Application Note

XL4*Plus* as a Modbus Slave

Introduction

The XL4plus can have many communication configurations. This application note covers a scenario where the XL4plus is configured as a Modbus Server (Slave) and communicates with a Modbus Client (Master) using Modbus TCP.

The Modbus Master simulator that is used is QModMaster 0.4.5

All configuration is done with Workbench.

Ethernet Port Configuration

The XL4plus can use DHCP or have a static IP assigned or use NetBEUI. The NetBEUI default name of the RTU is "xl4-serial number", where serial number is the unique RTU serial number. Optionally this can be changed by entering a name in the Configuration → Name field.
In this scenario we are using a fixed IP address of 192.168.3.97

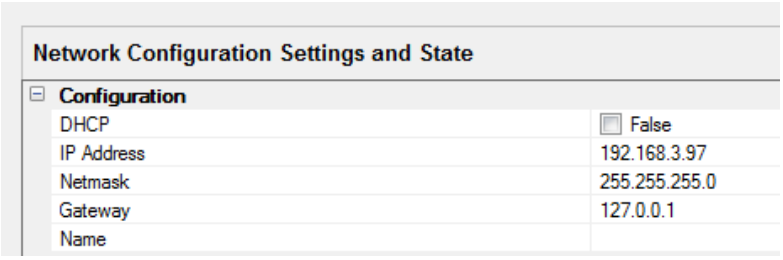


Figure 1. Ethernet configuration

Modbus Slave Configuration

Check the "Enabled" box, set the Modbus Slave ID ("1" in this example). The default TCP port is usually 502, do not alter unless you have a good reason to.
The data offset settings map the slave IO into the RTU IO space. Generally you would at least offset from the native RTU physical IO (8 DI, 8 DO, 6AI, 2AO) as shown below.

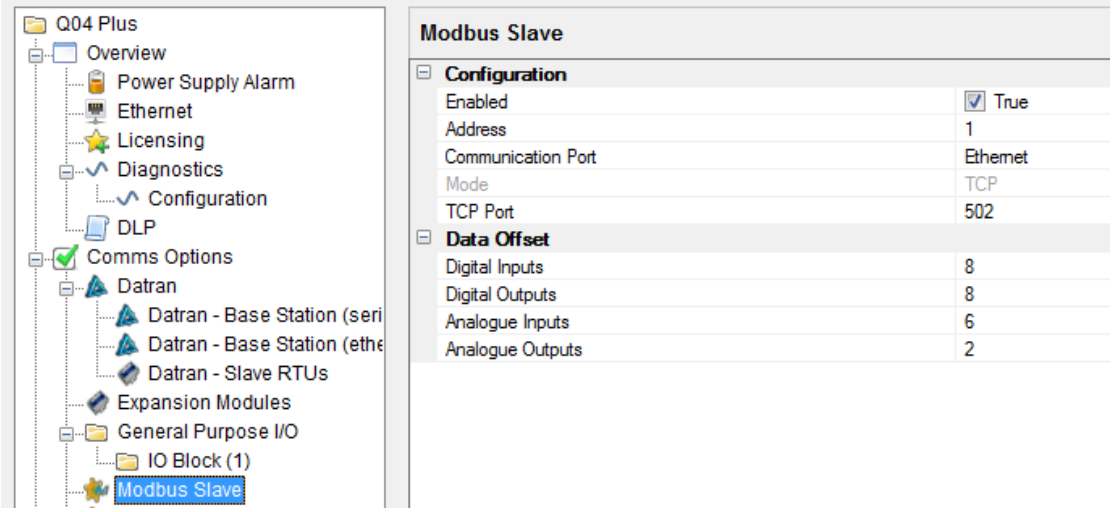


Figure 2.

QTech Data Systems Ltd
12 Midas Place
Middleton
Christchurch 8024
New Zealand

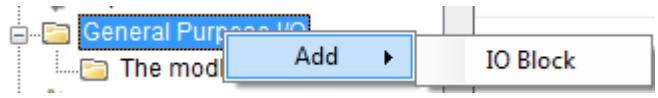
Phone: +64 3 366 3713
Fax: +64 3 365 2815
E-mail: sales@qtech.co.nz
Web: www.qtech.co.nz

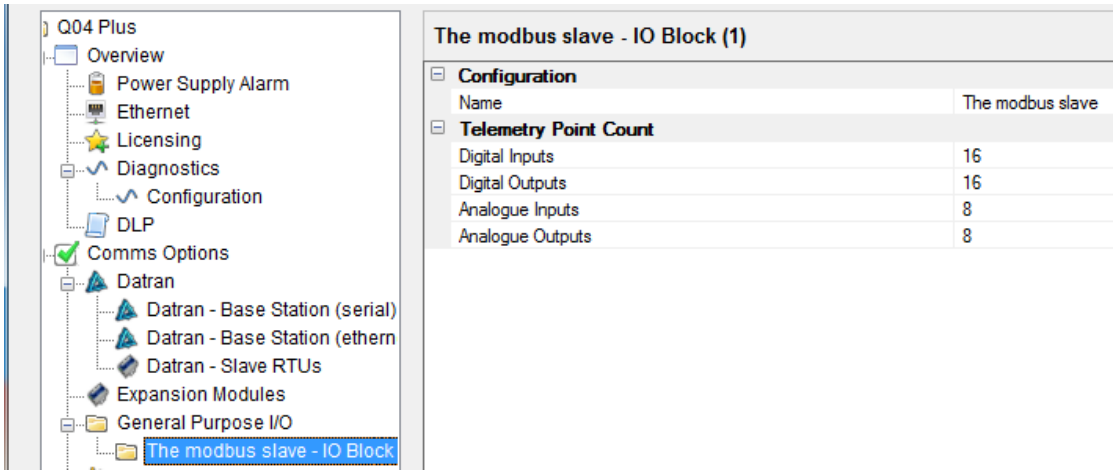
 Version 1

Application Note

Memory Allocation

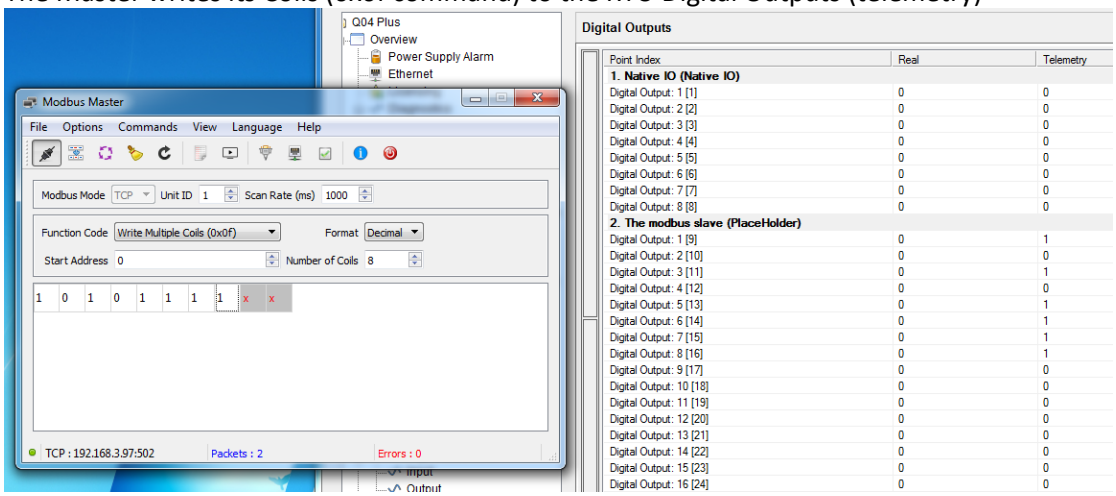
Memory needs to be allocated in the RTU for the Modbus Master to read/write into. This is done by allocating a General Purpose I/O block. Right click on “General Purpose I/O” and add a IO Block. Then specify the I/O allocation you want to be available for the Modbus Master





Modbus Master(simulator)

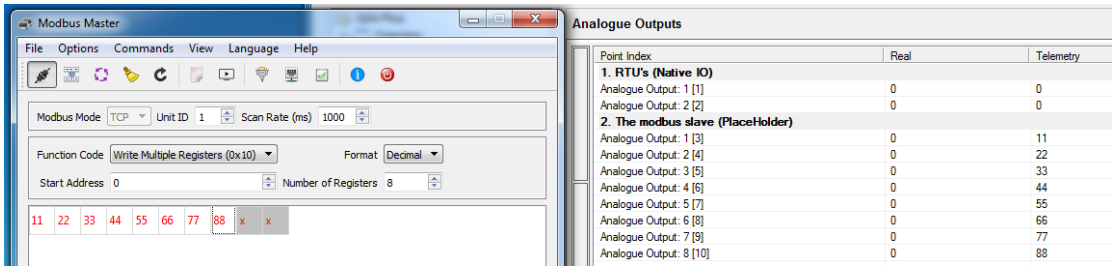
The Master writes its Coils (0x0f command) to the RTU Digital Outputs (telemetry)



QTech Data Systems Ltd
12 Midas Place
Middleton
Christchurch 8024
New Zealand

Phone: +64 3 366 3713
Fax: +64 3 365 2815
E-mail: sales@qtech.co.nz
Web: www.qtech.co.nz

The Modbus Master writes its Registers (0x10 command) to the RTU analogue outputs (telemetry)



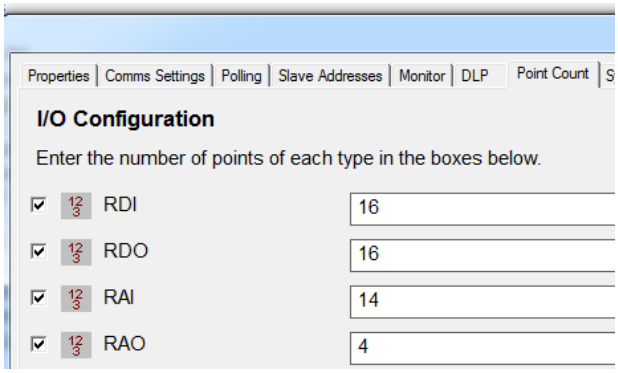
Note that only the Telemetry values have been updated at the RTU. This is because there is a DLP loaded with the telinp/telout commands that break the TAOUT to RAOUT automatic copying.

```
proginit

telinp
telout

progstart

progenc
```

Modbus Master (PLC)		**Modbus Slave (XL4plus)**
Write Multiple Registers (0x10)		
Write Register (0x06)	-->	Telemetry - Analogue Output
Read Holding Register (0x03)	<--	
Write Multiple Coils (0x0F)	-->	Telemetry - Digital Output
Read Coils (0x01)	<--	
Read Input Register (0x04)	<--	Telemetry - Analogue Input
Read Discrete Input (0x02)	<--	Telemetry Digital Input

Datran IO

Note that Datran will overwrite the telemetry values if they are in the Datran point count in the site comms object. For example. The site has declared 4 RAO (physically for an XL4plus there can only be 2

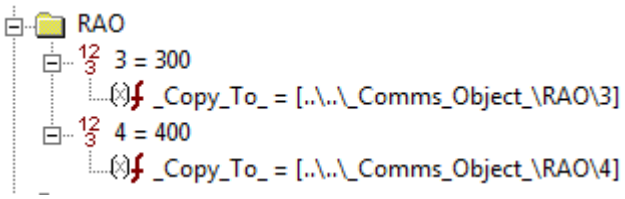


QTech Data Systems Ltd
12 Midas Place
Middleton
Christchurch 8024
New Zealand

Phone: +64 3 366 3713
Fax: +64 3 365 2815
E-mail: sales@qtech.co.nz
Web: www.qtech.co.nz

Application Note

Datran Comms Object Point Count configuration



Datran writing to RAOUT3 & RAOUT4

Analogue Outputs

Point Index	Real	Telemetry
1. RTU's (Native IO)		
Analogue Output: 1 [1]	0	0
Analogue Output: 2 [2]	0	0
2. The modbus slave (PlaceHolder)		
Analogue Output: 1 [3]	0	300
Analogue Output: 2 [4]	0	400
Analogue Output: 3 [5]	0	33
Analogue Output: 4 [6]	0	44
Analogue Output: 5 [7]	0	55
Analogue Output: 6 [8]	0	66
Analogue Output: 7 [9]	0	77
Analogue Output: 8 [10]	0	88

Values from Datran appear at RTU. However, these will be over written if the Modbus Master does a (0x10) write command that includes 40001 and 40002

To enable Datran to read the holding register values they will need to be copied from the telemetry analogue outputs to a telemetry analogue input that Datran can access. This can be a notional analogue input or you can use the General Purpose I/O.

Using General Purpose I/O to make Holding Registers available to Datran

Allocate the I/O. Right click on General Purpose I/O → Add → IO Block
Enter appropriate values for the required IO. For example

IO for Datran - IO Block (2)

Configuration	
Name	IO for Datran
Telemetry Point Count	
Digital Inputs	0
Digital Outputs	0
Analogue Inputs	8
Analogue Outputs	0

In the DLP copy the Holding Register data (TAOUT) to the newly allocated analogue Inputs

QTech Data Systems Ltd
12 Midas Place
Middleton
Christchurch 8024
New Zealand

Phone: +64 3 366 3713
Fax: +64 3 365 2815
E-mail: sales@qtech.co.nz
Web: www.qtech.co.nz

 Version 1

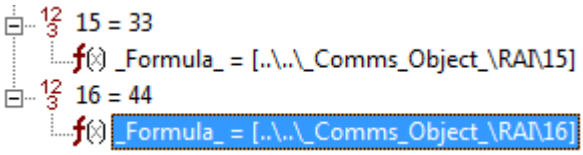
```
1  proginit
2
3  teling
4  telout
5
6  progstart
7     cpanl taout5, tain15,6
8  progenc
```

Here we are leaving TAOUT3 and 4 as data from Datran to the Modbus Master holding registers (40001 – 40002). And copying TAOUT5 – TAOUT10 (40003 – 40008) to TAIN15 – TAIN20. Data from the Modbus Master to Datran

Analogue Inputs

Point Index	Real	Telemetry
1. RTU's (Native IO)		
Analogue Input: 1 [1]	0	0
Analogue Input: 2 [2]	0	0
Analogue Input: 3 [3]	0	0
Analogue Input: 4 [4]	0	0
Analogue Input: 5 [5]	0	0
Analogue Input: 6 [6]	0	0
2. The modbus slave (PlaceHolder)		
Analogue Input: 1 [7]	0	0
Analogue Input: 2 [8]	0	0
Analogue Input: 3 [9]	0	0
Analogue Input: 4 [10]	0	0
Analogue Input: 5 [11]	0	0
Analogue Input: 6 [12]	0	0
Analogue Input: 7 [13]	0	0
Analogue Input: 8 [14]	0	0
3. IO for Datran (PlaceHolder)		
Analogue Input: 1 [15]	0	33
Analogue Input: 2 [16]	0	44
Analogue Input: 3 [17]	0	55
Analogue Input: 4 [18]	0	66
Analogue Input: 5 [19]	0	77
Analogue Input: 6 [20]	0	88
Analogue Input: 7 [21]	0	0
Analogue Input: 8 [22]	0	0

Data copied by DLP to the General Purpose I/O block



Data as retrieved by Datran

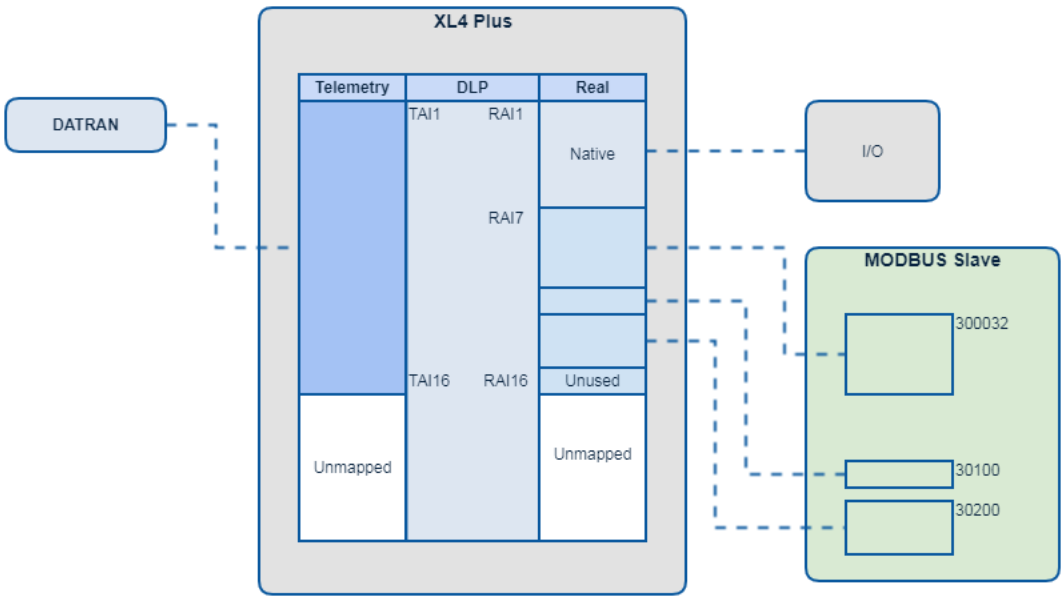
QTech Data Systems Ltd
12 Midas Place
Middleton
Christchurch 8024
New Zealand

Phone: +64 3 366 3713
Fax: +64 3 365 2815
E-mail: sales@qtech.co.nz
Web: www.qtech.co.nz

Appendix: Background Information - RTU as an IPB Master

RTU as a MODBUS Master

The MODBUS slave has a set of data values that are organised as MODBUS Coils, Discrete Inputs, Holding Registers, and Input Registers. Often we aren't interested in all of the slaves MODBUS IO, but just a few specific values that aren't necessarily organised in a contiguous block in the slaves MODBUS data table.



The process is to allocate a *MODBUS Range* that will create a slot in the RTU data table for the MODBUS values to be stored. In conjunction with the range we specify a list of *actions* that define the MODBUS communications messages that will gather the data from the slave. These separate pieces of data map as a contiguous block in the RTUs data table. It is good practice to allocate more data in the MODBUS range than required so that additional actions can be added later.

RTU with Multiple Masters

Consider a scenario where we want the RTU to poll for data from some local MODBUS devices; e.g. a flow meter and pH transducer. The RTU will be polled by the DATRAN base station for the RTUs native I/O and the MODBUS device I/O. There is also a PLC at site which has a block of data values that we want to be passed back to the base station, and some set points that we want to pass from the base station to the PLC via the RTU.

The RTU must have the following three comms tasks enabled

- **QComms slave** for communicating with the DATRAN Base Station (a QCOMMS master)
- **MODBUS master** for communicating with the transducers (MOBDBUS slaves)
- **MODBUS slave** for communicating with the PLC (a MODBUS master)

In this scenario there are two separate devices which are both acting as a master to the RTU; DATRAN and the PLC. When there is more than one master there is potential for I/O contention, because there is not necessarily any coordination between the two masters about who is in control of the RTU outputs. For example, the PLC may write to a particular COIL, but DATRAN wouldn't know that that had happened, and subsequently overwrite the TDO with a value that it thinks should be there.

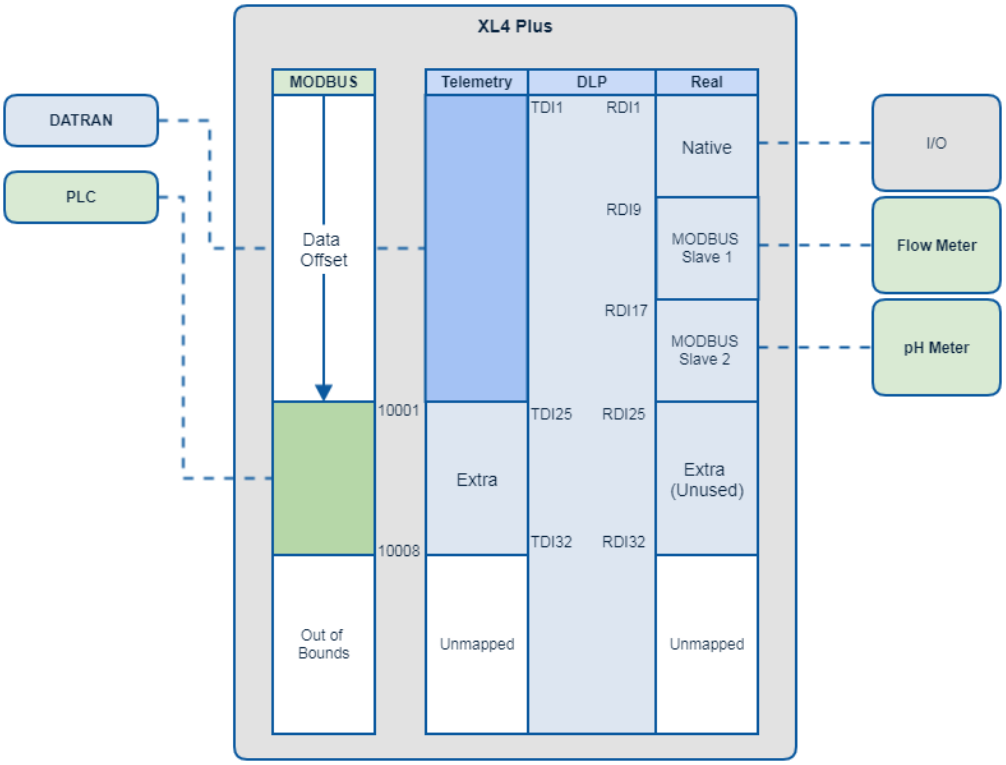
To resolve this issue the XL4 Plus has a configurable *Data Offset* in the MODBUS slave settings. This value effectively remaps the MODBUS I/O to a different position in the Telemetry Data Table. Normally the Data Offset value will be equal to or greater than the sites *Point Count* value configured in the DATRAN Base Station.

The diagram below illustrates this concept for the Digital Inputs, but there would be similar diagram for each type in the RTU Data Table: Digital Inputs, Digital Outputs, Analogue Inputs, and Analogue Outputs.

QTech Data Systems Ltd
12 Midas Place
Middleton
Christchurch 8024
New Zealand

Phone: +64 3 366 3713
Fax: +64 3 365 2815
E-mail: sales@qtech.co.nz
Web: www.qtech.co.nz

Notional Data is not mapped to third-party protocols, so doesn't normally need to be remapped (but see note on the *Master Notional Point Count*).



DATRAN is configured with a Digital Input Point Count of 24, so that it can access TDI1...TDI24 only. The PLC can read the Discrete Inputs 10001…10008, which are mapped into the data table as TDI25…TDI32. If the PLC attempts to read Discrete Inputs 10009 or greater the RTU will return a MODBUS I/O exception code.
Observe how the PLC data maps to a slot that we allocated using some *extra I/O*. For actual data values to be present in that slot we need to include some DLP commands copy values into it. For example, if we wanted the PLC to be able to read the following…

- RDI7 and RDI8 from the RTU
- RDI10 from the Flow meter
- RDI17 from the pH meter

We would include the following lines in the DLP:

```
CPDIG RDIN7,  TDIN25, 2
CPDIG RDIN10, TDIN27, 1
CPDIG RDIN17, TDIN28, 1
```

In fact, a much simpler approach would have been to set the Data Offset, the DATRAN Site Point Count, and the Extra I/O slot to all have the same value. That way, the data layout in DATRAN would exactly correspond to that in the PLC, and the DLP would only need the following line:

```
CPDIG RDIN1, TDIN25, 24
```

Note that these examples only relate to Digital Input values. The DLP would also need to include commands for copying any other data types that were used.

QTech Data Systems Ltd
12 Midas Place
Middleton
Christchurch 8024
New Zealand

Phone: +64 3 366 3713
Fax: +64 3 365 2815
E-mail: sales@qtech.co.nz
Web: www.qtech.co.nz

