CHURCHILL CONTROLS
www.churchill-controls.co.uk

Mega_Link 2

APPLICATION NOTE AN035
Using Modbus for PLC to PLC Register Transmission

Summary

In addition to transferring 'Real-World' physical data, Mega_Link can also interface directly to a PLC via the COM3A/3B serial port using the Modbus RTU protocol.

This can be done at both the Basestation and Outstation and therefore a Mega_Link 2 458 MHz licence free Radio, 4G or Ethernet TCP/IP link can be used to transmit register contents from PLC to PLC.

This document outlines examples of how this can be performed.

This document assumes a PLC with Modbus RTU style of communications and covers either Master or Slave modes.

This document does not offer specific details of any particular PLC manufacturer/types and instead uses commercially available Modbus RTU simulation software to demonstrate typical PLC operation.

Please contact us for example configuration files, simulation files and technical support.

1. Introduction

1.1 Modbus RTU

Many PLCs have the ability to communicate with other devices via a Modbus RTU serial link. This is generally done at one of two electrical interface standards – RS232 or RS485. The PLC manufacturer also defines the protocol (language) used for communication. The device to which it is linked, along with the data rate and format must match this.

The Modbus RTU protocol operates on a master/slave basis, whereby a master device can 'talk' to one or more slaves. All communication is instigated by the master, and each slave responds only to the commands addressed to it.

Mega_Link 2 has been designed to interface using Modbus RTU. The COM3A/COM3B serial interface port is included within each Mega_Link 2, and is named Fieldbus/Bus_Link to signify that it is a general-purpose port for communication with various PLC busses. It can be configured to operate using the Modbus RTU protocol in either master or slave mode, in various different data rates and serial format, and can use either RS232 or RS485 signal levels.

1.2 Example Demonstration Process

Commercially available software called Modbus Poll is used to simulate a PLC working as a Master and Modbus Slave will be used to simulate a PLC working as a Master.

These are available from:

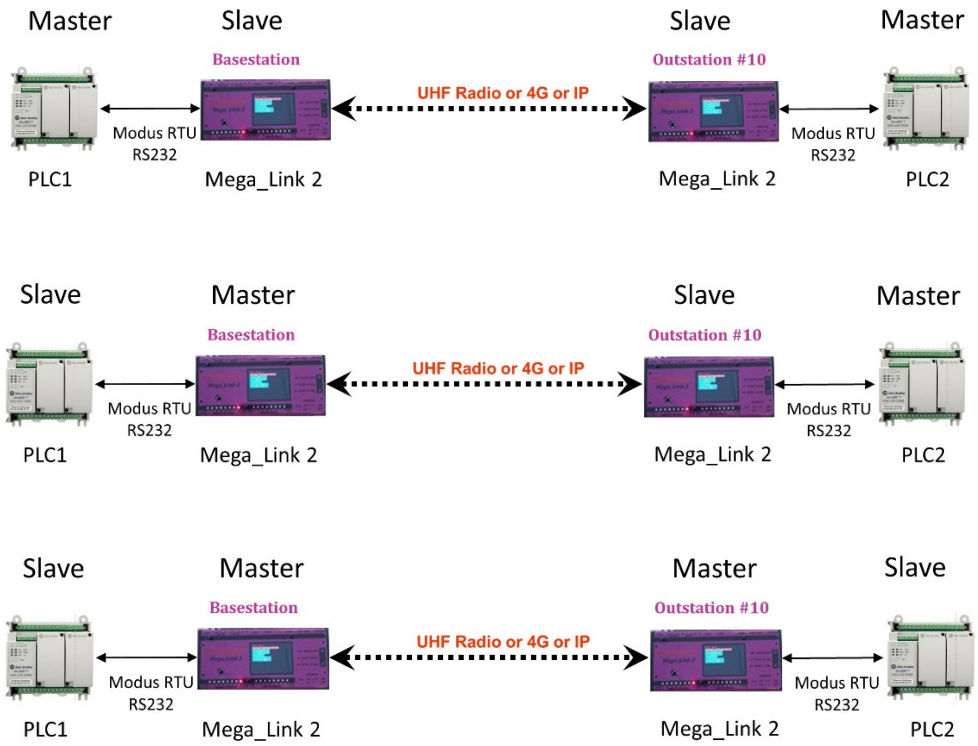
https://www.modbustools.com/modbus_poll.html

https://www.modbustools.com/modbus_slave.html

1.3 Overview of this document

This document gives examples of how to transfer 24 register values between two PLCs in both directions.

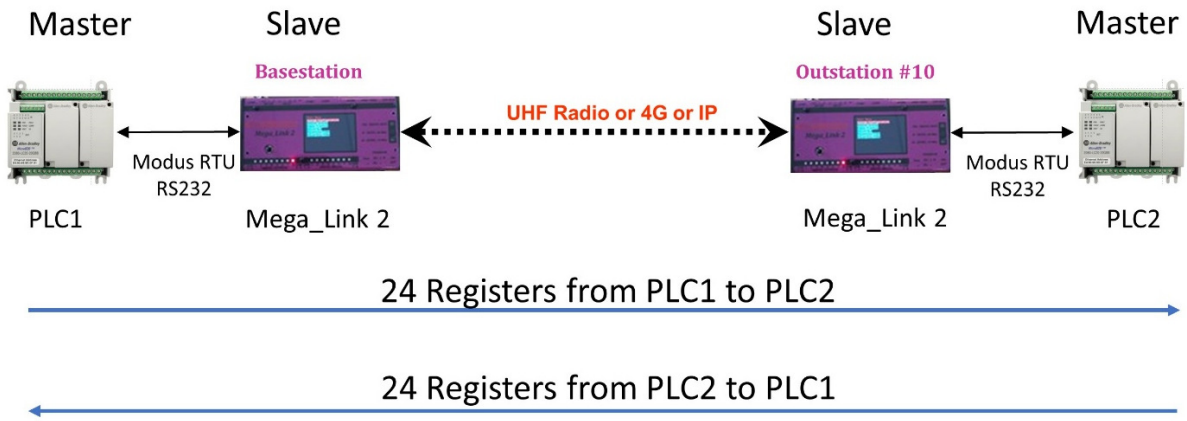
Three examples are described in sections 2, 3 & 4 of this document covering arrangements with different combinations of Master and Slave as shown.



2. Register Transfers Between PLC1 (Master) and PLC2 (Master)

2.1 Overview

This example shows how to transfer 24 Registers from PLC1 to PLC2 and 24 Registers from PLC2 to PLC1 when both PLCs are set up as Masters and the Basestation and Outstation #10 Mega_Link 2s are set in slave mode.

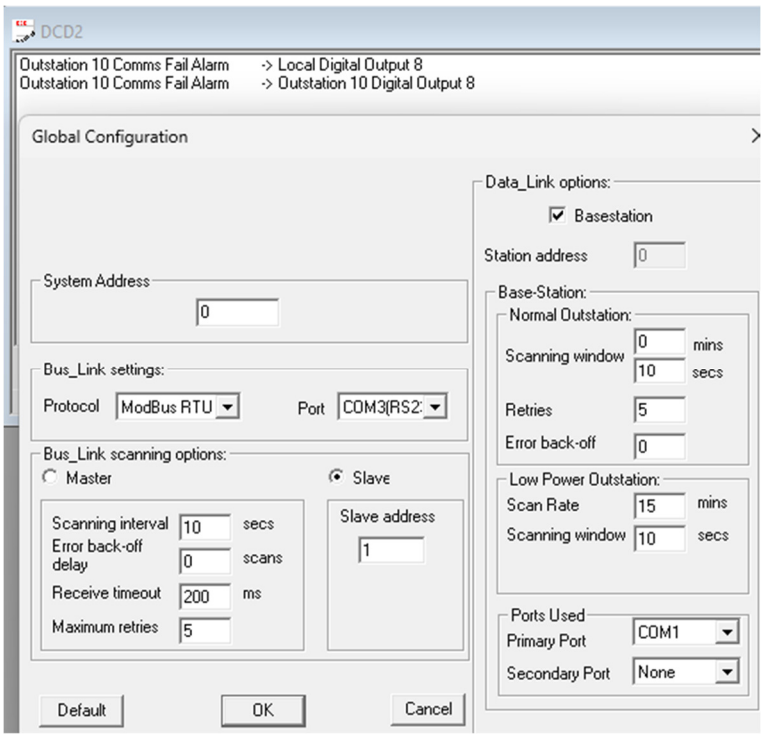


2.2 Mega_Link 2 Configurations

2.2.1 Basestation (Slave mode) Configuration

The Basestation is configured as a Slave using COM3 (RS232) with Slave address = 1.

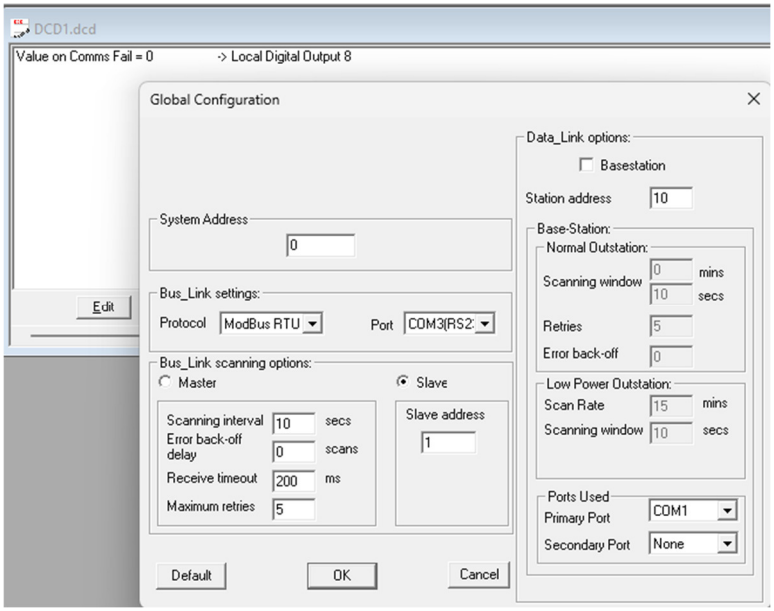
In this example the Basestation data routing configuration is trivial. Just the basic comms fail lines are required.



2.2.2 Outstation #10 (Slave Mode) Configuration

The Outstation #10 is configured as a Slave using COM3 (RS232) with Slave address = 1.

In this example the Outstation data routing is simply the basic Comms Fail output value on comms fail action.



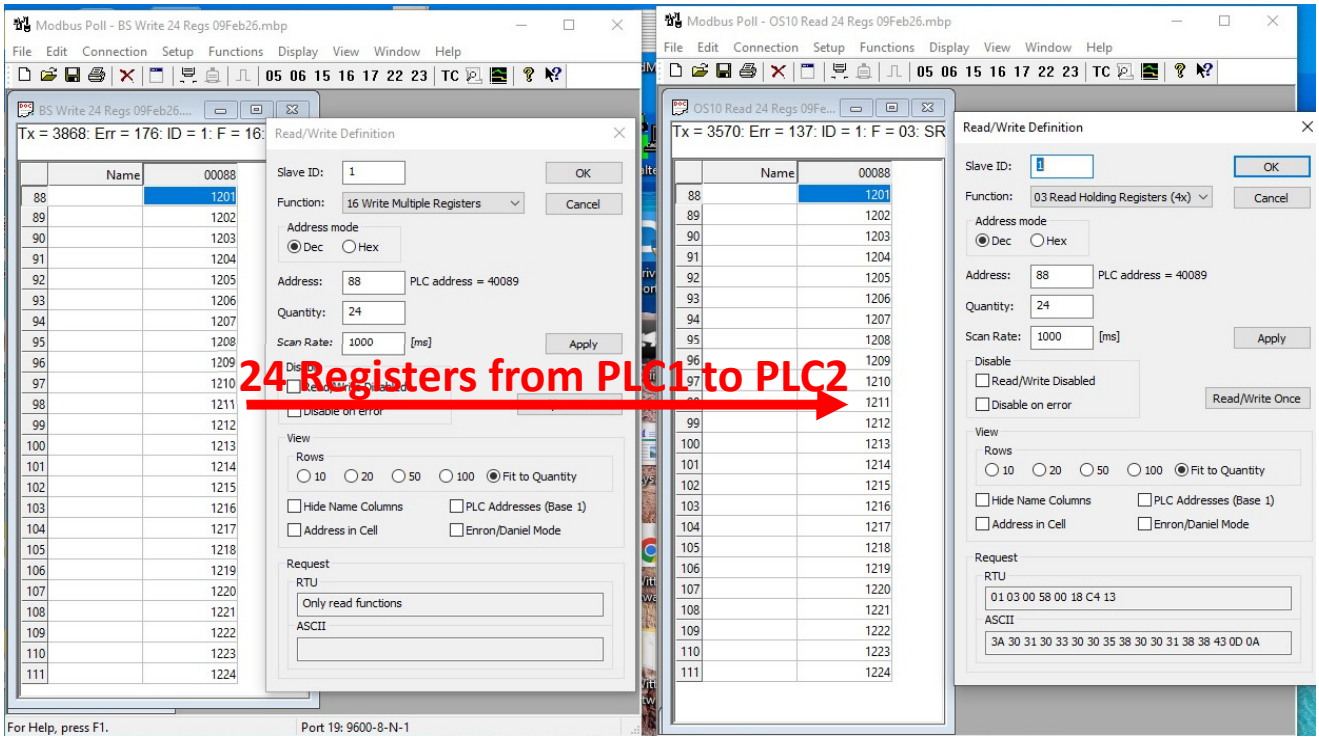
2.3 PLC1 (Master) to PLC2 (Master) Direction

2.3.1 PLC1 to PLC2 Operation

In this example Modbus Poll is used to simulate the PLC1 (Master) at the Basestation on the left and another instance of Modbus Poll is used to simulate PLC2 (Master) at the Outstation #10, on the right of screen grab.

The Modbus Poll on the left, simulating PLC1 is used to write 24 register values into the Basestation with destination starting address #88 by means of the “16 Write Multiple Registers” command. In this example the data values to be written are 1201, 1202, 1203……1224.

The Modbus Poll on the right, simulating PLC2 is used to read out 24 register values from the Outstation #10 at starting address #88 by means of the “03 Read Holding Registers (4x)” command. In this example shown the data values which are read out and displayed are 1201, 1202, 1203……1224, as expected.



2.3.2 Explanation

The address of the registers corresponding to Outstation #10 is starting address 80 and these physically exist in Outstation #10. The next block of 8 registers is address 88, these are virtual registers and these actually correspond to Outstation #11. For the transfer of 24 registers a further two blocks of 8, i.e. Outstation #12 and #13 are utilised.

The Basestation is already polling Outstation #10 because of the comms fail data routing and writing 24 registers into Basestation with output destination address 88 therefore automatically causes the Basestation to also poll virtual Outstations #11, #12 and #13, via Outstation #10.

Here is the "D" command in diagnostics at the Basestation:

```
Outstation: 10
10:49:08  TX: 01 1A 00 00 0A 0C 00 80 00 00 00 00 00 00 00 00 00 00 00 BB FF 00 00 00 00 0A 04 A4 55 9B 09 EB 51
Time-out set to 4700ms.
Primary comms rcvd pkt
10:49:10  RX: 01 16 00 0A 00 0D 7F 00 00 00 10 00 08 00 07 00 09 00 ED 06 BD FF 4C 00 45 00 01 00 0F 6B
    2800ms left. Good reply.


1 sec to next poll.

Outstation: 11
10:49:18  TX: 01 1A 00 00 0B 0C 00 00 00 00 B1 04 B2 04 B3 04 B4 04 B5 04 B6 04 B7 04 B8 04 0A 04 AE 55 9B 09 C7 F7
Time-out set to 4700ms.
Primary comms rcvd pkt
10:49:20  RX: 01 16 00 0B 00 0D 00 00 00 00 35 08 36 08 37 08 38 08 39 08 3A 08 3B 08 3C 08 03 00 7E 71
    3000ms left. Good reply.


1 sec to next poll.

Outstation: 12
10:49:28  TX: 01 1A 00 00 0C 0C 00 00 00 00 B9 04 BA 04 BB 04 BC 04 BD 04 BE 04 BF 04 C0 04 0A 04 B8 55 9B 09 07 2B
Time-out set to 4700ms.
Primary comms rcvd pkt
10:49:30  RX: 01 16 00 0C 00 0D 00 00 00 00 3D 08 3E 08 3F 08 40 08 41 08 42 08 43 08 44 08 03 00 A7 F7
    3000ms left. Good reply.

1 sec to next poll.

Outstation: 13
10:49:38  TX: 01 1A 00 00 0D 0C 00 00 00 00 C1 04 C2 04 C3 04 C4 04 C5 04 C6 04 C7 04 C8 04 0A 04 C2 55 9B 09 DB 9C
Time-out set to 4700ms.
Primary comms rcvd pkt
10:49:40  RX: 01 16 00 0D 00 0D 00 00 00 00 45 08 46 08 47 08 48 08 49 08 4A 08 4B 08 4C 08 03 00 CE A4
    2800ms left. Good reply.

secs
Scanned 4 Outstations with 0 failures
```

Here is the **OR** (Output Registers) diagnostics command **at the Basestation** (the database of the system states).

You can see that the data 1201, 1202, 1203….1224 is being sent to Outstation **Output** registers starting at address 88.

OUTPUT REGISTERS

If analogues not 0..20mA, scale as necessary

Reg No.	Block No.	0	1	2	3	4	5	6	7
0:	0:	0	0	0	0	0	0dBm	0.00mA	0.00mA
8:	1:	-	-	-	-	-	-	-	-
16:	2:	-	-	-	-	-	-	-	-
24:	3:	-	-	-	-	-	-	-	-
32:	4:	-	-	-	-	-	-	-	-
40:	5:	-	-	-	-	-	-	-	-
48:	6:	-	-	-	-	-	-	-	-
56:	7:	-	-	-	-	-	-	-	-
64:	8:	-	-	-	-	-	-	-	-
72:	9:	-	-	-	-	-	-	-	-
80:	10:	0	0	0	0	0	-78dBm	0.00mA	0.00mA
88:	11:	1201	1202	1203	1204	1205	1206	1207	1208 B
96:	12:	1209	1210	1211	1212	1213	1214	1215	1216 B
104:	13:	1217	1218	1219	1220	1221	1222	1223	1224 B
112:	14:	-	-	-	-	-	-	-	-
120:	15:	-	-	-	-	-	-	-	-

Here is the **OR** (Output Registers) diagnostics command **at the Outstation #10**.

You can indeed see that the data 1201, 1202, 1203….1224 has been successfully sent to the Outstation **Output** registers starting at address 88, ready to be read out by PLC2.

OUTPUT REGISTERS

If analogues not 0..20mA, scale as necessary

Reg No.	Block No.	0	1	2	3	4	5	6	7
0:	0:	-	-	-	-	-	-	-	-
8:	1:	-	-	-	-	-	-	-	-
16:	2:	-	-	-	-	-	-	-	-
24:	3:	-	-	-	-	-	-	-	-
32:	4:	-	-	-	-	-	-	-	-
40:	5:	-	-	-	-	-	-	-	-
48:	6:	-	-	-	-	-	-	-	-
56:	7:	-	-	-	-	-	-	-	-
64:	8:	-	-	-	-	-	-	-	-
72:	9:	-	-	-	-	-	-	-	-
80:	10:	0	0	0	0	0	-78dBm	0.00mA	0.00mA
88:	11:	1201	1202	1203	1204	1205	1206	1207	1208 B
96:	12:	1209	1210	1211	1212	1213	1214	1215	1216 B
104:	13:	1217	1218	1219	1220	1221	1222	1223	1224 B
112:	14:	-	-	-	-	-	-	-	-
120:	15:	-	-	-	-	-	-	-	-

2.3.3 Basestation (Slave) Modbus Eavesdrop

Running "B" diagnostics command on the Basestation (Slave mode) shows the following:

10:50:29 RX: 01 10 00 58 00 18 30 04 B1 04 B2 04 B3 04 B4 04 B5 04 B6 04 B7 04 B8 04 B9 04 BA 04 BB 04 BC 04 BD 04 BE 04 BF 04 C0 04 C1 04 C2 04 C3 04 C4 04 C5 04 C6 04 C7 04 C8 FE 21 **Preset multiple registers 88..111**

10:50:29 TX: 01 10 00 58 00 18 41 D0

10:50:30 RX: 01 04 00 58 00 18 71 D3 **Read input registers 88..111**

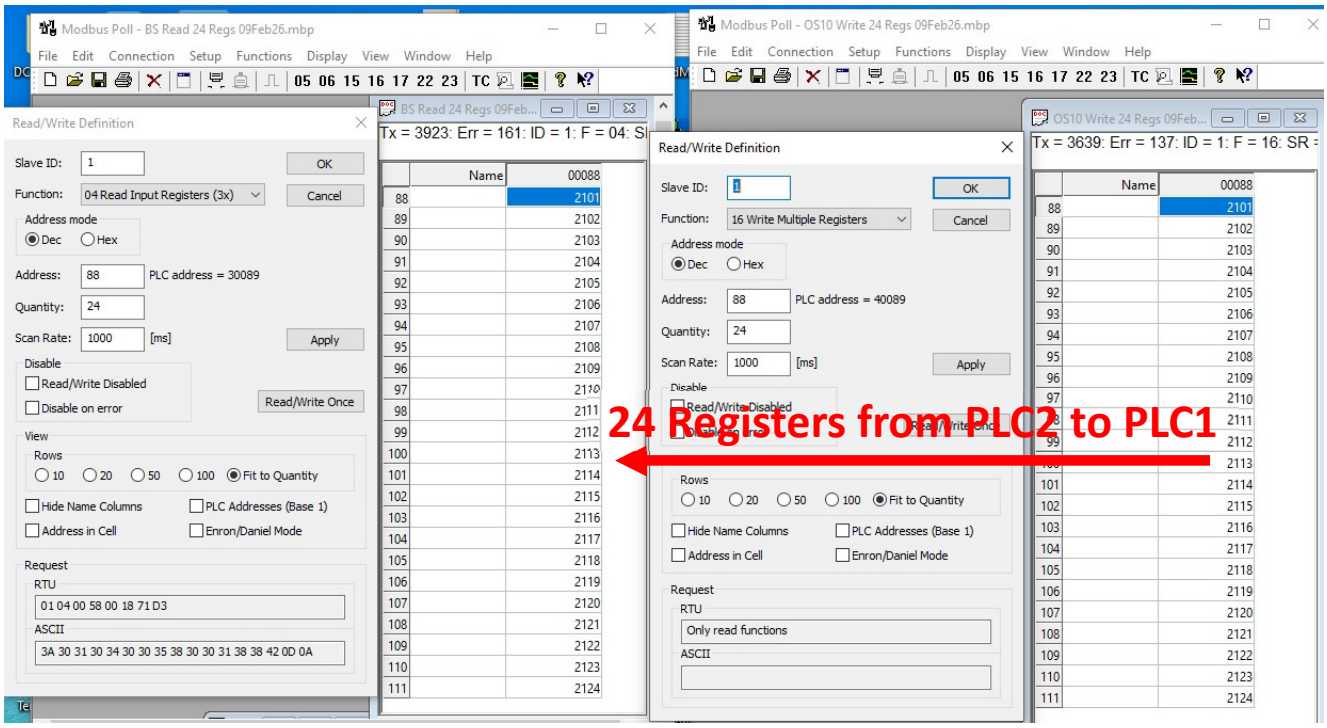
10:50:30 TX: 01 04 30 08 35 08 36 08 37 08 38 08 39 08 3A 08 3B 08 3C 08 3D 08 3E 08 3F 08 40 08 41 08 42 08 43 08 44 08 45 08 46 08 47 08 48 08 49 08 4A 08 4B 08 4C A9 4E

2.4 PLC2 (Master) to PLC1 (Master) Direction

2.4.1 PLC2 to PLC1 Operation

The Modbus Poll on the right, simulating PLC2 is used to write 24 register values into Outstation #10 with destination starting address #88 by means of the "16 Write Multiple Registers" command. In this example the data values to be written are 2101, 2102, 2103……2124.

The Modbus Poll on the left, simulating PLC1 is used to read out 24 register values from the Basestation from starting address #88 by means of the "04 Read **Input** Registers (3x)" command. In the example shown, the data values which are read out and displayed are 2101, 2102, 2103……2124, as expected.



2.4.2 Mega_Link 2 Configuration

There are no further data routing entries required for sending 24 registers in the opposite direction, i.e. PLC2 to PLC1 because this is taken care of by the symmetric nature of polling data in both directions.

2.4.3 Explanation

Here is the **IR** (Input Registers) diagnostics command **at the Outstation #10**.

You can see that the data 2101, 2102, 2103….2124 is being written into the **Input** registers starting at address 88.

INPUT REGISTERS

If analogues not 0..20mA, scale as necessary

Reg No.	Block No.	0	1	2	3	4	5	6	7
0:	0:	-	-	-	-	-	-	-	-
8:	1:	-	-	-	-	-	-	-	-
16:	2:	-	-	-	-	-	-	-	-
24:	3:	-	-	-	-	-	-	-	-
32:	4:	-	-	-	-	-	-	-	-
40:	5:	-	-	-	-	-	-	-	-
48:	6:	-	-	-	-	-	-	-	-
56:	7:	-	-	-	-	-	-	-	-
64:	8:	-	-	-	-	-	-	-	-
72:	9:	-	-	-	-	-	-	-	-
80:	10:	16	8	7	9	11.93V	-79dBm	0.38mA	0.35mA
88:	11:	2101	2102	2103	2104	2105	2106	2107	2108 B
96:	12:	2109	2110	2111	2112	2113	2114	2115	2116 B
104:	13:	2117	2118	2119	2120	2121	2122	2123	2124 B
112:	14:	-	-	-	-	-	-	-	-
120:	15:	-	-	-	-	-	-	-	-

Here is the **IR** (Input Registers) diagnostics command **at the Basestation** (the database of the system states).

You can see that the data 2101, 2102, 2103….2124 has successfully been transferred over to the database of **Input** registers starting at address 88, ready to be read out by PLC1 using the read input registers command.

INPUT REGISTERS

If analogues not 0..20mA, scale as necessary

Reg No.	Block No.	0	1	2	3	4	5	6	7
0:	0:	37	38	35	36	12.03V	-75dBm	9.29mA	18.43mA
8:	1:	-	-	-	-	-	-	-	-
16:	2:	-	-	-	-	-	-	-	-
24:	3:	-	-	-	-	-	-	-	-
32:	4:	-	-	-	-	-	-	-	-
40:	5:	-	-	-	-	-	-	-	-
48:	6:	-	-	-	-	-	-	-	-
56:	7:	-	-	-	-	-	-	-	-
64:	8:	-	-	-	-	-	-	-	-
72:	9:	-	-	-	-	-	-	-	-
80:	10:	16	8	7	9	11.99V	-69dBm	0.38mA	0.34mA
88:	11:	2101	2102	2103	2104	2105	2106	2107	2108 B
96:	12:	2109	2110	2111	2112	2113	2114	2115	2116 B
104:	13:	2117	2118	2119	2120	2121	2122	2123	2124 B
112:	14:	-	-	-	-	-	-	-	-
120:	15:	-	-	-	-	-	-	-	-

2.4.4 Outstation 10 (Slave) Modbus Eavesdrop

Running “B” diagnostics command on the Outstation #10 (Slave mode) shows the following:

10:52:14 RX: 01 03 00 58 00 18 C4 13 **Read holding registers 88..111**

10:52:14 TX: 01 03 30 04 B1 04 B2 04 B3 04 B4 04 B5 04 B6 04 B7 04 B8 04 B9 04 BA 04 BB 04 BC 04 BD 04 BE 04 BF 04 C0 04 C1 04 C2 04 C3 04 C4 04 C5 04 C6 04 C7 04 C8 D8 F7

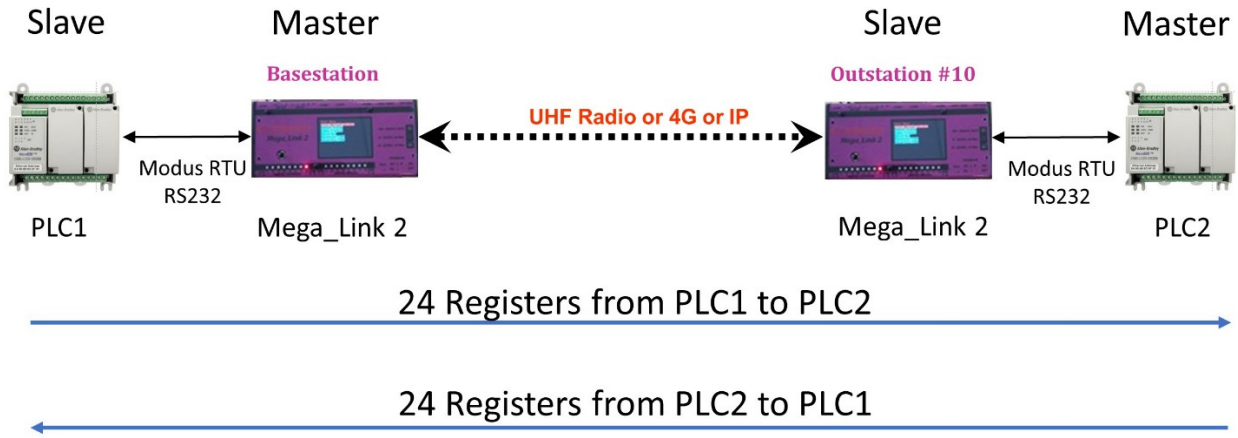
10:52:16 RX: 01 10 00 58 00 18 30 08 35 08 36 08 37 08 38 08 39 08 3A 08 3B 08 3C 08 3D 08 3E 08 3F 08 40 08 41 08 42 08 43 08 44 08 45 08 46 08 47 08 48 08 49 08 4A 08 4B 08 4C 0D 09 **Preset multiple registers 88..111**

10:52:16 TX: 01 10 00 58 00 18 41 D0

3. Register Transfers Between PLC1 (Slave) and PLC2 (Master)

3.1 Overview

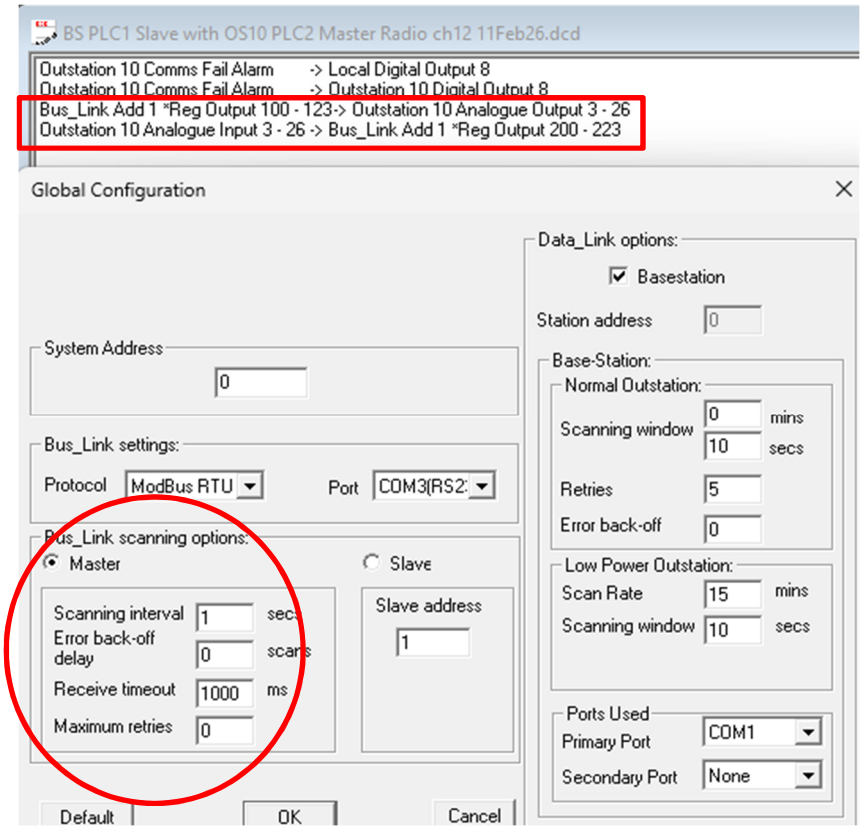
This example shows how to transfer 24 Registers from PLC1 to PLC2 and 24 Registers from PLC2 to PLC1 when PLC1 is set up as a Slave and PLC2 is set as a Master and the Basestation is working in Master mode and Outstation #10 is working as a slave like before.



3.2 Mega_Link 2 Configuration Files

3.2.1 Basestation (Master mode) Configuration

The Basestation is configured as a Master using COM3 (RS232).



Two lines of data routing are required.

Line 3

This is the Basestation master command to read 24 registers at address 100-123 from PLC1 slave address=1 and send to 24 registers at Outstation #10 Analogue Outputs 3-26.

Outstation #10 Analogue Output number 3 starts at Mega_Link 2 address 88 and is the start of virtual Outstation #11.

Line 4

This is the Basestation master command to take 24 registers at Outstation #10 Analogue Inputs 3-26 and write to 24 registers at address 200-223 of PLC1 slave address=1.

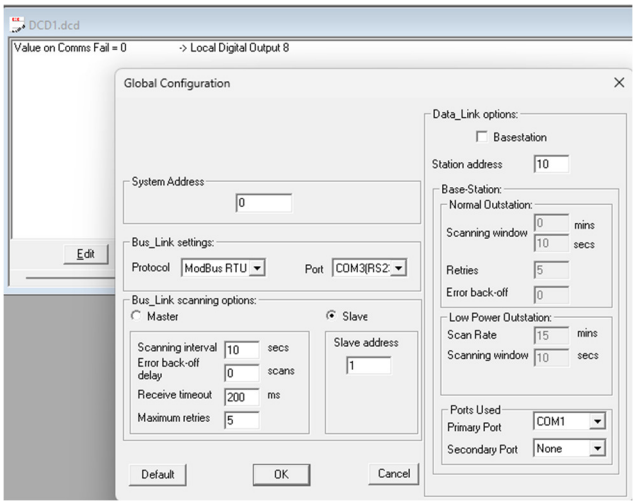
In the example shown, the Basestation is set to Modbus RTU Master mode via COM3A RS232.

The Master mode scanning rate is set at 1 second with 1 second timeout and no retries.

3.2.2 Outstation #10 (Slave Mode) Config

The Outstation #10 is configured as a Slave using COM3 (RS232) with Slave address = 1.

In this example the Outstation data routing is simply the basic Comms Fail output value on comms fail action.



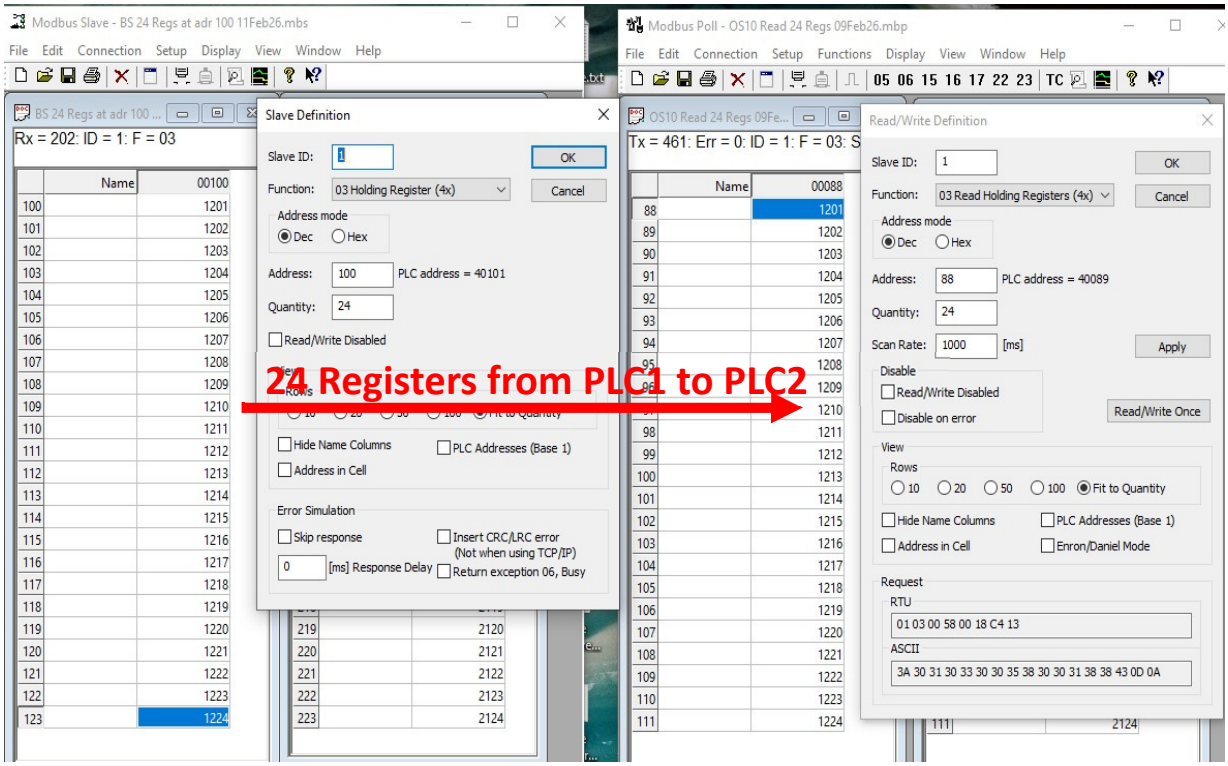
3.3 PLC1 (Slave) to PLC2 (Master) Direction

3.3.1 PLC1 to PLC2 Operation

In this example Modbus Slave is used to simulate the PLC1 (Slave) at the Basestation on the left and Modbus Poll is used to simulate PLC2 (Master) at the Outstation #10, on the right of screen grab.

The Modbus Slave on the left, simulating PLC1 is used to allow the Basestation to read 24 register values from it starting at address 100 to be sent to destination starting address #88. In this example the data values to be written are 1201, 1202, 1203……1224.

The Modbus Poll on the right, simulating PLC2 Master is used to read out 24 register values from the Outstation 10 from starting address #88 by means of the "03 Read Holding Registers (4x)" command. In this example shown the data values which are read out are 1201, 1202, 1203……1224, as expected.



3.3.2 Basestation (Master) Modbus Eavesdrop

Running "B" diagnostics command on the Basestation (Master mode) shows the following:

09:04:12 TX: 01 10 00 C8 00 18 30 08 35 08 36 08 37 08 38 08 39 08 3A 08 3B 08 3C 08 3D 08 3E 08 3F 08 40 08 41 08 42 08 43 08 44 08 45 08 46 08 47 08 48 08 49 08 4A 08 4B 08 4C 3E 17 **Presetting Multiple registers 200..223 of Modbus address 1**

09:04:13 RX: 01 10 00 C8 00 18 41 FD

09:04:13 TX: 01 03 00 64 00 18 04 1F **Read holding register 100..123 of Modbus address 1**

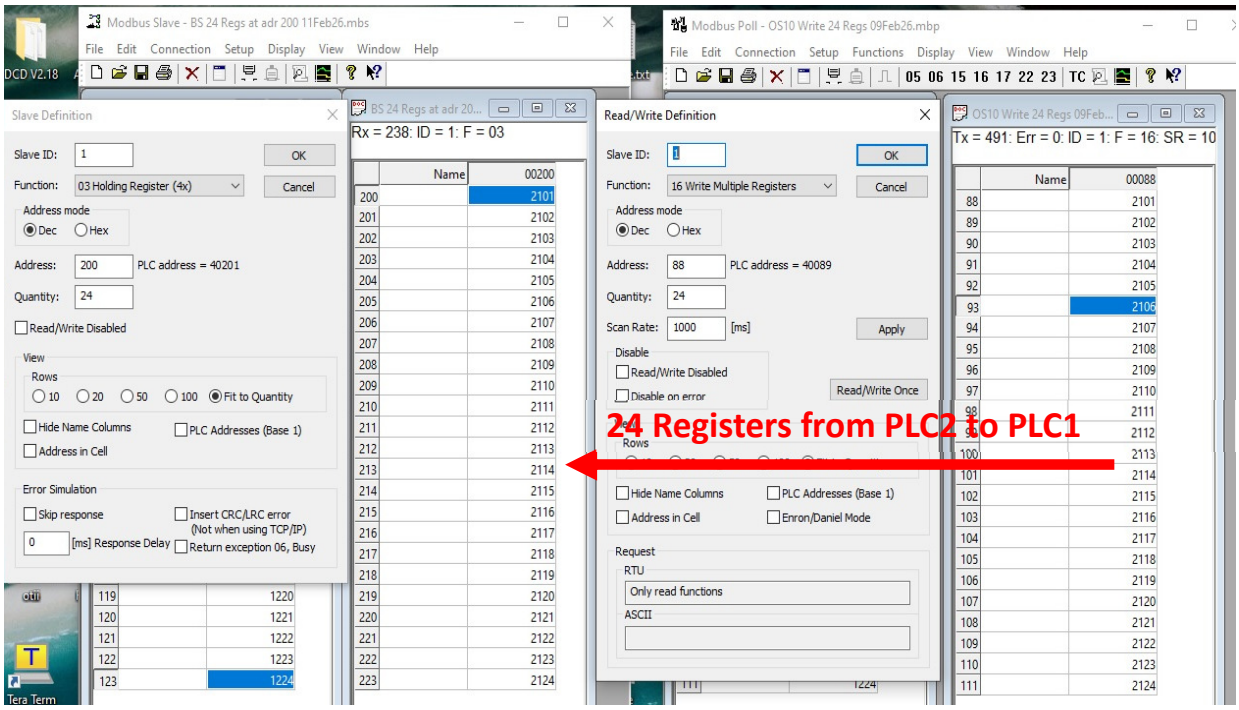
09:04:13 RX: 01 03 30 04 B1 04 B2 04 B3 04 B4 04 B5 04 B6 04 B7 04 B8 04 B9 04 BA 04 BB 04 BC 04 BD 04 BE 04 BF 04 C0 04 C1 04 C2 04 C3 04 C4 04 C5 04 C6 04 C7 04 C8 D8 F7

3.4 PLC2 (Master) to PLC1 (Slave) Direction

3.4.1 PLC2 to PLC1 Operation

The Modbus Poll on the right, simulating PLC2 is used to write 24 register values into the Outstation #10 with destination starting address #88 by means of the "16 Write Multiple Registers" command. In this example the data values to be written are 2101, 2102, 2103……2124.

The Modbus Slave on the right, simulating PLC1 is used to allow the Basestation master to write 24 register values to starting address 100. In this example shown the data values which are read out are 2101, 2102, 2103……2124, as expected.



3.4.2 Outstation 10 (Slave) Modbus Eavesdrop

Running "B" diagnostics command on the Outstation 10 (Slave mode) shows the following:

09:10:42 RX: 01 03 00 58 00 18 C4 13 **Read holding registers 88..111**

09:10:42 TX: 01 03 30 04 B1 04 B2 04 B3 04 B4 04 B5 04 B6 04 B7 04 B8 04 B9 04 BA 04 BB 04 BC 04 BD 04 BE 04 BF 04 C0 04 C1 04 C2 04 C3 04 C4 04 C5 04 C6 04 C7 04 C8 D8 F7

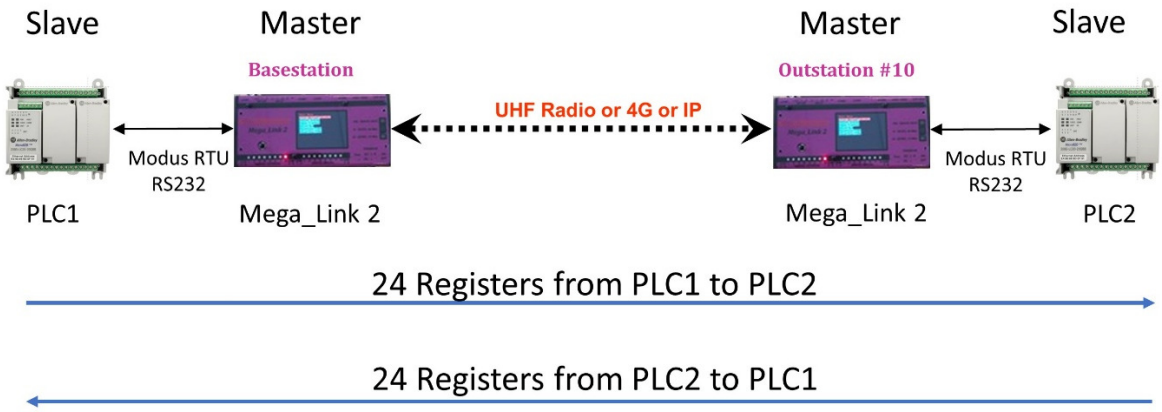
09:10:43 RX: 01 10 00 58 00 18 30 08 35 08 36 08 37 08 38 08 39 08 3A 08 3B 08 3C 08 3D 08 3E 08 3F 08 40 08 41 08 42 08 43 08 44 08 45 08 46 08 47 08 48 08 49 08 4A 08 4B 08 4C 0D 09 **Preset multiple registers 88..111**

09:10:43 TX: 01 10 00 58 00 18 41 D0

4. Register Transfers Between PLC1 (Slave) and PLC2 (Slave)

4.1 Overview

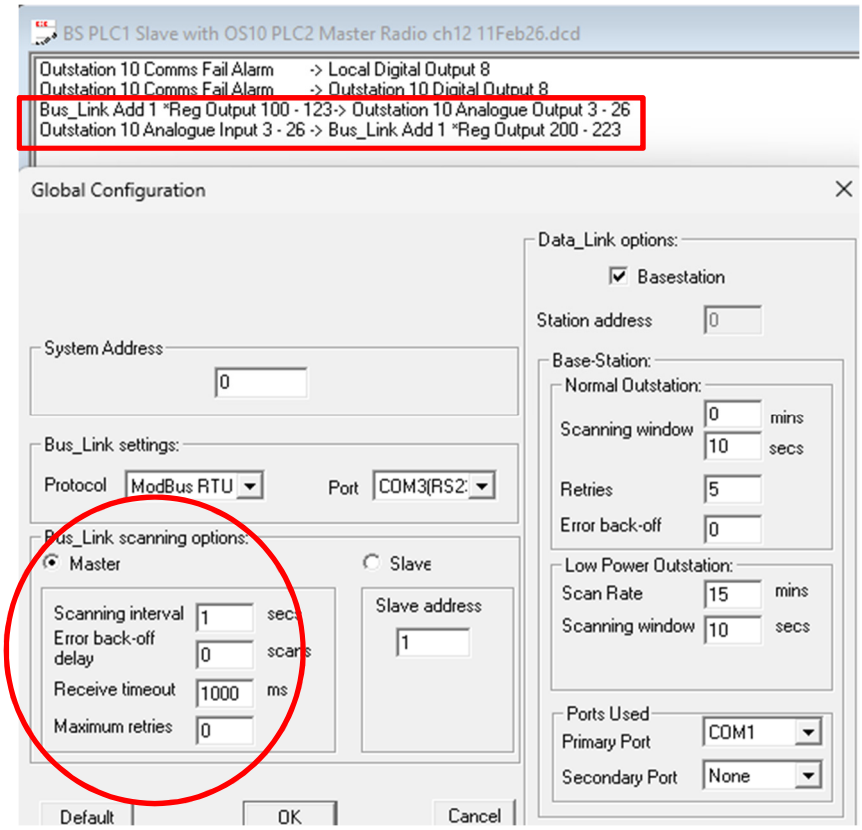
This example shows how to transfer 24 Registers from PLC1 to PLC2 and 24 Registers from PLC2 to PLC1 when PLC1 is set up as a Slave and PLC2 is set as a Slave and the Basestation is working in Master mode and Outstation #10 is working is also working in Master Mode.



4.2 Mega_Link 2 Configuration Files

4.2.1 Basestation (Master mode) Configuration

The Basestation is configured as a Master using COM3 (RS232).



Two lines of data routing are required.

Line 3

This is the Basestation master command to read 24 registers at address 100-123 from PLC1 slave address=1 and send to 24 registers at Outstation #10 Analogue Outputs 3-26.

Outstation #10 Analogue Output number 3 starts at Mega_Link 2 address 88 and is the start of virtual Outstation #11.

Line 4

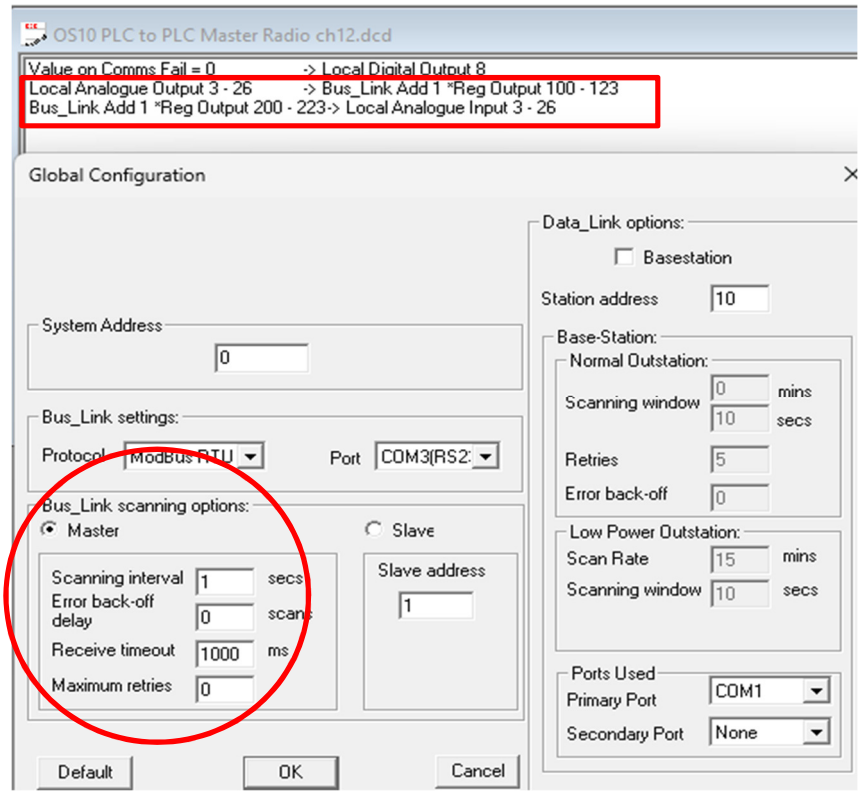
This is the Basestation master command to take 24 registers at Outstation #10 Analogue Inputs 3-26 and write to 24 registers at address 200-223 of PLC1 slave address=1.

In the example shown, the Basestation is set to Modbus RTU Master mode via COM3A RS232.

The Master mode scanning rate is set at 1 second with 1 second timeout and no retries.

4.2.2 Outstation #10 (Master Mode) Configuration

The Outstation #10 is configured in Master mode using COM3 (RS232).



Two extra lines of data routing are required.

Line 2

This is the Outstation #10 master command to take 24 registers at Outstation #10 Analogue Outputs 3-26 and write them to 24 registers at address 100-123 of PLC2 slave address=1.

Line 3

This is the Outstation #10 master command to read out 24 registers at address 200-223 from PLC2 slave address=1 and store as 24 registers in Outstation #10 (Local) Analogue Inputs 3-26.

Outstation #10 Analogue Input number 3 starts at Mega_Link 2 address 88 and is the start of virtual Outstation #11.

In the example shown, the Basestation is set to Modbus RTU Master mode via COM3A RS232.

The Master mode scanning rate is set at 1 second with 1 second timeout and no retries.

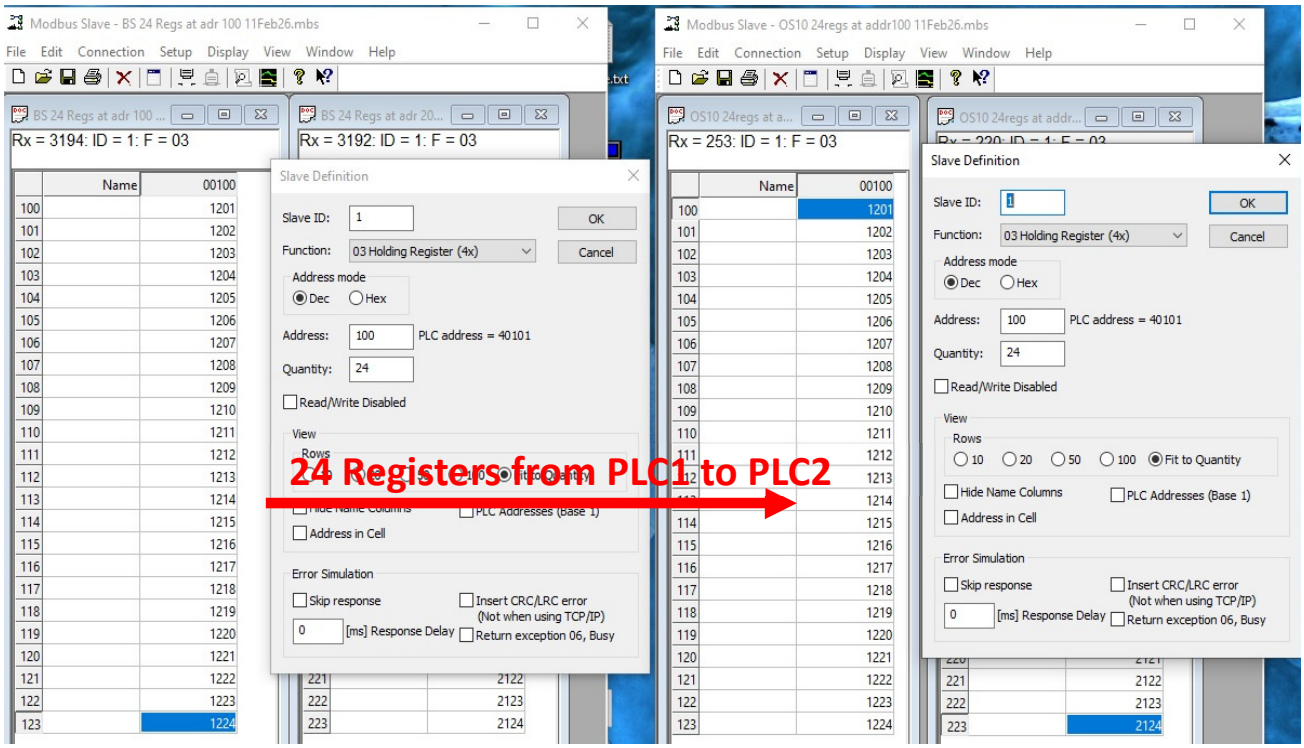
4.3 PLC1 (Slave) to PLC2 (Slave) Direction

4.3.1 PLC1 to PLC2 Operation

In this example Modbus Slave is used to simulate the PLC1 (Slave) at the Basestation on the left and another instance of Modbus Slave is used to simulate PLC2 (Slave) at the Outstation #10, on the right of screen grab.

The Modbus Slave on the left, simulating PLC1 is used to allow the Basestation master to read 24 register values from it starting at address 100 to be sent to destination starting address #88. In this example the data values to be written are 1201, 1202, 1203……1224.

The Modbus Slave on the right, simulating PLC2 is used to allow the Outstation #10 master to write 24 register values to starting address 100 of slave address=1. In this example shown the data values which are read out are 1201, 1202, 1203……1224, as expected.



4.3.2 Basestation (Master) Modbus Eavesdrop

Running "B" diagnostics command on the Basestation (Master mode) shows the following:

09:04:12 TX: 01 10 00 C8 00 18 30 08 35 08 36 08 37 08 38 08 39 08 3A 08 3B 08 3C 08 3D 08 3E 08 3F 08 40 08 41 08 42 08 43 08 44 08 45 08 46 08 47 08 48 08 49 08 4A 08 4B 08 4C 3E 17 **Presetting Multiple registers 200..223 of Modbus address 1**

09:04:13 RX: 01 10 00 C8 00 18 41 FD

09:04:13 TX: 01 03 00 64 00 18 04 1F **Read holding register 100..123 of Modbus address 1**

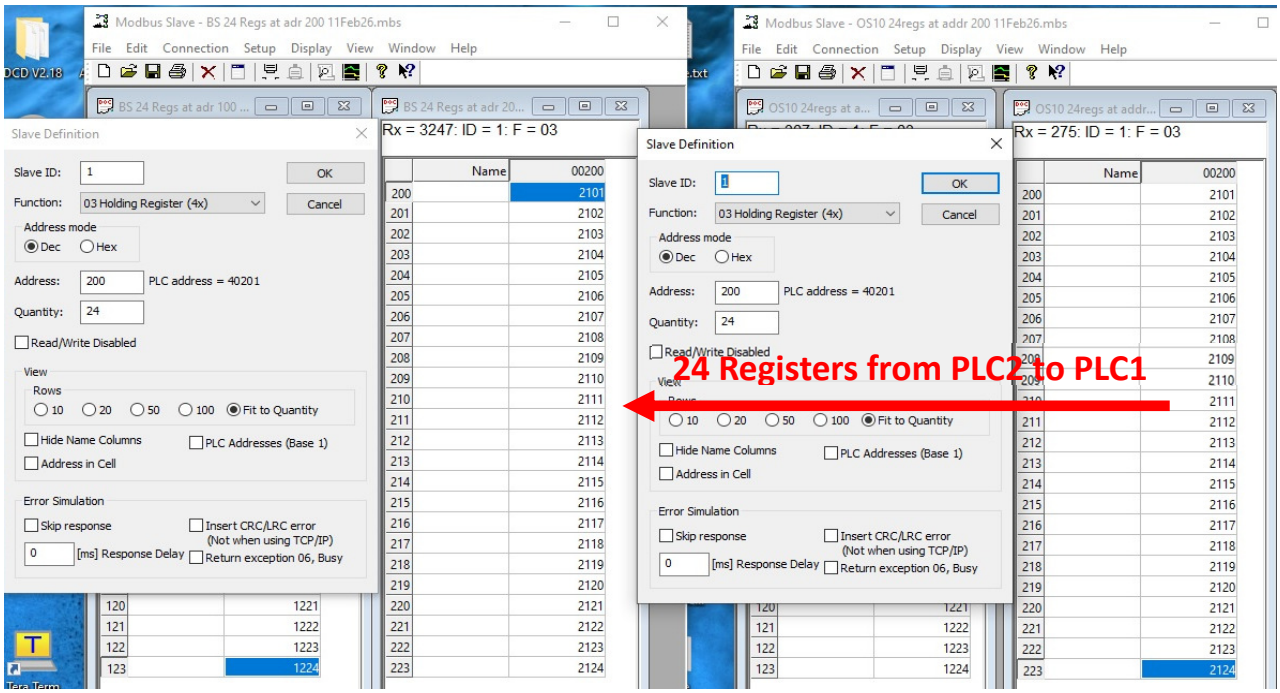
09:04:13 RX: 01 03 30 04 B1 04 B2 04 B3 04 B4 04 B5 04 B6 04 B7 04 B8 04 B9 04 BA 04 BB 04 BC 04 BD 04 BE 04 BF 04 C0 04 C1 04 C2 04 C3 04 C4 04 C5 04 C6 04 C7 04 C8 D8 F7

4.4 PLC2 (Slave) to PLC1 (Slave) Direction

4.4.1 PLC2 to PLC1 Operation

The Modbus Slave on the left, simulating PLC1 is used to allow the Basestation master to read 24 register values from it starting at address 200. In this example the data values to be written are 2101, 2102, 2103……2124.

The Modbus Slave on the right, simulating PLC2 is used to allow the Basestation master to write 24 register values to it starting at address 200. In this example shown the data values which are displayed are 2101, 2102, 2103……2124, as expected.



4.4.2 Outstation 10 (Master) Modbus Eavesdrop

Running “B” diagnostics command on the Outstation 10 (Master mode) shows the following:

10:07:42 TX: 01 10 00 64 00 18 30 04 B1 04 B2 04 B3 04 B4 04 B5 04 B6 11 5C 04 B8 04 B9 04 BA 04 BB 04 BC 04 BD 04 BE 04 BF 04 C0 04 C1 04 C2 04 C3 04 C4 04 C5 04 C6 04 C7 04 C8 7D 8D **Presetting Multiple registers 100..123 of Modbus address 1**

10:07:42 RX: 01 10 00 64 00 18 81 DC

10:07:42 TX: 01 03 00 C8 00 18 C4 3E **Read holding register 200..223 of Modbus address 1**

10:07:43 RX: 01 03 30 08 35 08 36 08 37 08 38 08 39 08 3A 08 3B 08 3C 08 3D 08 3E 08 3F 08 40 08 41 08 42 27 0F 08 44 08 45 08 46 08 47 08 48 08 49 08 4A 08 4B 08 4C FE D8

End.