DATRAN XL4 Plus

Application Note

DATRAN XL4 Plus

Floating-point number support

April 2021

Rev 1





Introduction

This note gives some examples of how to use the floating-point number conversion functionality available in the DATRAN XL4 *Plus* RTU.

In the examples the RTU will use the MODBUS protocol to read a set of Input Registers from a Swan AMI Turbiwell Turbidity meter. The values read from the meter are converted from floating-point format into integers by the RTU, and then communicated to DATRAN VI.

Product	Version Required
XL4 *Plus* Firmware	>= 7.16
DLP IDE Software	>= 4.04

Preliminaries

The manual for the AMI Turbiwell device outlines how the measurement values are arranged in the MODBUS Input Register memory range as shown in the figure below. The manual specifies that the floating-point values are represented in bid-endian format, and that each value occupies 4 bytes (ie 2 x 16 bit integers).

30.2.2 Read Input Registers (Process data): Function 04

The process data are made up of the measuring values of the AMI Turbidity family.

Tab. 30-1 *Input Registers*

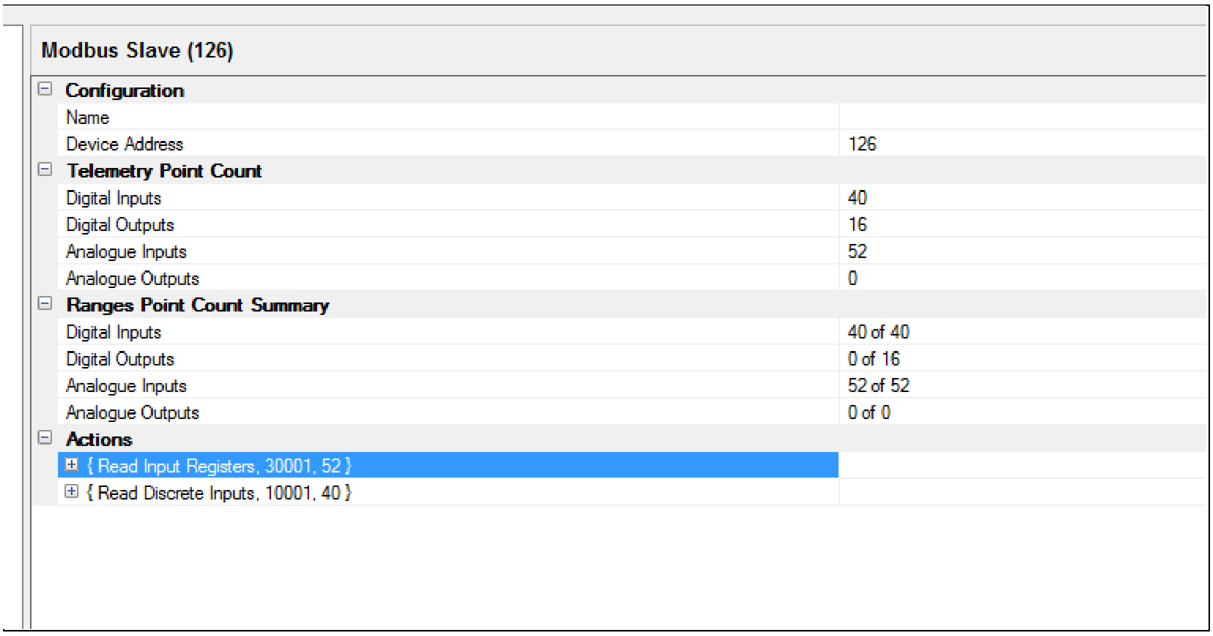
Byte Offset	Data type	Description	Range of physical value	
0	Float	Turbitity	0–300	FNU
4	Float	Temperature (future use fix 25°C)	-25 to +250	°C
8	Float	Flow	0–100	l/h
12	Float	Turbitity raw	0–250'000	Hz
16	Float	Pt 1000 (future use)	500–2400	Ω
20	Float	Raw Flow	0–100	Hz

The Turbidity value has a range of 0 to 300, but no mention is made about how much precision the measurement has.

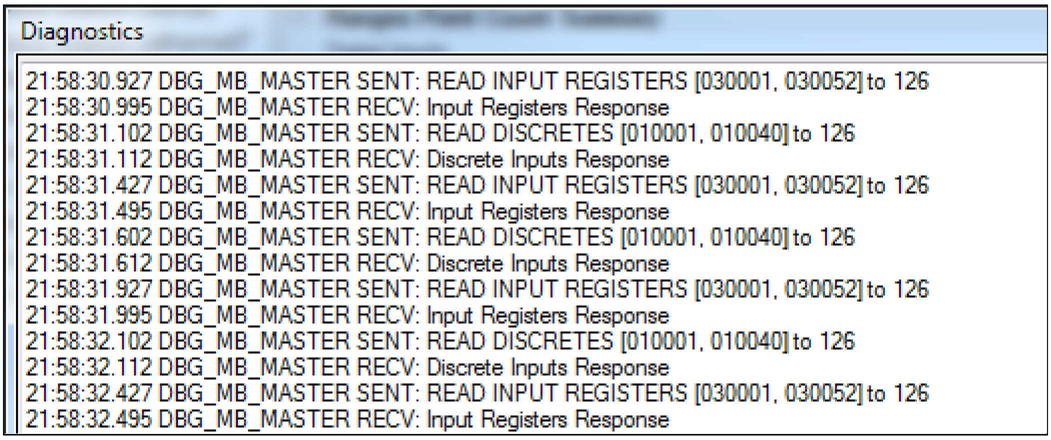
.

Interface Configuration

In order to read the values from the meter, QTech Workbench is used to configure the RTU to read the Input Registers, as shown below.



Once that configuration has been applied to the RTU and the appropriate cabling has been connected, QTech Workbench Diagnostics shows that the RTU is successfully reading the values from the device.



The Input Register values can be viewed in the Analogue Inputs section of Workbench.

Analogue Input 7 and 8 hold the Turbidity value but note that the values displayed in the table are decimal representations of the Input Registers, so are not much use in that format.

2. Unnamed (Modbus Module)		
Analogue Input: 7 [1]	16639	16639
Analogue Input: 8 [2]	13060	13060
Analogue Input: 9 [3]	16840	16840
Analogue Input: 10 [4]	0	0
Analogue Input: 11 [5]	0	0
Analogue Input: 12 [6]	0	0
Analogue Input: 13 [7]	18131	18131
Analogue Input: 14 [8]	13607	13607
Analogue Input: 15 [9]	17640	17640
Analogue Input: 16 [10]	20919	20919
Analogue Input: 17 [11]	0	0

Floating-Point Number Conversion Commands

The DATRAN XL4 *Plus* RTU supports eight different commands covering all the permutations of converting back and forth between floating-point numbers and Integers, as shown in the table below.

Convert a floating-point number to an integer		
Endian-ness	**Precision**	**Command**
Big	4 bytes (single)	`be_float_to_int`
Little	4 bytes (single)	`le_float_to_int`
Big	8 bytes (double)	`be_double_to_int`
Little	8 bytes (double)	`le_double_to_int`

Convert an integer to a floating-point number		
Endian-ness	**Precision**	**Command**
Big	4 bytes (single)	`be_int_to_float`
Little	4 bytes (single)	`le_int_to_float`
Big	8 bytes (double)	`be_int_to_double`
Little	8 bytes (double)	`le_int_to_double`

Each command is followed by six parameters. The first two are the names of the input and output registers, and the remaining parameters define the scaling that will be applied to the conversion.

```
command, input, output, input min, input max, output min, output max
```

Note that floating-point numbers will occupy either 2 or 4 of the RTUs analogue registers depending on the format (single or double precision).
The register name in the DLP just specifies the first of those registers.

Turbidity Meter Example Usage

The DLP floating-point conversion commands can be used to make the RTU convert the Input Register values into meaningful numbers. The AMI Turbiwell manual indicates that the floating-point numbers are encoded in big-endian format. That means that the most significant part of the floating-point number is stored in the first Input Register, and the least significant part is stored in the next one. So for this device we must use the big-endian version of the DLP commands (those starting with “be_”).

This short DLP example shows two approaches on how the conversion could be achieved.

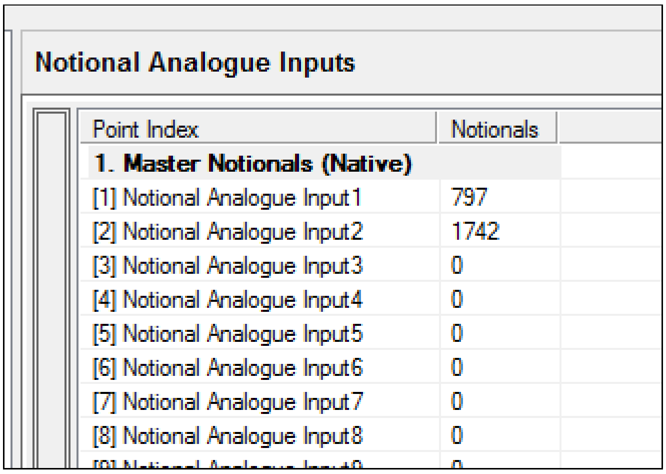
```
Proginit

progstart
    ; example 1, with the integer value as float value x 100
    be_float_to_int rain7, nain1, 0.0, 300.0, 0, 30000

    ; example 2, with the integer value using the full range
    be_float_to_int rain7, nain2, 0.0, 300.0, 0, 65535

progend
```

In the first example the floating-point value is scaled into a form in which the resultant integer value resembles the floating-point number (being 100x greater).

The process to convert the value back to a floating-point number would be to simply divide it by 100. An advantage of this approach is that the scaled value can be read and understood more easily by people looking at the number using QTech Workbench (eg 550 = 5.5 NTU).

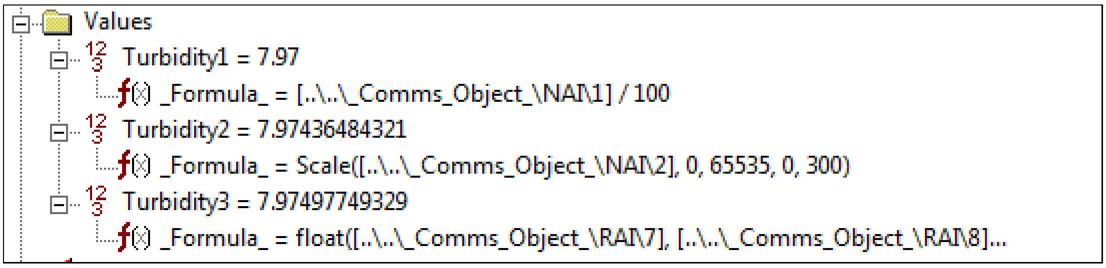
A disadvantage of that approach is that the converted value will have less precision than if the full range of the integer were used. The second example shows how to perform a conversion which results in the least loss of precision. To convert the number back to a floating-point value, a formula using the “scale” function in DATRAN VI can be used.

The figure below shows the resulting Notional Analogue Input values.


Notional Analogue Inputs

Point Index	Notionals
1. Master Notionals (Native)	
[1] Notional Analogue Input1	797
[2] Notional Analogue Input2	1742
[3] Notional Analogue Input3	0
[4] Notional Analogue Input4	0
[5] Notional Analogue Input5	0
[6] Notional Analogue Input6	0
[7] Notional Analogue Input7	0
[8] Notional Analogue Input8	0

The figure below shows how the Notional Analogue Input values can be converted into a form suitable for display and/or logging in DATRAN VI. Observe how NAIN1 corresponds to the measured value, but that the Trubidity2 value in DATRAN VI has more precision.



The Turbidity3 value shown above is computed using the “float” function directly from the Input Register values read from the meter. As such it can be treated as being the “true” value that the meter has produced, where there has been no loss of precision due to any intermediary conversion operations. It is apparent therefore, in this example that the Turbidity2 value is accurate to within 0.01% of the true value.

All rights reserved

Disclaimer

The information in this document is subject to change without notice and does not represent a commitment on any part of QTech Data Systems Limited. While the information contained herein is assumed to be accurate, QTech Data Systems Limited assumes no responsibility for any errors or omissions.

Warranty and Liability

This document, the products described herein and all QTech products in general, are all subject to the terms and conditions of business made by QTech Data Systems Limited. For a copy of the company's terms and conditions including warranty policies please contact QTech.

www.qtech.co.nz