



Knowledge Base Articles

Expression To Reverse The Word Order For Floating Point PLC Register Values

Article ID: NA-0200-0173
Publish Date: 11 Jan 2019
Article Status: Approved
Article Type: Product Issues
Required Action: Information Only

Recent Article Revision History:

Revision/Publish	Description of Revision
11 Jan 2019	Added a note that the expression only works on 32 bit integer parameters. (See end of article for a complete revision history listing.)

Affected Products:

Product Line	Category	Device	Version
DeltaV	Controller & I/O Hardware	VE4006P2 Serial Interface with 2-Ports	
DeltaV	Workstation Software	VE2162 Control Studio - Edit	

1 Introduction

This Knowledge Base Article, **NA-0200-0173**, provides instructions to allow DeltaV to correctly read floating-point values via Modbus from a PLC or other device, which has stored the floating-point value(s) in a different register order than DeltaV is able to access.

2 Background

The Modbus specification does **not** identify a register order for the two 16-bit words that make up the 32 bits of the **IEEE754** floating-point number. DeltaV uses the register order found in Modicon PLCs, which have the Least Significant Word in the lower address register and the Most Significant Word in the higher address register. If the PLC or other field device uses the opposite order, there are two ways to read the correct value from the pair of registers.

1. The **preferred** method is to modify the register values in the PLC or field device to put the values in a compatible order.
2. The **other** method is to use an Expression in a CALC block in a DeltaV Module to rearrange the word order. An Expression that will accomplish this rearrangement along with the necessary information for its use is described below.

In order for DeltaV to manipulate the bits without losing any resolution, the register values must be read into DeltaV as a **32-bit unsigned integer**. For DeltaV versions earlier than V6, the Expression must be modified to use individual **16-bit unsigned integer** registers, since the 32-bit integers were not handled correctly in older DeltaV versions.

In a CALC block, lines similar to the following must be entered in the Expression for each floating-point value to define an OUT value for each floating-point value:

```
OUT1 := (' PLC Register'=0) OR (' PLC Register'=32768)?0: (SHR(ROR(' PLC Register',7),9)+8388608)*EXPT(2,((SHR(' PLC Register',7)&255)-150))*((' PLC Register'&32768)=0?1:-1);
```

where:

the same actual register External Parameter is substituted for 'PLC Register' in each case.

Examples of actual register External Parameters are:

'//C1/I01/C05/P02/DEV01/DS06/R40129.CV' for an I/O Parameter

'//C105020106/R40129.CV' for a DST Parameter

The following portion of the expression is used to force a zero (0) value when the register values are zero (0):

```
(' PLC Register'=0) OR (' PLC Register'=32768)?0:
```

The above portion of the expression can be deleted to save processing time, but a value of 1.17549421e-38 will result instead of zero (0) when the registers both contain zero (0).

Note: The expression above only works on 32-bit integer parameters. The CALC block IN1 through IN16 parameters are floating point registers. Wiring a 32-bit unsigned integer register to IN1 will convert the value to floating point format. The SHR and ROR functions will produce incorrect results if using the IN1 through IN16 CALC block parameters.

Contact Information

Services are delivered through our global services network. To contact your Emerson local service provider, click [Contact Us](#). To contact the Global Service Center, click [Technical Support](#).

Complete Article Revision History:

Revision/Publish	Description of Revision
11 Jan 2019	Added a note that the expression only works on 32 bit integer parameters.
20 Dec 2003	KnowledgeBase Article Revision History feature added. Any revisions to this article made before December 20, 2003 are not tracked by this Article Revision History feature.

©Emerson Automation Solutions 2009-2021. All rights reserved. For Emerson Automation Solutions trademarks and service marks, [click this link to see trademarks](#). All other marks are properties of their respective owners. The contents of this publication are presented for informational purposes only, and while diligent effort has been made to ensure their accuracy, they are not to be construed as warranties or guarantees, express or implied, regarding the products or services described herein or their use or applicability. All sales are governed by our terms and conditions, which are available on request. We reserve the right to modify or improve the design or specification of such products at any time without notice.

[View Emerson Products and Services: Click This Link](#)