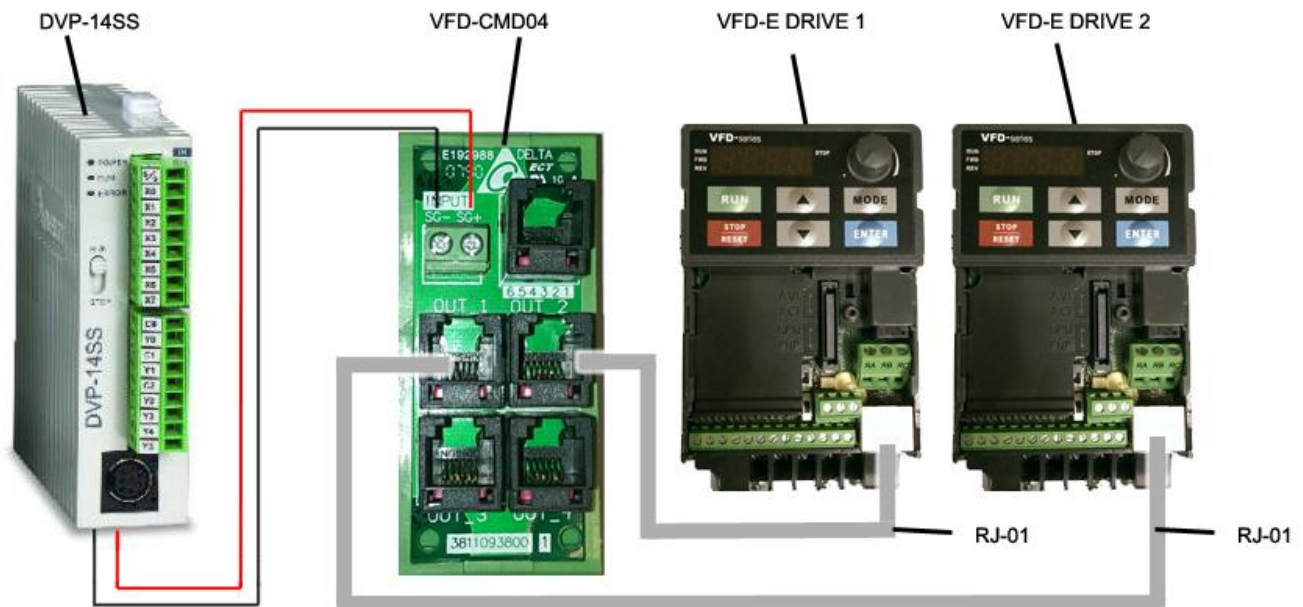


SOFTSTART UK

PLC – VFD-E Communications – RS-485 RTU Network

1. Hardware

In this example I will use a PLC to communicate to multiple drives to send run, stop and frequencies.



2. Decide the protocol to be used.

Write the protocol down on paper avoids future problems, in this example I am going to use (8, E, 1, 9600, RTU)

3. Set up the Inverter drives

The important thing in this step is get ready the inverter to receive commands from the RS-485 interface,

All Drive Parameters

- 2.00 - Source of First Master Frequency Command
= **3: (RS-485 communication)**
- 2.01 - Source of First Operation Command
= **3: (RS-485 Communication and keypad STOP/RESET key enabled)**
- 9.01 - Transmission Speed
= **1: (Baud rate 9600bps)**
- 9.04 - Communication Protocol
= **4: (8, E, 1, Modbus, RTU)**

VFD-E Drive 1 Parameters

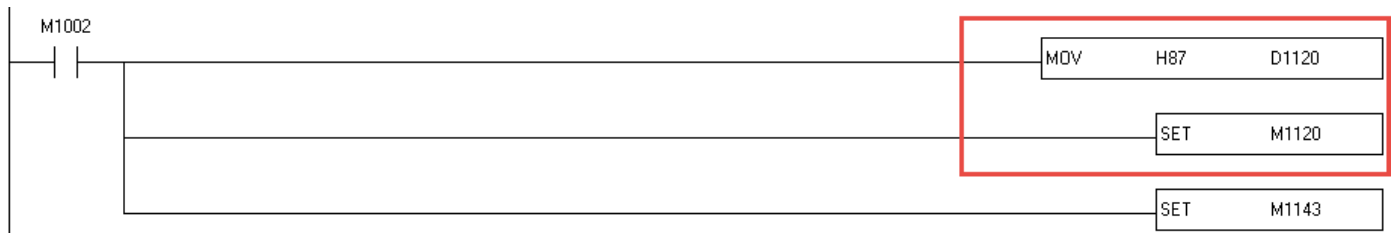
- 9.00 - Communication Address
= **1: (Address of the second drive)**

VFD-E Drive 2 Parameters

- 9.00 - Communication Address
= **2: (Address of the second drive)**

4. Configure the PLC Communication.

In this example I am using a Delta DVP-14SS PLC and will need to set up communications in order to talk to the inverter. We do this on first scan by using the special relay M1002 and set up the communications protocol in D1120. In this example we are using (8, E, 1, 9600), which equates to H87.



BINARY								BINARY TO DECIMAL	DECIMAL TO HEX
B7	B6	B5	B4	B3	B2	B1	B0		
1	0	0	0	0	1	1	1	K135	H87

Using the tables below work out what value you need set for D1120. You can double click on the table above and update the binary values to show what the values are to set.

B7	B6	B5	B4	Baud
0	0	0	1	110
0	0	1	1	300
0	1	0	0	600
0	1	0	1	1200
0	1	1	0	2400
0	1	1	1	4800
1	0	0	0	9600
1	0	0	1	19200
1	0	1	0	38400
1	0	1	1	57600
1	1	0	0	115200

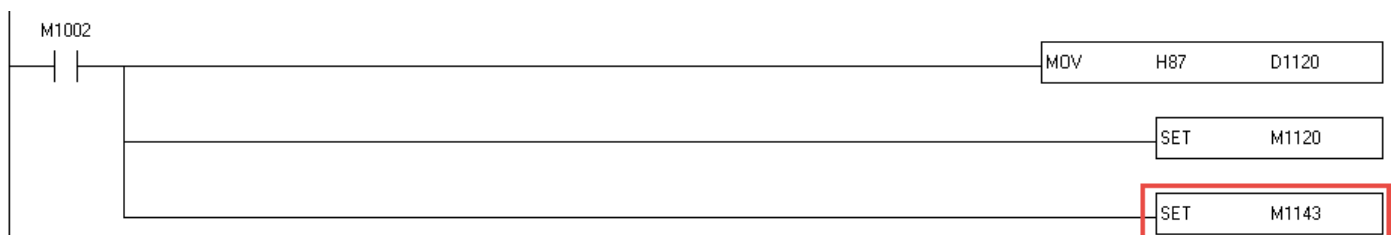
B0	DATA LENGTH
0	7
1	8

B2	B1	Parity
0	0	None
0	1	Odd
1	1	Even

B3	Stop Bits
0	1 Bit
1	2 Bits

Once we have set our communication protocol we need to retain it and we do this by setting M1120.

We now need to tell the PLC to use RTU Mode and we do this by setting M1143.



5. PLC Software (send).

Now we have set up our communications we can begin to send some commands through the PLC RS-485 port, we're going to use the instruction MODWR (API 101) and the special mark M1122 to trigger the communication.

The MODWR has the next composition:

API 101 MODWR Modbus Data Write

MODWR	S ₁	S ₂	n	
				S ₁ : External device address (K0~K254)
				S ₂ : Register(s) of external device to write to
				n : Number of values to write

Operand: [Device Range](#)

S₁ : K, H, D

S₂ : K, H, D

n : K, H, D

Explanations:

- MODWR is a drive instruction exclusively for peripheral communication equipment in MODBUS ASCII mode/RTU mode. The built-in RS-485 communication ports in Delta VFD drives (except for VFD-A series) are all compatible with MODBUS communication format. MODRD can be used for controlling communication (write data) of Delta drives.
- The feedback (returned) data from the peripheral equipment will be stored in D1070 ~ D1076. After receiving the feedback data is completed, PLC will auto-check if all data are correct. If there is an error, M1140 will be On.

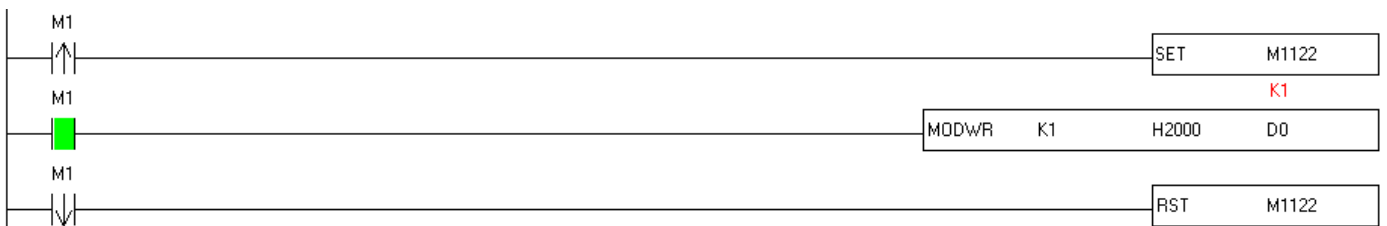
MODWR S1 S2 N

S1 = Device Address (k1 for VFD-E Drive 1 Address or K2 for VFD-E Drive 2 Address)

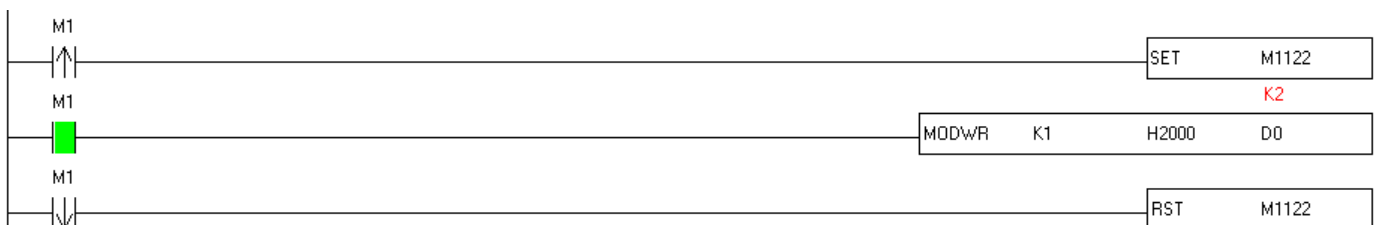
S2 = Data Address (H2000)

N = Data to be written (D0 just 1 word to Stop D0=K1/ Run D0=K2)

Stop



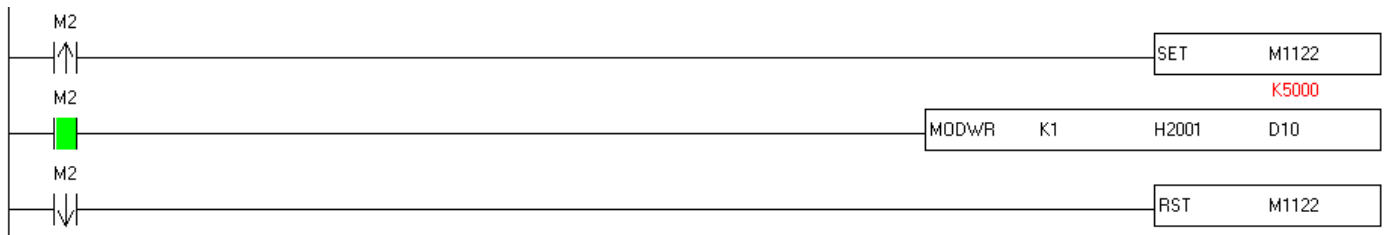
Run



Set Frequency

MODWR S1 S2 N

- S1 = Device Address (k1 for VFD-E Drive 1 Address or K2 for VFD-E Drive 2 Address)
 S2 = Data Address (H2001)
 N = Data to be written (D10 just 1 word to change frequency D10 = K5000 50Hz)



6. PLC Software (read).

To read some data through the PLC RS-485 port, we are going to read the output frequency, in this example we're going to use the instruction MODRD (API 100) and again the special mark M1122 to trigger the communication.

The received data are stored in D1070 ~ D1085 in ASCII format. PLC will automatically convert the data into numerals and store them in D1050 ~ D1055. From this we can read our frequency.

The MODRD has the next composition:

API 100 MODRD Modbus Data Read

MODRD	S ₁	S ₂	n
-------	----------------	----------------	---

S₁ : External device address (K0~K254)

S₂ : Register(s) of external device to read

n : Number of registers to read $K1 \leq n \leq K6$

Operand: [Device Range](#)

S₁ : K, H, D

S₂ : K, H, D

n : K, H, D

Explanations:

- MODRD is a drive instruction exclusively for peripheral communication equipment in MODBUS ASCII mode/RTU mode. The built-in RS-485 communication ports in Delta VFD drives (except for VFD-A series) are all compatible with MODBUS communication format. MODRD can be used for controlling communication (read data) of Delta drives.
- The feedback (returned) data from the peripheral equipment will be stored in D1070 ~ D1085. After receiving the feedback data is completed, PLC will auto-check if all data are correct. If there is an error, M1140 will be On.

MODRD S1 S2 N

- S1 = Device Address (k1 as we configured in step 3)
 S2 = Data Address of the slave (H2103 to read output frequency)
 N = Data to be read (k10 just 1 word to read frequency)

Read Frequency



Using M1127 we can reset the flag when sending and receiving of the data is complete.

NOTICE

Use of the software and program examples within this document, follow the terms set in our software disclaimer. This is viewable on our website. (<http://www.softstartuk.com/terms-and-conditions#software>)

SOFTSTART UK
www.softstartuk.com

Softstart UK Limited
14 Brinell Way
Great Yarmouth
Norfolk
NR31 0LU

Tel: +44 (0) 1493 660510 | Fax: +44 (0) 1493 660511 | Email: sales@softstartuk.com

V1.0 CH