FeSCADA & PowerFlex 4M

Introduction

An application was done to show the possibilities of FeSCADA. By the end of this paper the reader will learn how FeSCADA can be used to control an AC motor with a PowerFlex 4M variable frequency drive (VFD).

1. Description
2. Hardware
3. RS485 serial communication
4. PowerFlex 4M Modbus registers
5. FeMODBUS communication setup
6. FeSCADA project
7. Conclusions

1) Description

In the following pages an application is developed for controlling and monitoring a PowerFlex VFD. This type of VFD is used extensively in industry to control AC motors. The example is common for other brands of VFDs, because it is an industrial practice for many of VFD producers to use Modbus RTU communication.

2) Hardware

The hardware is composed of one PowerFlex4M variable frequency drive, 22F-V1P6N103, with the following characteristics:

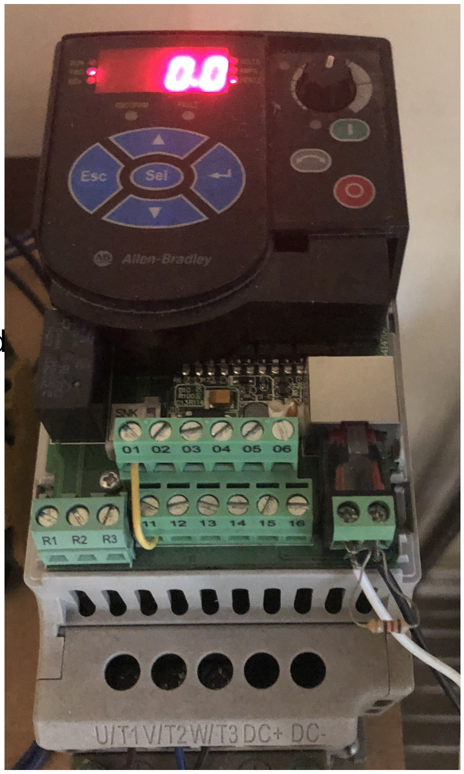
- AC Drive, 0.2 kW (0.25 HP),
  Input: 120V AC, 1 Phase, 50-60 Hz,
  Output: 3 Phase, 1.6 A.

- AC motor, 3 Phase, 230V, 0.25HP, FL 1A

More information at:

https://www.rockwellautomation.com/en-pr/products/details.22F-V1P6N103.html

3) RS485 serial communication

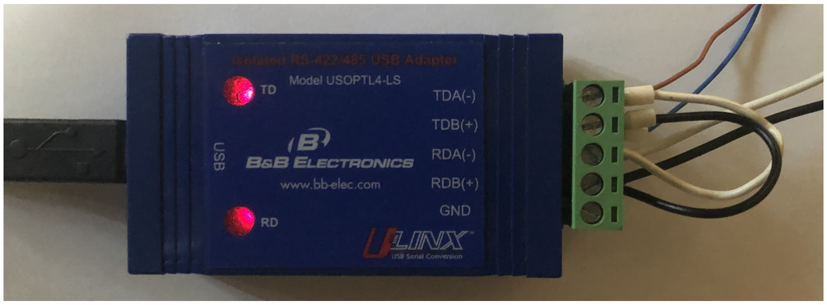


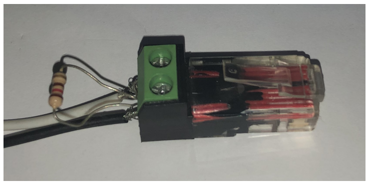
The PowerFlex 4M has one RS485 serial communication port. A small screw terminal block adaptor RJ45 is used to connect the A and B wires. If the VFD is the final device in the RS485 network, a termination resistor of 120Ω needs to be mounted in parallel to the A and B wires.

The PowerFlex VFD is a Modbus RTU server. It is shipped with the default address 100.

The serial communication settings are set with the aid of front panel buttons and display, or by using a dedicated software. The default settings are: 9600 baud, no parity, 8 data and 1 stop bits. We changed the baud rate to 38400.

On the PC side we used an optically isolated USB to RS485 adapter, B&B Electronics USOPTL4-LS.





Hardware prices.

Name	**Unit price**	**Qty**	**Price**	**Description**
22F-V1P6N103	$300.00	1	$300.00	PowerFlex 4M VFD
056H17T2011	$254.00	1	$254.00	Marathon Electric AC motor
USOPTL4-LS	$143.00	1	$143.00	USB to RS485 Adapter
		TOTAL =	**$697.00**	

4) PowerFlex 4M Modbus registers

The following table is showing the Modbus register numbers for various VFD parameters. We distinguish between command registers and status registers.

Parameter	Modbus register number	Description
Logic Command (write)	8192	P106 [Start Source] must be set to 5 in order to accept the commands.
Speed Reference $Hz \cdot 10$ (write or read)	8193	P108 [Speed Reference] must be set to 5 in order to accept the commands.
Logic Status (read)	8448	Status bits from drive (see below)
Output Frequency (read)	8451	The actual speed of the drive
Drive Error Codes (read)	8449	Error code (see below)
Output current (read)	3	Output current present at U, V, W Value $\cdot$ 0.01 A

Logic Command - Bit(s) Description 8192	Logic Status - Bit(s) Description 8448	Error Code – (Decimal) Description 8449
Bit 0 1 = Stop, 0 = Not Stop 1 1 = Start, 0 = Not Start 2 1 = Jog, 0 = No Jog 3 1 = Clear Faults, 0 = Not Clear Faults 5,4 00 = No Command 01 = Forward Command 10 = Reverse Command 11 = No Command ……	Bit 0 1 = Ready, 0 = Not Ready 1 1 = Active (Running), 0 = Not Active 2 1 = Cmd Forward, 0 = Cmd Reverse 3 1 = Rotating Forward, 0 = Rotating Reverse 4 1 = Accelerating, 0 = Not Accelerating 5 1 = Decelerating, 0 = Not Decelerating 6 1 = Alarm, 0 = No Alarm 7 1 = Faulted, 0 = Not Faulted 8 1 = At Reference, 0 = Not At Reference 9 1 = Reference Controlled by Comm 10 1 = Operation Cmd Controlled by Comm 11 1 = Parameters have been locked 12 Digital Input 1 Status 13 Digital Input 2 Status 14 Not Used 15 Not Used	0 No Fault 2 Auxiliary Input 3 Power Loss 4 Undervoltage 5 Overvoltage 6 Motor Stalled 7 Motor Overload 8 Heatsink Overtemperature 12 HW Overcurrent (300%) 13 Ground Fault 29 Analog Input Loss 33 Auto Restart Tries 38 Phase U to Ground Short 39 Phase V to Ground Short 40 Phase W to Ground Short 41 Phase UV Short 42 Phase UW Short 43 Phase VW Short 63 Software Overcurrent 64 Drive Overload 70 Power Unit Fail 80 AutoTune Fail 81 Communication Loss

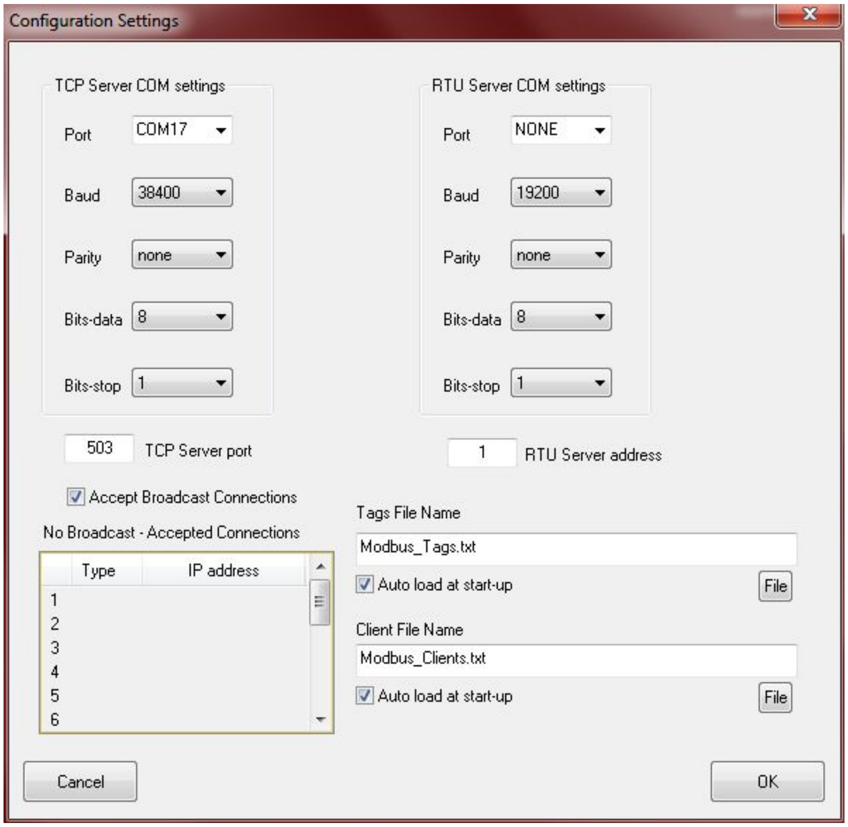
5) FeMODBUS communication setup

FeMODBUS software is used to connect to PowerFlex 4M.

In the picture on the right the serial connection settings are presented.

COM17 was assigned for our USB to RS485 adapter by Windows, when we plugged in the adapter. The other settings are matching the values from PowerFlex 4M.

The next settings are used to connect to the server and to send requests.



Configuration Settings

TCP Server COM settings

Port	COM17
Baud	38400
Parity	none
Bits-data	8
Bits-stop	1

RTU Server COM settings

Port	NONE
Baud	19200
Parity	none
Bits-data	8
Bits-stop	1

503 TCP Server port

1 RTU Server address

☑ Accept Broadcast Connections

No Broadcast - Accepted Connections

	Type	IP address
1		
2		
3		
4		
5		
6		

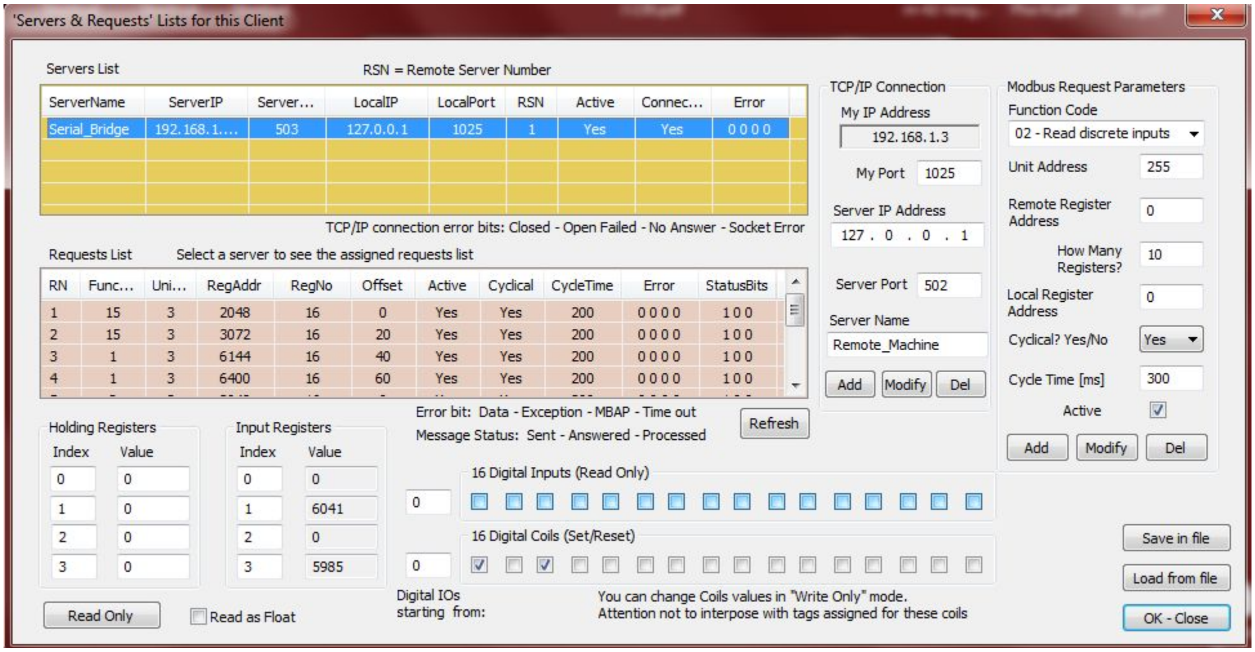
Tags File Name: Modbus_Tags.txt

☑ Auto load at start-up

Client File Name: Modbus_Clients.txt

☑ Auto load at start-up

We connect to our own PC, which it is a Modbus TCP server, and we send requests to the address 100. Because the address 100 is less than 255, the server will send them onto the serial COM port assigned for TCP server, and will send the answers back.



'Servers & Requests' Lists for this Client

Servers List — RSN = Remote Server Number

ServerName	ServerIP	Server...	LocalIP	LocalPort	RSN	Active	Connec...	Error
Serial_Bridge	192.168.1....	503	127.0.0.1	1025	1	Yes	Yes	0 0 0 0

TCP/IP connection error bits: Closed - Open Failed - No Answer - Socket Error

Requests List — Select a server to see the assigned requests list

RN	Func...	Uni...	RegAddr	RegNo	Offset	Active	Cyclical	CycleTime	Error	StatusBits
1	15	3	2048	16	0	Yes	Yes	200	0 0 0 0	1 0 0
2	15	3	3072	16	20	Yes	Yes	200	0 0 0 0	1 0 0
3	1	3	6144	16	40	Yes	Yes	200	0 0 0 0	1 0 0
4	1	3	6400	16	60	Yes	Yes	200	0 0 0 0	1 0 0

Error bit: Data - Exception - MBAP - Time out

Message Status: Sent - Answered - Processed

TCP/IP Connection

My IP Address	192.168.1.3
My Port	1025
Server IP Address	127 . 0 . 0 . 1
Server Port	502
Server Name	Remote_Machine

Modbus Request Parameters

Function Code	02 - Read discrete inputs
Unit Address	255
Remote Register Address	0
How Many Registers?	10
Local Register Address	0
Cyclical? Yes/No	Yes
Cycle Time [ms]	300
Active	☑

Holding Registers

Index	Value
0	0
1	0
2	0
3	0

Input Registers

Index	Value
0	0
1	6041
2	0
3	5985

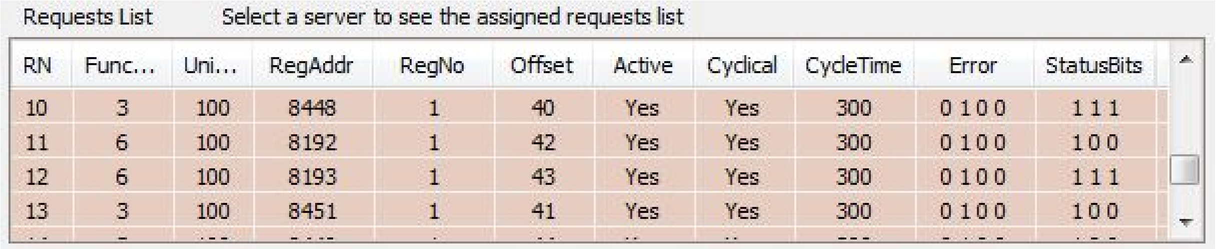
16 Digital Inputs (Read Only): 0 □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □

16 Digital Coils (Set/Reset): 0 ☑ □ ☑ □ □ □ □ □ □ □ □ □ □ □ □ □

Digital IOs starting from:

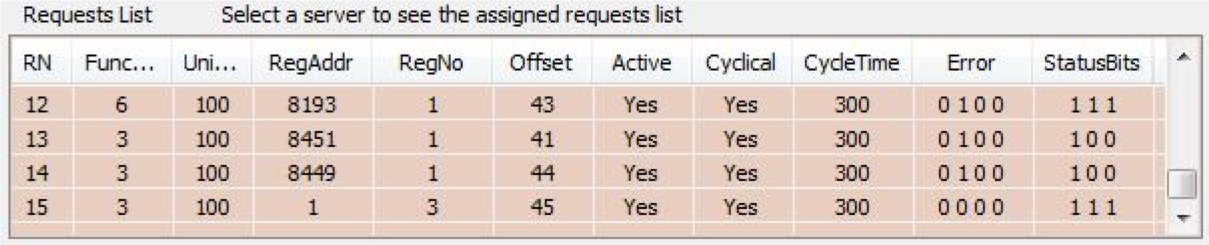
You can change Coils values in "Write Only" mode.
Attention not to interpose with tags assigned for these coils

□ Read as Float



Requests List   Select a server to see the assigned requests list

RN	Func...	Uni...	RegAddr	RegNo	Offset	Active	Cyclical	CycleTime	Error	StatusBits
10	3	100	8448	1	40	Yes	Yes	300	0 1 0 0	1 1 1
11	6	100	8192	1	42	Yes	Yes	300	0 1 0 0	1 0 0
12	6	100	8193	1	43	Yes	Yes	300	0 1 0 0	1 1 1
13	3	100	8451	1	41	Yes	Yes	300	0 1 0 0	1 0 0



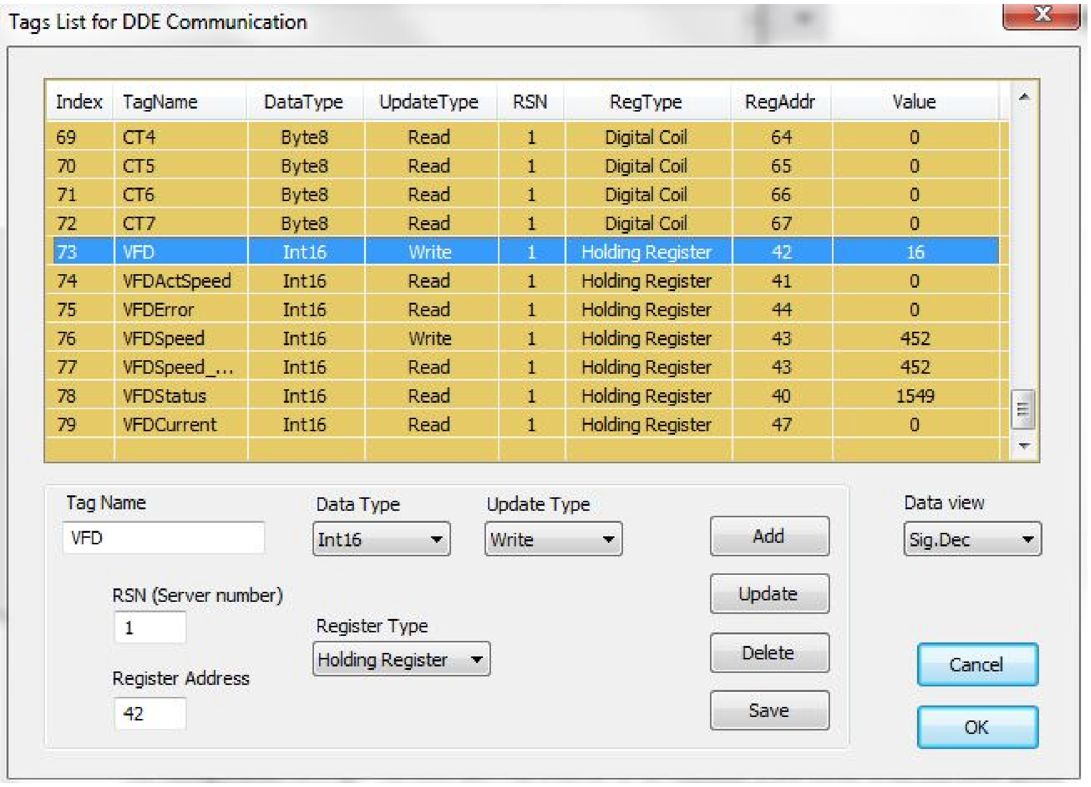
Requests List   Select a server to see the assigned requests list

RN	Func...	Uni...	RegAddr	RegNo	Offset	Active	Cyclical	CycleTime	Error	StatusBits
12	6	100	8193	1	43	Yes	Yes	300	0 1 0 0	1 1 1
13	3	100	8451	1	41	Yes	Yes	300	0 1 0 0	1 0 0
14	3	100	8449	1	44	Yes	Yes	300	0 1 0 0	1 0 0
15	3	100	1	3	45	Yes	Yes	300	0 0 0 0	1 1 1

The pictures above are showing that FeMODBUS was setup to connect at the address 192.168.1.3, which is our own PC address, and to cyclically (every 300ms) send some of the possible requests:

- Function 3 – read one or multiple holding registers
- Function 6 – write one holding register
- Function 16 – write one or multiple holding registers

After setting up all the requests we defined tags which we linked with different registers on the local computer. The tags are used for DDE communication between FeMODBUS and FeSCADA. All the tags are assigned to the Remote Server Number 1 (RSN 1).



Tags List for DDE Communication

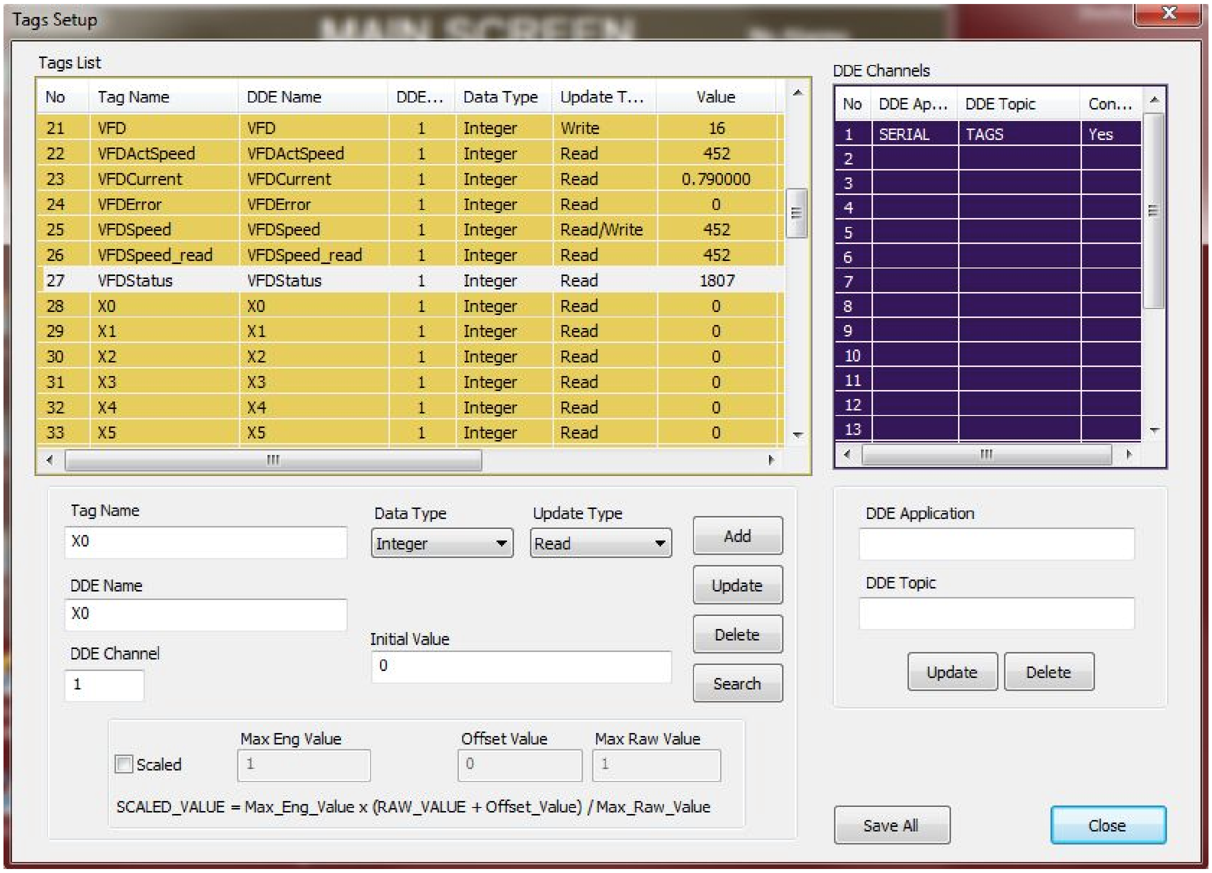
Index	TagName	DataType	UpdateType	RSN	RegType	RegAddr	Value
69	CT4	Byte8	Read	1	Digital Coil	64	0
70	CT5	Byte8	Read	1	Digital Coil	65	0
71	CT6	Byte8	Read	1	Digital Coil	66	0
72	CT7	Byte8	Read	1	Digital Coil	67	0
73	VFD	Int16	Write	1	Holding Register	42	16
74	VFDActSpeed	Int16	Read	1	Holding Register	41	0
75	VFDError	Int16	Read	1	Holding Register	44	0
76	VFDSpeed	Int16	Write	1	Holding Register	43	452
77	VFDSpeed_...	Int16	Read	1	Holding Register	43	452
78	VFDStatus	Int16	Read	1	Holding Register	40	1549
79	VFDCurrent	Int16	Read	1	Holding Register	47	0

Tag Name
VFD
Data Type
Int16
Update Type
Write
Data view
Sig.Dec
RSN (Server number)
1
Register Type
Holding Register
Register Address
42

6) FeSCADA project

The first step in a FeSCADA project is to define the DDE communication channels and the tags. In the picture below we defined the DDE channel 1(one) as: DDE_Application = “SERIAL” and DDE_Topic = “TAGS”.

Every tag has an internal name used in FeSCADA and a DDE Name for communication with the “SERIAL” DDE server. We kept the names the same. All the tags defined in FeMODBUS will have a correspondent tag in FeSCADA.



No	Tag Name	DDE Name	DDE...	Data Type	Update T...	Value
21	VFD	VFD	1	Integer	Write	16
22	VFDActSpeed	VFDActSpeed	1	Integer	Read	452
23	VFDCurrent	VFDCurrent	1	Integer	Read	0.790000
24	VFDError	VFDError	1	Integer	Read	0
25	VFDSpeed	VFDSpeed	1	Integer	Read/Write	452
26	VFDSpeed_read	VFDSpeed_read	1	Integer	Read	452
27	VFDStatus	VFDStatus	1	Integer	Read	1807
28	X0	X0	1	Integer	Read	0
29	X1	X1	1	Integer	Read	0
30	X2	X2	1	Integer	Read	0
31	X3	X3	1	Integer	Read	0
32	X4	X4	1	Integer	Read	0
33	X5	X5	1	Integer	Read	0

No	DDE Ap...	DDE Topic	Con...
1	SERIAL	TAGS	Yes
2			
3			
4			
5			
6			
7			
8			
9			
10			
11			
12			
13			

Tag Name: X0
Data Type: Integer
Update Type: Read
DDE Name: X0
DDE Channel: 1
Initial Value: 0

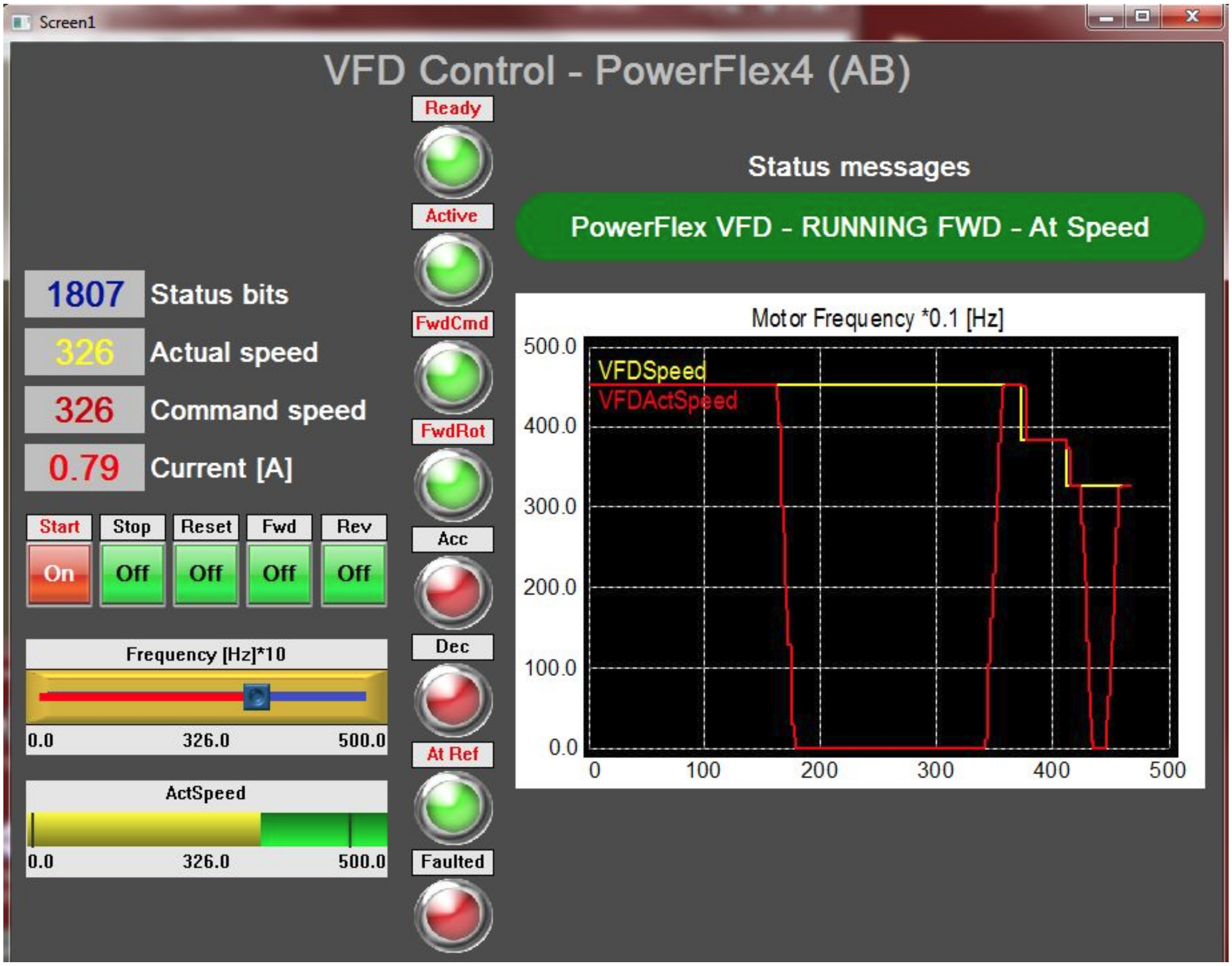
☐ Scaled
Max Eng Value: 1
Offset Value: 0
Max Raw Value: 1

SCALED_VALUE = Max_Eng_Value x (RAW_VALUE + Offset_Value) / Max_Raw_Value

DDE Application:
DDE Topic:

Now we can build a screen (window) to show the data, and to put some indicators and buttons. In the picture below one can see a snapshot of this screen. We used:

- 5 buttons for commands: Start, Stop, Reset, Fwd, and Rev
- 1 slider to adjust the commanded drive speed from 0 to 50 Hz
- 1 analog indicator to show the actual speed
- 8 digital indicators to show different bits from the status word
- 4 numeric displays to show status word, actual and command speed and current
- 1 text message display to translate status word in text messages
- 1 trend to show the evolution of the commanded and actual speed in time.



Screen1

VFD Control - PowerFlex4 (AB)

Status messages

PowerFlex VFD - RUNNING FWD - At Speed

1807 Status bits

326 Actual speed

326 Command speed

0.79 Current [A]

7) Conclusions

The application has shown an example of using FeSCADA and FeMODBUS to control an AC motor with a PowerFlex 4M variable frequency drive.

The communication used is serial RS485. The protocol is Modbus RTU.