	HMI command	Disabled

Code	Long Label	Current Value
▶ Macro Configuration		
▶ Motor parameters		
▶ Sensors Assignment		
▶ Command and Reference		
▶ Fan		
▶ Generic functions		
▶ Generic monitoring		
▶ Input/Output		
▼ Error/Warning handling		
▶ Auto Fault Reset		
▼ Fault reset		
RSF	Fault reset input assignment	Virtual Digital input CMD5.7
RPA	Product restart assignment	Not assigned

Generic Ethernet Settings

— IP Address —

Device name:

IP assignment mode:

IP address:

Subnet mask:

Gateway address:

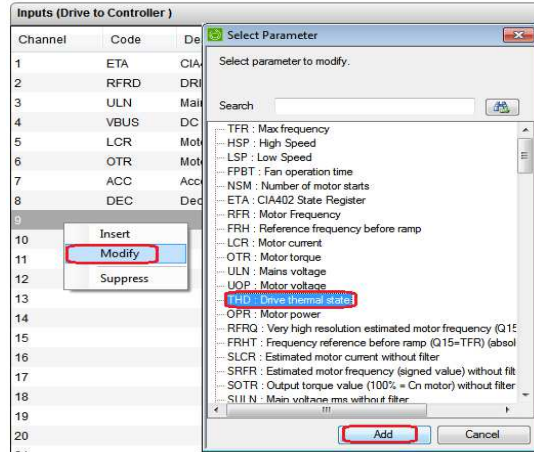
Rate setting:

Storm protection:

Swap word order:

4.1.3 Customize the Ethernet I/O scanner content by:

Select the Inputs/Output row -> right-click -> Modify -> Select from list -> Add



up to your application needs, par ex.:

The screenshot shows the Schneider Electric VSD CCC Expert Center software interface. The 'Parameters List' tab is active, showing the 'Protocol & Drive Profiles' section. The 'Modbus TCP' profile is selected, and the 'Drive I/O Profile' is configured. The 'Inputs (Drive to Controller)' and 'Outputs (Controller to Drive)' tables are displayed.

Channel	Code	Description	Logical address
1	ETA	CIA402 State Register	3201
2	RFRD	DRIVECOM : Actual speed value	8604
3	ULN	Mains voltage	3207
4	VBUS	DC bus voltage	3243
5	LCR	Motor current	3204
6	OTR	Motor torque	3205
7	ACC	Acceleration ramp time	9001
8	DEC	Deceleration ramp time	9002
9			
10			

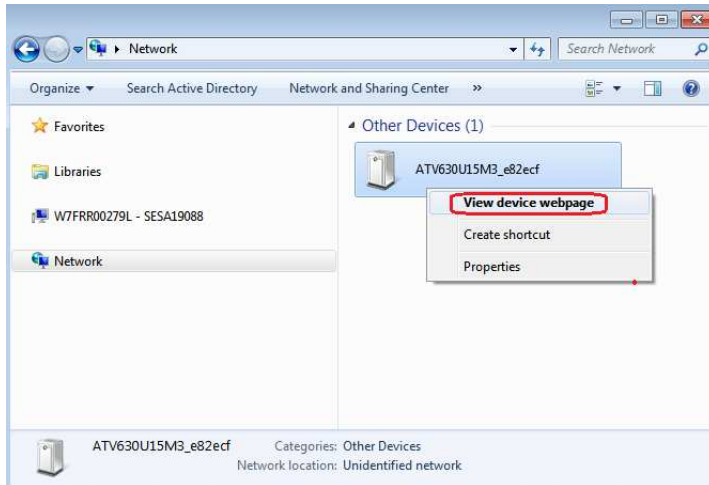
Channel	Code	Description	Logical address
1	CMD	Command register	8501
2	LFRD	DRIVECOM : Nominal speed value	8602
3	ACC	Acceleration ramp time	9001
4	DEC	Deceleration ramp time	9002
5			
6			
7			
8			
9			
10			



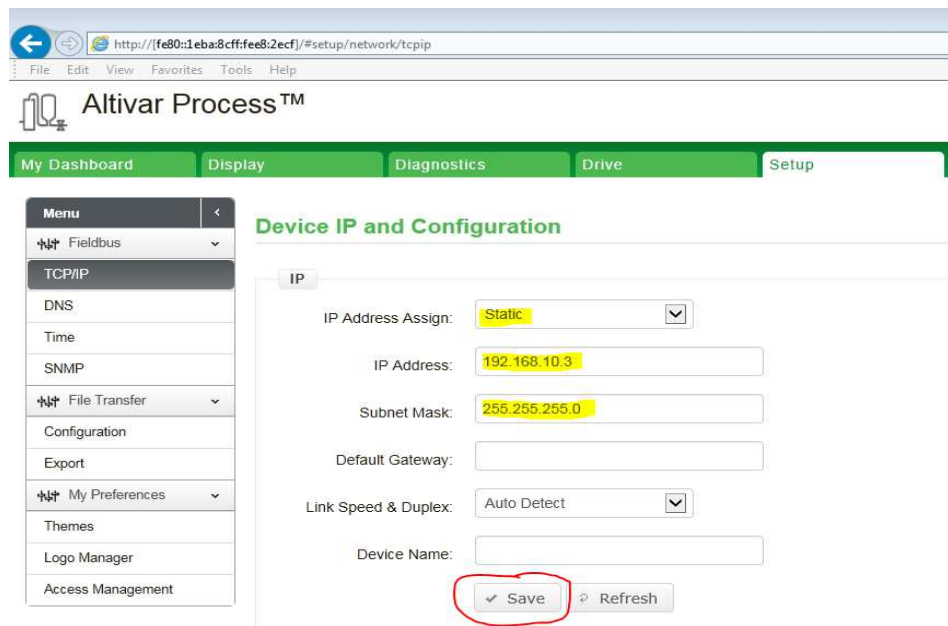
5. Transferring project to ATV Process via Ethernet

5.1 Assigning IP address to ATV630

Discover your ATV6xx on the network using par ex. Windows Explorer:



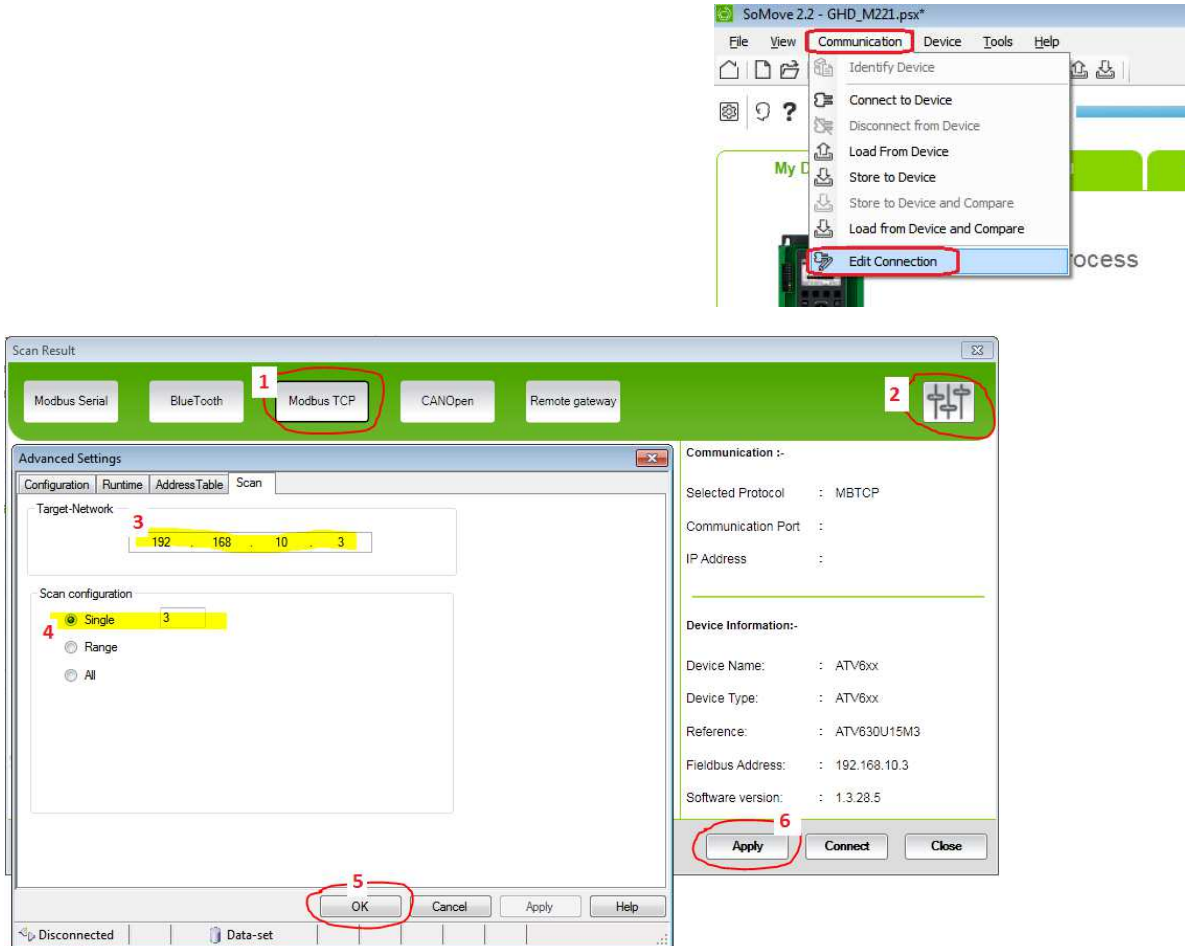
Open ATV6xx's webpage, set the IP@ and save it:



Power cycle the drive.

5.2 Transferring the project to ATV630

Come back to SoMove and set the connection to the drive:



Store the project from SoMove to the drive:

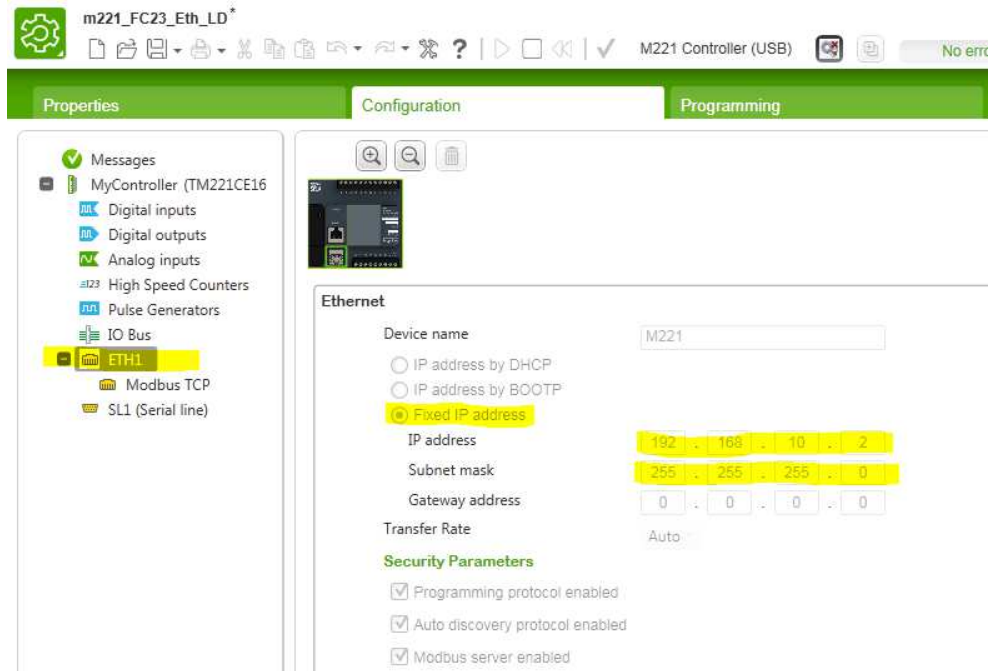


Power cycle the drive.

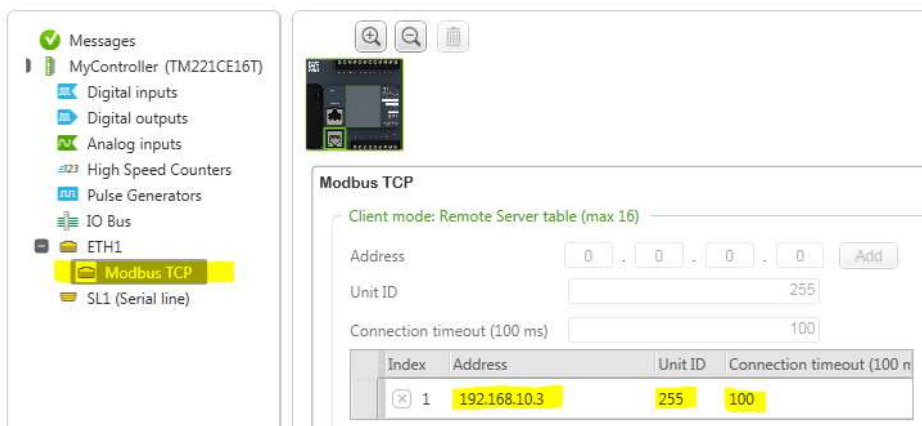


6. PLC application

After selecting the controller, enter its IP@:



Create the ATV6xx server entry, on M221's Modbus TCP Remote Server Table:



Write your program:

The screenshot displays the Schneider Electric software interface for programming a VSD drive. The interface is divided into several tabs: Properties, Configuration, Programming, Display, and Commissioning. The Programming tab is currently selected, showing a ladder logic diagram. The diagram consists of four rungs, each starting with a 'LD' (Load) instruction. Rung 0 contains a function block %WRITE_READ_VAR0 with various inputs and outputs. Rung 1 contains a normally open contact for %M1 (FORWARD). Rung 2 contains a normally open contact for %M2 (REVERSE). Rung 3 contains a normally open contact for %M7 (RESET_ERR). The Master Task configuration is shown at the bottom, with Scan mode set to Normal and Period set to 100 ms.

Write Read Var properties

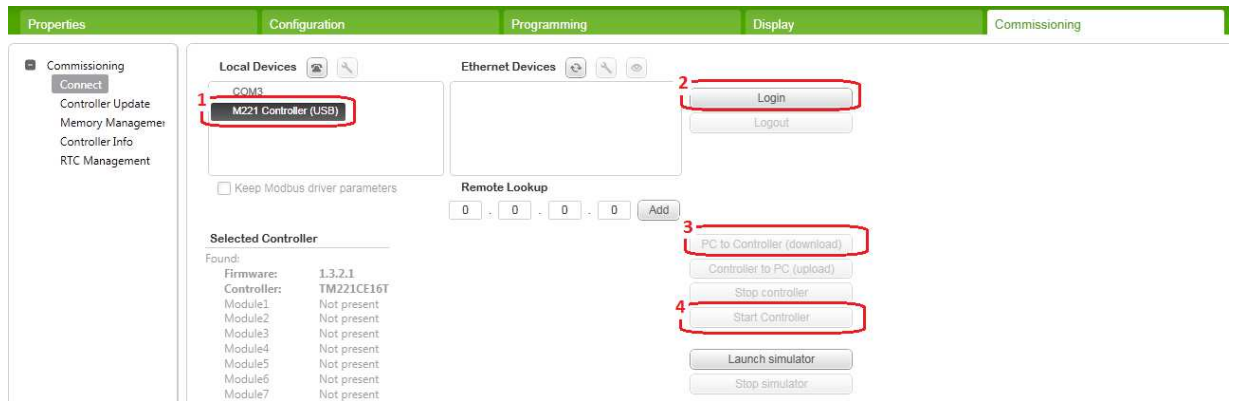
Used	Address	Symbol	Link	Id	Timeout	ObjType	FirstWriteObj	WriteQuantity	IndexDataOut	FirstReadObj	ReadQuantity	IndexDataIn	Comment
<input checked="" type="checkbox"/>	%WRITE_READ_VAR0		3 - ETH1	1	100	0 (Mult. reg. - Mbs 23)	0	32	0	0	32	32	

Explanation:

Bit %M1[Forward] starts/stops forward the drive.
 Bit %M2[Reverse] starts/stops reverse the drive.
 Bit %M3[Reset_Err] resets the drive's faults.
 Word %MW1 [LFRD] is the setup speed for the drive.

Function block %WRITE_READ_VAR0 transfers:
 %MW0..31 → ATV OUTPUTS
 %MW32..63 ← ATV INPUTS

Connect to M221, download the program and run controller:



Debug and check your program, using the Animation Table:

